

# **Scalable Congestion Aware Replication framework for Mobile Ad hoc Networks**

*Thesis submitted in partial fulfilment of the requirements for the award of  
degree of*

**Master of Engineering**  
in  
**Information Security**

*Submitted By*  
**Navroop Kaur**  
**801233011**

Under the supervision of:  
**Dr. Neeraj Kumar**  
Associate Professor



COMPUTER SCIENCE AND ENGINEERING DEPARTMENT  
THAPAR UNIVERSITY  
PATIALA – 147004

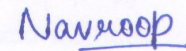
**June 2014**

## Certificate

---

I hereby certify that the work which is being presented in the thesis entitled, "*Scalable Congestion Aware Replication framework for Mobile Ad hoc Networks*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Information Security* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Neeraj Kumar* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.



Navroop Kaur

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.



Dr. Neeraj Kumar

Associate Professor,

Computer Science and Engineering Department

Countersigned by

  
(Dr. Deepak Garg)

Head

CSED

Thapar University

Patiala

  
(Dr. S. K. Mohapatra)

Dean (Academic Affairs)

Thapar University

Patiala

## Acknowledgment

---

No volume of words is enough to express my gratitude towards my guide, **Dr. Neeraj Kumar**, Associate Professor, Computer Science and Engineering Department, Thapar University, who have been very concerned and have supervised the work presented in this thesis report. He has helped me to explore this vast field in an organized manner and provided me with all the ideas on how to work towards a research oriented venture.

I am also thankful to **Dr. Deepak Garg**, Head of Department, CSED and **Ms. Jhilik Bhattacharya**, P.G. Coordinator, for the motivation and inspiration that triggered me for the thesis work.

I would also like to thank the staff members and my colleagues who were always there in the need of the hour and provided with all the help and facilities, which I required, for the completion of my thesis.

Most importantly, I would like to thank my **parents, friends** and the **Almighty** for showing me the right direction out of the blue, to help me stay calm in the oddest of the times and keep moving even at times when there was no hope.

**Navroop Kaur**  
(801233011)

## Abstract

---

Mobile Ad hoc Networks (MANETs) are types of decentralized networks which does not depend upon any existing infrastructure such as access points, routers or gateway. The nodes can communicate with each other by forming the network at fly, i.e., at any instant these can communicate with IEEE 802.11 a/b/g standards. The network consists of mobile nodes that are multi-hop, self-configured connected by wireless links. Each node randomly moves and changes its positions arbitrarily. So, the topology of network changes very frequently. In MANETs, dynamic mobile nodes are stationed in such a manner that communication between nodes does not rely on any existing network infrastructure. Due to high velocity of nodes in MANETs, fault can occur at any time due to the instability of the links.

This thesis provides solution to overcome the difficulty of faults that occur in the network by tolerating fault and providing data availability. The solution proposes for fault tolerance is by using the technique of replication of data items at the backbone nodes. A new Scalable Congestion Aware Replication (SCAR) framework is designed that overcomes the fault and increases the data availability. In this the data is replicated over the backbone nodes that are created in the network by the marking and pruning process. Thus in case the data over one node is not present it can get that from the nearby backbone node. The data is replicated depending upon the cost function of that particular data item. The data item having more value of the cost function that will be replicated in comparison to the item having less cost function. The designed algorithm is adaptive with respect to the rate of replication, time to overcome the fault and success ratio.

# Table of Contents

---

---

Certificate	ii
Acknowledgment	iii
Abstract	iv
Table of Contents	v
List of Figures	viii
List of Tables	ix
Abbreviations	x
<b>Chapter 1</b> .....	<b>1</b>
<b>Introduction</b> .....	<b>1</b>
1.1 Background.....	1
1.2 Properties of MANETs .....	3
1.3 Applications of MANETs .....	4
1.4 Routing in MANETs.....	4
1.4.1 Various desirable properties of MANETs protocols .....	5
1.4.2 Classification of routing protocols of MANETs.....	6
1.4.2.1 Proactive (Table Driven) routing protocols .....	7
1.4.2.2 Reactive (On-Demand) routing protocols.....	8
1.4.2.3 Hybrid routing protocols .....	9
1.4.2.4 Multipath routing protocols .....	9
1.4.2.5 Location Aware routing protocols .....	9
1.5 Need of new routing protocols.....	10
1.6 Fault tolerance in MANETs.....	12
1.6.1 Fault tolerance in node failure .....	12

1.6.2 Fault tolerance in network and link failure .....	13
1.6.3 Fault tolerance in transmission energy and power.....	13
1.6.4 Fault tolerance in various other approaches.....	13
1.7 Motivation.....	14
1.8 Thesis outline .....	15
<b>Chapter 2</b> .....	<b>16</b>
<b>Literature Survey</b> .....	<b>16</b>
2.1 Data replication schemes for MANETs .....	16
2.2 Cooperative caching schemes for MANETs.....	24
2.2.1 CCS (Cooperative Caching Service) .....	26
2.2.2 Data caching for benefit .....	27
2.3 Comparison of Data Replication and Caching properties.....	28
2.4 Algorithms for Connected Dominating Set (CDS) formation .....	30
2.5 Congestion aware routing protocols .....	31
<b>Chapter 3</b> .....	<b>34</b>
<b>System Model and Problem Formulation</b> .....	<b>34</b>
<b>Chapter 4</b> .....	<b>37</b>
<b>Implementation</b> .....	<b>37</b>
4.1 Virtual backbone formation .....	37
4.2 Process of locating data items.....	38
4.3 Congestion Detection.....	39
4.4 On-demand replication.....	39
4.5 Scenario in case congestion occurs.....	40
4.6 SCAR Algorithm .....	42
4.7 Network Simulator.....	43
4.8 Mobility Model .....	44
4.9 Simulation Setup.....	44
<b>Chapter 5</b> .....	<b>46</b>
<b>Results and Discussion</b> .....	<b>46</b>

5.1 Performance Metrics .....	46
5.2 Results and Performance Analysis.....	47
<b>Chapter 6</b> .....	<b>52</b>
<b>Conclusion and Future Scope</b> .....	<b>52</b>
6.1 Conclusion .....	52
6.2 Future Scope .....	52
<b>References</b> .....	<b>53</b>
<b>List of Publications</b> .....	<b>58</b>

## List of Figures

---

Figure 1.1	A generalized framework of MANETs .....	1
Figure 1.2	A simple MANET having three participating nodes .....	2
Figure 1.3	Classification of MANETs routing protocols.....	7
Figure 4.1	Scenario in case congestion occurs .....	41
Figure 5.1	Success Ratio comparison w.r.t number of nodes between SCAR and AODV .....	47
Figure 5.2	Success Ratio comparison w.r.t Rate of requests between SCAR and AODV .....	48
Figure 5.3	Success Ratio comparison w.r.t Cache memory size between SCAR and AODV .....	49
Figure 5.4	Number of hops traversed w.r.t number of nodes between SCAR-w and SCAR-w/o replication .....	50
Figure 5.5	Number of hops traversed w.r.t Rate of requests between SCAR-w and SCAR-w/o replication .....	50
Figure 5.6	Number of hops traversed w.r.t Cache memory size between SCAR-w and SCAR-w/o replication .....	51

## List of Tables

---

Table 1.1 Categories of routing protocols for MANETs .....	10
Table 2.1 Comparison between various properties of caching and replication ...	29
Table 4.1 Simulation Parameters .....	45

## Abbreviations

---

ABR	Associativity Based Routing
AODV	Ad-hoc On Demand Distance Vector
AOMDV	Ad-hoc On Demand Multipath Distance Vector
BN	Backbone Node
CADRE	Collaborative Allocation and Deallocation of Replicas with Efficiency
CBRP	Cluster Based Routing Protocol
CCS	Cooperative Caching Service
CDS	Connected Dominating Set
CGSR	Cluster head Gateway Switch Routing
CH	Cluster Head
CHAMP	Caching and Multipath
CHN	Cluster Head Nodes
CLEAR	Consistency and Load Based Efficient Allocation of Replicas
DAF	Dynamic Access Frequency
DCG	Dynamic Connectivity grouping
DSDV	Destination-Sequenced Distance Vector
DSR	Dynamic Source Routing
DREAM	Data replication technique for Real-Time Ad hoc Mobile databases
FSR	Fisheye State Routing

GN	Gateway Node
GPS	Global Positioning System
GSR	Global State Routing
HSR	Hierarchical State Routing
IP	Internet Protocol
LAKER	Location Aided Knowledge Extraction Routing
LAR	Location Aided Routing
LRU	Least Recently Used
MANETs	Mobile Ad Hoc Networks
MCDS	Minimum Connected Dominating Set
MORA	Movement Based Routing Algorithm
NS-2	Network Simulator
OLSR	Optimized Link State Routing
RK	Replica Keeper
RM	Replica Manager
RRT	Recent Request Table
RWR	Read Write Ratio
SAF	Static Access Frequency
SCAR	Scalable Congestion Aware Replication
SSR	Signal Stability Routing
UDP	User Datagram Protocol
TORA	Temporarily Ordered Routing Algorithm

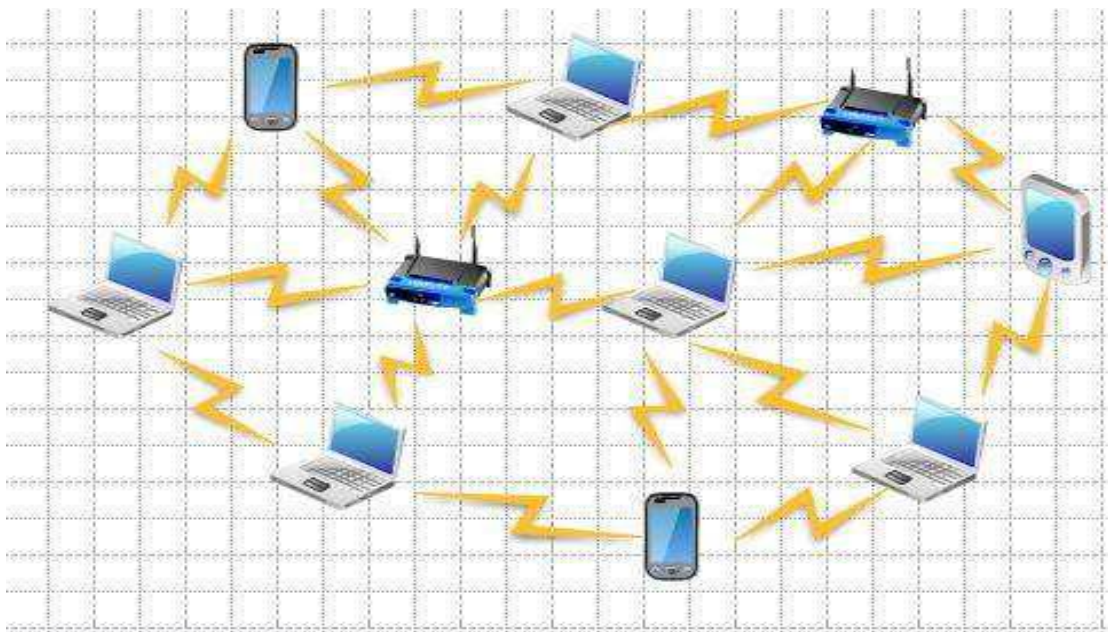
TTL	Time-To-Live
WCDS	Weekly Connected Dominating Set
WLAN	Wireless Local Area Network
WRP	Wireless Routing Protocol
ZHLS	Zone-based Hierarchical Link State
ZRP	Zone Routing Protocol



### 1.1 Background

Due to advancement in the software and the processing capabilities of devices the mobile computing has become popular and adapted widely. In fact, both radio communications and computer networks combination was first introduced by the University of Hawaii in an experimental network in 1971.

This Wireless Local Area Network (WLAN) was the first that offered star topology based bidirectional communications [1]. The technology was improved considerably during 80's and these networks made great revolution at the end of 90's as it has being employed in various fields due to the growth of internet.

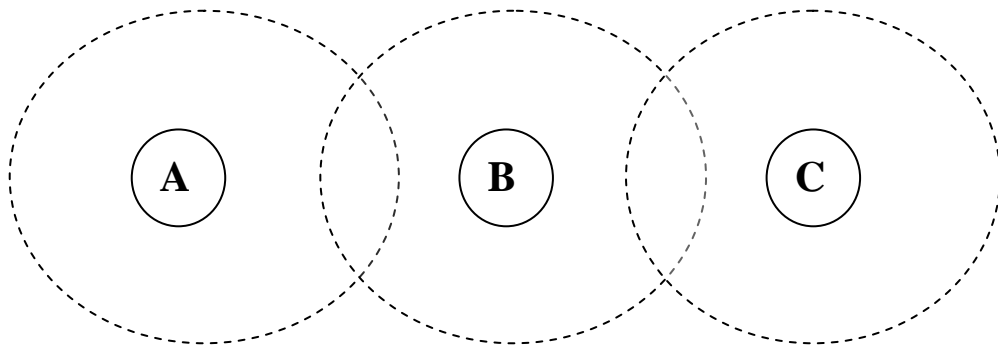


**Figure 1.1:** A generalized framework of MANETs

Due to rapid increase in mobile devices the MANETs have been gathering lot of attention during last few years. MANETs in today's world of dynamic changing technology of communication networks, plays a vital role in wireless communication.

MANETs consists of group of wireless mobile nodes forming infrastructure less dynamic network as shown in Figure 1.1. These mobile nodes in MANETs communicate with each other to forward both data and control packets using wireless connections within the wireless network [2].

Unlike a wired network, in these type of networks all the nodes are mobile or free to move independently to communicate with other nodes present in the network, so the topology changes very frequently. All the nodes here act like a router to provide multihop routing which help nodes in communicating with each other [3]. Due to the lesser range of transmission of a node the communication between two nodes get information for routing through the several intermediate nodes to route a packet. Each node here can communicate directly to the node present in its transmission range via radio wave. Each node in MANETs takes routing decision independently without depending upon any central authority like for route selection, route request, route update and for creating a new communication link with their neighbours. Generally, on all nodes combined effort whole networks functionality is dependent. A simple example of MANETs is shown below.



**Figure 1.2:** A simple MANET having three participating nodes

In the Figure 1.2, it can be seen that three nodes are present and their transmission range can also be depicted. Here node A can communicate to node B but in case node A and C wants to exchange information between each other they will not be able to do so as the node C is not present in the transmission range of node A. So in order to

communicate the node A sends information to C by discovering route through the node B.

MANETs are having no fixed infrastructure which property makes these networks robust and low-cost. There are various benefits of these networks that come along with certain challenges like facing the various variable characteristics such as load balancing, power transmission conditions and variations in traffic distribution. To form the better type of decentralized and more flexible network various new protocol and algorithms has been designed. The network may perform its functionalities in isolation and even in the situation having interface with a fixed network.

## 1.2 Properties of MANETs

MANETs consists of group of nodes in which each node is having receivers comprising of antennas that are directional, unidirectional or both. The properties of these networks are as follows [4]:

- **Distributed operation:** There is no central authority for controlling the operation of the nodes present in the network rather the control is done by the nodes themselves. Here the nodes exchange information among themselves and also act like a router for performing various functions like that of routing and providing the security in the network.
- **Hop-by-hop routing:** In these networks in case the two nodes communicating with each other are not in their transmission range then they can transmit the packets using one or several nodes in the way.
- **Dynamic topology:** Here the network topology changes very frequently as the nodes can move freely with different speed. The nodes are having the authority to move independently to communicate with the other nodes present in the network leading to establishment of network for itself.
- **Less-weight terminals:** The nodes in MANETs are having less CPU capacity, small memory size and least power saving capability.

- **Autonomous terminal:** In these networks each node acts both like a router and host as they can transmit the request for packets and also receives the packets.
- **Shared medium:** The wireless channel for communication is easily used by various sources having the required equipment and resources needed. In similar way communication medium is also easily accessible.

### 1.3 Applications of MANETs

- **Emergency services:** These types of networks can be deployed in areas where disaster occurs. At these places in the time of emergency the mobile users may need to communicate with each other using different types of wireless devices and also for maintaining connectivity to the other far areas from where they can request for any kind of help.
- **Home networking:** At home the wireless devices can also be used to form an ad hoc network so that the different devices can communicate with each other without considering the original point of their connection. This method is usually alternate to giving multiple IP addresses to each wireless device for identifying them.
- **Sensor dust:** This is also the applications of the ad hoc networks that are applicable of combining both sensor and ad hoc networks. Generally in case of dangerous situation to know about present situation of those area wireless sensors with wireless transceivers are deployed there to form an ad hoc network to give the critical situation information.
- **Conferencing:** The mobile conferencing is also one of the most common application which allow the various users to establish the ad hoc environment outside the company to discuss over the project through communicating the important matters there itself.

### 1.4 Routing in MANETs

Routing is the process of forwarding the message or packets from source node to the destination node in an inter-network. While sending packets to the distant node multiple intermediate nodes come across the way to reach destination. Routing is done generally by firstly finding the minimal path to route the packets and then to

transmit the packets over that path from sending to the receiving node. The routing algorithms decide while transmission of packet from source to destination that on which next intermediate node the packet should be passed i.e. deciding output line for an incoming packet on the node. On the other hand the routing protocols decide the minimal or the best route from source to destination through which the packet should be transmitted. There are various parameters used by the routing algorithms known as metrics such as hop count, delay or reliability etc; that help to determine the minimal route to the destination node from the source node in the network.

The routing algorithms help to determine the routing path for which it maintains the routing tables, that contain the information for routing the packets through the optimal path. In routing tables the information such as hop count, logical and physical address of the source and destination is maintained. Generally the next hop is actually an association with the destination address like when the packet is transmitted the next hop is represented as a particular node on the way to destination. The router on receiving the packet to be further transmitted associates the destination address in that packet to the next hop for sending it to the final destination.

#### **1.4.1 Various desirable properties of MANETs routing protocols**

The properties required by the routing protocols are:

- **Demand based operation:** The routing protocols should be such that they help in minimizing the overhead and resources consumption. So, the routing protocols should be reactive which means the protocol will perform operation only when the source node needs to communicate to destination node rather than broadcasting the packets in the network even when not needed, leading to the wastage of resources.
- **Distributed operations:** The routing protocol should be distributed in nature. The nodes in these types of networks are free to move independently with different speeds without depending upon any central authority. Thus the nodes can perform the functions as an independent entity and it can enter and leave the network easily.
- **Security:** The routing protocols in these types of networks work in the radio environment due to which they are prone to different types of threats related to their security. But different kind of mechanisms are discovered to mitigate these types of

threats like authentication, trust based routing, encryption and threshold cryptography etc.

- **Quality of service support:** The routing protocols should be such that they support the quality of service parameter such as throughput, delay and packet delivery ratio. The protocols designed must be having better output in terms of these parameters so that routing can be done achieving better performance.
- **Power conservation:** In MANETs the life of mobile nodes is dependent upon the battery life. Like here the wireless devices such as the mobile phones, laptops and PDAs are the nodes which work on the battery power. So the criteria should be to reduce the power consumption. Thus, the routing protocols should be designed in such a way that they must be having some sort of mechanism that the nodes not participating in routing should go into the sleep mode.
- **Loop free:** It should be ensured that the ad hoc routing protocols are free of loops so that consumption of the channel bandwidth will be reduced. Due to which the overall performance of the network will be increased and also the resources will be utilised in a better way without any wastage.

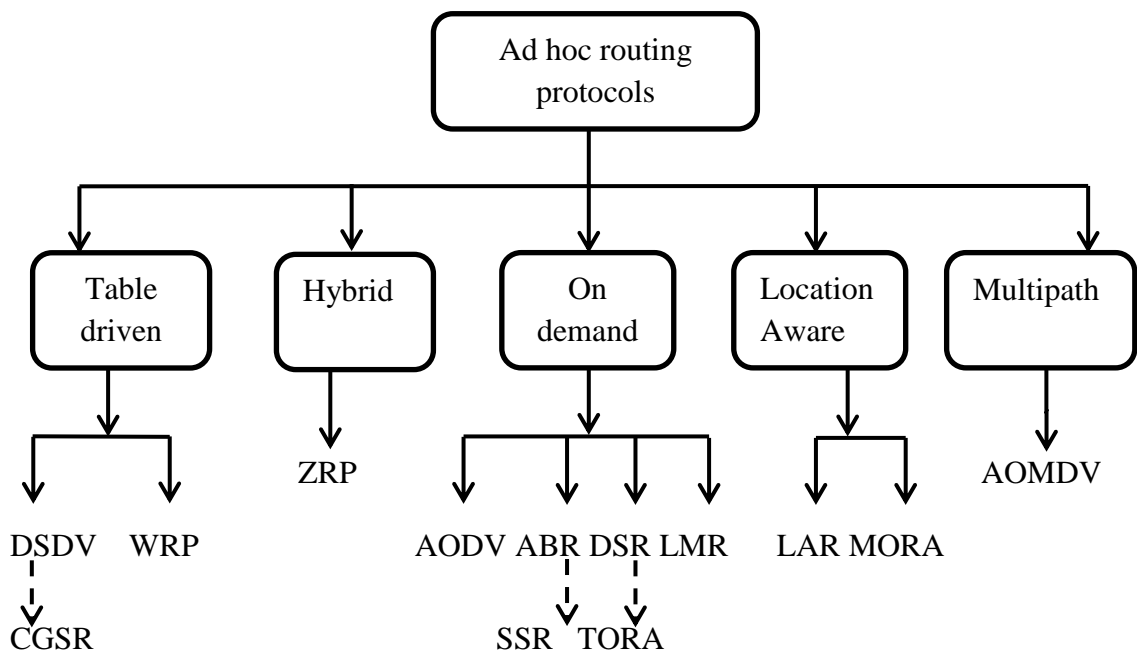
The routing protocols are designed with a view that these properties should be satisfied. There are many routing protocols that fulfil these properties but not all. The protocols are still under designing to cover all these properties and also extended with more functionality.

#### **1.4.2 Classification of routing protocols of MANETs**

There are many routing protocols existing for ad hoc networks considering different implementation scenarios. But always the main aim have been to device a routing protocol that minimizes the packet loss overhead, control overhead and energy consumption while maximizing the throughput. The routing protocols in ad hoc networks can be divided into various categories as shown in Figure 1.3 below.

All the nodes in MANETs are mobile and form the dynamic network in an arbitrary manner in comparison to the wired networks. The intermediate nodes in MANETs act as routers. They help in finding and maintaining the route between

source and destination node. Thus the routing protocols designed for the wired networks cannot be used in the wireless networks. Many protocols have been designed till now for MANETs. These routing protocols depending upon the criteria of managing the routing tables are divided into the following categories. These categories are mainly the proactive routing protocols (Table driven), hybrid, reactive routing protocols (On-demand), multipath routing protocols and the location-aware protocols.



**Figure 1.3:** Classification of MANETs routing protocols [2]

#### 1.4.2.1 Proactive (Table driven) routing protocols

Table driven protocols always keep updated record of paths from every single node in the network to the other [2]. Each node in the network maintains routing information in its routing tables and to keep this information as recent, the continuous route updates are passed through the whole network. Each node sends the information about any change in the network topology in the whole network so that other nodes can keep their routing table reliable. However, no latency issue comes in these as the routes to the different destinations are present already. Moreover there is also one good point in these protocols that each node maintains some sort of routing

information. The frequent updates about the network topology makes the nodes active all the time but there comes the problem of power consumption as it becomes more than normal. These types of protocols are not good to be employed in highly dynamic networks because to keep routing table of each node in the network fresh involves extra overhead. Sometimes it happens that the duplicate entries in case of route to the particular destination, occurs in the routing table because of the periodic updates.

Wireless Routing Protocol (WRP), Global State Routing (GSR), Fisheye State Routing (FSR), Destination-Sequenced Distance Vector Routing protocol (DSDV), Hierarchical State Routing (HSR), Cluster head Gateway Switch Routing protocol (CGSR) and Zone-based Hierarchical Link State Routing protocol (ZHLS) are Table driven routing protocols.

#### **1.4.2.2 Reactive (On-demand) routing protocols**

Source-initiated routing protocols or reactive protocols are an approach opposite to the Table driven routing protocols [2]. These protocols represent the category of routing protocols generally where the route from source to destination is created only when the source requests the route. When the route is requested by the source a route discovery procedure is invoked and special request packets are routed to the network from the immediate neighbours. Once the single route or more than one route are obtained from source to destination, the route discovery procedure comes to an end. When the route has been established, the route maintenance procedure helps to maintain the routes alive for the duration of their lifetimes until either the route is no longer desired or until the destination becomes inaccessible along every path from the source. The optimal route to particular destination remains in the routing table of the node for the time it is no longer required.

Ad-hoc On-Demand Distance Vector Routing (AODV), Temporally Ordered Routing Algorithm protocol (TORA), Dynamic Source Routing protocol (DSR), Cluster Based Routing Protocol (CBRP), Signal Stability Routing (SSR) and Associativity Based Routing (ABR) are On- demand routing protocols.

### **1.4.2.3 Hybrid routing protocols**

Hybrid protocols are combination of features of both the On-demand and Table-driven routing protocols. In these initially some proactive protocols are used to establish the route and then through reactive flooding it serves the demand by additionally activated nodes. Both the reactive and proactive protocol performs its operation well in different scenarios. It forms a link between the various types of protocols for routing and the performance can be made effective to certain extent. Static routing is used at the boundaries of the network whereas the on-demand routing is used inside the network as here the routes are changed frequently. Thus the proactive protocols can be used in a defined or small domain whereas the reactive protocols can be used to find the route to the nodes outside the defined domain or region [5]. Zone Routing Protocol (ZRP) is an example of the hybrid routing protocol.

### **1.4.2.4 Multipath routing protocols**

The multipath routing scheme helps to create multiple routes from source to the destination node. This kind of routing protocol can be used in case of node or link failure so that an alternate route to the destination can be used to route the packets. The bandwidth is also an issue in case of routing in the networks. But with the discovery of the multipath routes the bandwidth in the links is used efficiently with more reliable delivery of the needed information. Moreover this type of routing also helps to route the packets to destination in case congestion occurs by sending the packets through alternate paths. The multiple routes are formed proactively or on demand. The multipath routes are of great significance as nodes are free to move independently with different speeds due to which they get disconnected very easily. Caching and Multipath Routing Protocol (CHAMP) and Ad Hoc On-Demand Multipath Distance Vector protocol (AOMDV) are the example of multipath routing protocols [6] [7].

### **1.4.2.5 Location Aware routing protocols**

Location-aware routing methods are based on the assumption that every single node in the network is aware about the location of every other node. The simplest method to find co-ordinates of these nodes is Global Positioning System (GPS). This

determination of location proves helpful for the routing protocol to find the routes. Location-Aided Routing (LAR), Location Aided Knowledge Extraction Routing (LAKER), Movement- Based Algorithm (MORA) are some of the location aware routing protocols [8] [9] [10].

**Table 1.1:** Categories of routing protocols for MANETs

<b>Routing Protocols in MANETs</b>	
<b>Table Driven Routing Protocols</b>	<b>On Demand routing Protocols</b>
Destination-Sequenced Distance Vector Routing Protocol (DSDV)	Ad-Hoc On-Demand Distance Vector Routing (AODV)
Clusterhead Gateway Switch Routing Protocol (CGSR)	Associativity Based Routing (ABR)
Wireless Routing Protocol (WRP)	Signal Stability Routing (SSR)
Global State Routing (GSR)	Dynamic Source Routing Protocol (DSR)
Hierarchical State Routing Protocol (HSR)	Temporarily Ordered Routing Algorithm (TORA)

### **1.5 Need of new routing protocols**

MANETs are the infrastructure less type of networks in which the nodes are free to move independently with different speeds. The nodes communicate with each other by forwarding the data packets or messages. The nodes have to find such a path through which when the data packets are transmitted the performance of the network will not degrade. The route to transmit the packet from source to destination should be the shortest path among all the other routes to that particular destination. There are many routing protocols discovered for MANETs with a view that they will minimize the packet loss overhead, control overhead and energy consumption while maximizing the throughput. The designed protocols for these networks perform well in regard to the above parameters but they do not acquire all these every time. This

is the reason that there is still need to discover more protocols for these networks that will fulfil all these metrics. The protocols should be designed to improve the overall performance of the network in various aspects. In MANETs the routing protocols are required because of the below described reasons:

1. The nodes in the MANETs are free to move with different speeds and the topology connecting the nodes in the network is very dynamic. So it requires that some kind of regulations may be there so that the packets from source node to the destination node can be transmitted in a way which will not degrade the network performance.

2. The previous protocols existing for these networks perform the functions with least expected behaviour in case of highly dynamic topology condition. Thus it is needed to develop such routing protocols that they can show the better performance in case they are presented in dynamic inter-connection topology.

3. The working of routing protocols is to transmit the packets from the source node to the destination node but for that it may involves the presence of the intermediate nodes. So it usually requires the memory size of the nodes present in the way to destination, the processing power and also the power of the working devices which can also cause the node failure. Thus the existing protocols puts heavy load on the intermediate nodes by consuming their power and memory. Therefore, the new protocols should be designed that will help facing these problems that comes while routing.

4. The protocols are designed keeping in view the strategy of routing the packets from one node to the other but these protocols does not satisfy the requirements of the users that wants to use these networks just because of the reason that they exhibit dynamic topology and show self- starting feature. Thus it is needed that the new routing protocols incorporated with these features should be developed.

5. In case of failure of link the existing routing protocols takes much longer time for convergence but this is not a case with MANETs.

6. The existing routing protocols are not loop free, there exists looping problem either short lived or long lived. Therefore, there must be present such routing protocols that should be loop free so that the network performance will not degrade.

7. Moreover the techniques used to solve the looping problems in the existing routing problems cannot be applied in case of the ad hoc networks. Thus such routing protocols should be developed in which the loops should not occur. It will also help to minimize the loss of bandwidth which may be occupied in case loops occur in the network. This will enhance working of the overall network by providing more space.

## **1.6 Fault tolerance in MANETs**

MANETs are the type of wireless ad hoc networks in which the topology changes frequently as the nodes are free to move independently without depending upon any central authority. The nodes in these networks can communicate with the other nodes only if they are present in the transmission range of each other. But as in wireless networks the nodes are free to move due to this basic reason the topology changes. Due to change in topology the nodes may move out of their transmission range. There are many consequences in case the nodes move out of their transmission range, the main are node failure and link failure and also more power will be consumed in transmitting the packets from source node to the destination node. Moreover there are chances of occurrence of congestion due to which the link can also fail. The performance of routing may decrease as a result of these failures or causes in the network. Hence the new routing protocols should be developed which may overcome these failure which occur in the network and perform routing in an efficient manner. Moreover the network would be adaptable to modify the routing paths to overcome these problems. There are many protocols designed to tolerate the fault that may occur in the network. The fault tolerance is done by designing different protocols in case the fault occurs in the network due to failure or congestion.

### **1.6.1 Fault tolerance in node failure**

The nodes in MANETs communicate with the each other only if the node which wants to communicate with the other node is present in the transmission range of that node. But in case the nodes with which the other node want to communicate does not lies in the other nodes transmission range then the intermediate nodes are used to transfer the packets to the destination to help in communication of the two nodes. In transmitting through the intermediate nodes there is also the chance of failure of the intermediate nodes due to which the communication is not possible. Therefore, when

the nodes does not lie in the transmission range of the other node the failure of node occurs. To overcome the node failure problem many protocols and algorithms are designed to be used in these networks. The routing protocols developed will then help to find the alternate path to the destination node in case on the route any intermediate node fails.

### **1.6.2 Fault tolerance in network and link failure**

The fault in MANETs can also occur because of link or network failure. The network fault occurs in case any network component or device present in the network partially or fully fails and does not able to work properly. Also the other causes of the network failure may be the malfunctioning or as a result of natural calamity or disaster the network breaks. On the other hand the link failure occurs when the node moves out of its transmission range or from the cluster in which it is present. Congestion may also be one reason due to which the link may block which ultimately leads to the link failure, as the packets cannot be transmitted through that route or link. The link failure also leads to the failure of whole network. To overcome this issue many approaches has been proposed. There are various types of protocols that are developed to avoid this link failure, by sending the packets through alternate path.

### **1.6.3 Fault tolerance in transmission energy and power**

The life of nodes in MANETs depends upon the battery life. Like the different devices such as mobile phones, laptops form the ad hoc network and these all work depending upon the battery power. Thus the transmission energy and power is an important issue in the MANETs. Moreover the energy and power is also used for discovering the route to destination, route selection and for repairing the link or node failure in the network. So in this way battery power is commonly used in the network. To avoid the problems occurring because of power related issues research has been carried by proposing various approaches.

### **1.6.4 Fault tolerance in various other approaches**

There are various protocols and algorithms designed for increasing the network performance thereby reducing the faults that occur in the network due to various

reasons. There is a novel algorithm that achieves the optimal utilization of resources and fairness.

## **1.7 Motivation**

In MANETs routing is a critical problem but at the same time accessing data items in these networks is also an issue to be considered. In case of wired networks it is easy to access the data items because the nodes are static. But in MANETs as the nodes are free to move independently with different speed, it is difficult to even locate the data items in the network. This is so because there is no central authority to control the working in these networks. Due to which it is difficult to learn the position of particular data items in the network. Also with the movement of nodes topology in MANETs changes very frequently. With the change in topology sometimes the node or link failure occurs or in some cases the whole network fails. Due to link or node failure generally the data availability in the network decreases.

One solution is to achieve fault tolerance and to increase the data availability and for which the data replication is done. Thus the SCAR framework is designed for this purpose. If in a network large numbers of request packets are broadcasted over a node as a result of which congestion will occur on that particular node. Due to which the node may fail or that link may fail. So for fault tolerance problem in case of congestion and to make data item available replication technique is used. That helps to increase data availability by ensuring replicas of data items on the closer nodes of the node which need that data.

This problem is being simulated in the MANETs and then the effects of it are analysed. Network Simulator (NS-2) is used for performing simulation. It is a type of simulator that consists of all the protocols for simulating the network topologies that are present. Having implemented the problem, various performance tests are made on different parameters to compare the network performance after employing the described strategy to check whether it gives the better results in terms of different parameters than the previous work.

## **1.8 Thesis outline**

This thesis consists of 6 chapters and they are organised as follows: Chapter 1 explains the introduction about MANETs, Routing protocol, need for new routing protocol and fault tolerance approaches. Chapter 2 describes the survey on various replication and caching techniques, connected dominating set formation algorithms and the congestion detection strategies. Chapter 3 introduces the assumed system model and the problem statement. In Chapter 4, the simulation study required to carry out this work is described. Chapter 5 shows the results of simulation. Finally, in Chapter 6, the thesis is concluded and future scope is given.

## Chapter 2

### Literature Review

---

In MANETs for ensuring the availability of data items and to improve the efficiency of accessing those data items various data replication schemes and caching techniques can be used that are reviewed in this chapter. Data replication generally overcomes the data losses when the connections in the network fails by making the data available even under circumstances when fault occurs [11]. The prior research on various replication schemes and caching techniques for MANETs is surveyed in the section 2.1 and 2.2. Later on the comparison between each of the caching techniques and the replication schemes is done in section 2.3, respectively for further elaborating the concept of data availability that out of the two which technique is better to be used in the approach for enhancing the availability of data items in the MANETs. Then the work related to Connected Dominating Set (CDS) formation algorithm problem in the literature is reviewed in section 2.4. At last in section 2.5, the congestion aware routing protocols are reviewed. Generally the problem behind formation of these protocols is to maintain the availability of data in the network. This problem is solved by replicating the data items over the various nodes in the network. There are various techniques of data replication available in the literature that was being reviewed in this chapter.

#### **2.1 Data replication schemes for MANETs**

In case of link or node failure due to mobility of the random nodes the main problem which comes is that data availability in the network decreases. Thus the data replication scheme was employed for solving the data availability problem in the system [11]. Moreover, the replication scheme helped in decreasing the number of nodes through which the data item is to transmit to decrease the overall distance from source to the destination. As the replication process helps in replicating the data items over the closer nodes from the source node. In this section, mainly the various types of replication schemes designed for MANETs were reviewed. The detailed survey on

the various replication techniques for MANETs, there classification and comparison was reviewed from [12].

Hara *et al.* [13] proposed the most commonly known technique of data replication in case of MANETs. In the proposed scheme three schemes of allocating the replicas were given. Each of the schemes had been studied for both the sequentially updated systems and also for the update-less systems. The main parameter taken for making decision for replication in these schemes of allocating the replicas were frequencies with which these data items were accessed. There were various assumptions made in the proposed scheme of data replication. Few of the assumptions made were as listed below:

- The frequency with which each of the data item was accessed were known at the starting and it did not change in the whole process.
- There was very less memory space with every mobile node or host for replicating the data items.

The following described three schemes of allocating the data items was proposed for the network surroundings which were not having any mechanism for updating the data items:

#### *Static Access Frequency (SAF) Scheme*

In this method low traffic and overhead was created as every node gave the decision for replication depending only upon frequency with which it accesses that data item. So there the client replicates the data for its requirement and did not take into the account any replication decisions or access frequencies in the neighbouring nodes. Generally, every mobile node check the access frequency of each data item and then placed the data items in descending order according to their access frequencies. Afterwards each node chooses some of the topmost nodes from this order to replicate. Some of the nodes that were chosen for replication were based upon the number of empty spaces for replication present in the memory of that particular node.

Low overhead was created during processing and also low message traffic generated in it. But this method generally gave low accessibility to the data items because if in case there was similarity in the characteristics of accessing the data items

of many mobile nodes then they may usually replicate the similar data items. So this scheme of allocating the replicate did not considered much in the long run.

#### *Dynamic Access Frequency (DAF) Scheme*

In the SAF method each node accessed data item for itself. It may happen that all nodes accessed the similar data item and in the network there was present duplicate data which lead to wastage of the memory space. In order to remove the concept of having replica duplication, dynamic method was used which suggested that if there were present replica duplicates between the nodes then the neighbour having low frequency of access to the data item changes its replica to another data item. In this scheme it was considered that in case the data item was owned by the neighbour of the data item node then that node changes the replicated data item over itself. Moreover in this dynamic access frequency scheme the overhead involved in processing and the traffic created in forwarding the packets was higher than in case of the static access frequency scheme. But in this method the data availability usually get increased as the numbers of replicas over each node were distinct due to which in turn in network the number of distinguished replicas over each node gets increased.

#### *Dynamic Connectivity Based Grouping (DCG) Scheme*

In this method replicas were shared between stable and larger group of nodes unlike the concept that was used in DAF scheme. It divides the network to form bi-connected groups. Usually this scheme forms the bi-connected components and each of those components was called a group. In this scheme even if single mobile host was not present the bi- connected groups were not separated and in turn it gives high data availability. Every component then take into consideration the access frequency of the data item after forming the bi-connected components to determine that how many times the particular data item is accessed in the group. Then the data items are arranged on basis of the access frequencies and afterward it was determined that which data item was to be replicated until the memory slots of the nodes present in the groups are consumed fully. This method could be used to share multiple different types of the data item replicas between the groups created in the network. Thus we can say that the data availability problem can be corrected using this scheme. But overhead involved in this scheme was higher than that was in SAF scheme and DAF

scheme. Also the traffic generated due to messages was also very much in this scheme. The reason behind these was that more information passed during every allocation slot by the mobile hosts. The previously described were the three schemes of allocating the data items that was proposed for the networks which were not having any mechanism for updating the data items. But for the networks having involved the data updating mechanism [13], the previously described schemes were extended to control the effects created by periodic updates:

1. Modified Static Access Frequency plus (M- SAF +)
2. Modified Dynamic Access Frequency plus (M-DAF +)
3. Modified Dynamic Connectivity Based Group plus (M-DCG +)

In M- SAF +, M-DAF + and M-DCG +, the decision for allocation of replica of data items was done on basis of the Read Write Ratio (RWR) parameter of the data items in place of the access frequency as were used in the previous schemes. Here RWR signified the probabilistic ratio of the access request sent by the mobile node for data item  $d$  in time period  $\Delta t$  over the probability of an update for the data item from the node that owns that particular data item. It means that if the RWR is high there were more access requests to the data item than the number of updates to that data item. During replication the data items with high RWR values were considered so that the consistency of the distinct replicas was maintained in the network for longer period. The major disadvantage of the M-DAF + and M-DCG + was that in case the number of nodes in the network become more they can create a large broadcast storms [14]. The reason for this was that while allocation period each node in the network for each of its data item broadcasted its access frequency or RWR value which can create a storm.

Moreover, the concept of load balancing between the mobile nodes was not considered in these schemes. Sometimes as after replication of the data items the total number of accesses on basis of access frequencies for every mobile node was not balanced. Also, the host which does the replication of the more common data items than others used large amount of energy. In these schemes, it was not be possible to find the reliable hosts to replicate the data items as the nodes move randomly and also failure of nodes occurred frequently.

Pabmanabhan *et al.* [15] proposed another data replication scheme, Data Replication Technique for Real-Time Ad Hoc Mobile databases (DREAM) which provides data availability and balances energy level in the network. Here time limitations were considered as metric for considering the access frequencies. Usually the data items that were accessed frequently by the jobs having time limitations were given much priority over the others. In this the data transactions were used for choosing the replicated data. The data transactions were classified here depending upon their read or write status or their time limitations. In DREAM also the replication decision is done depending upon the parameter like how frequently the data items were accessed similar to [13]. This method gave replication decision based on other parameters also like: link stability, data types, remaining power and transaction type.

Mondal *et al.* [16] proposed Collaborative Allocation and Deallocation of Replicas with Efficiency (CADRE) that helped in replication allocation among nodes but collaboratively make decisions for de-allocation. CADRE performed replica de-allocation and fairness in replication. The thrashing conditions which occurred were controlled by the CADRE by taking decision jointly for allocation and de-allocation among nodes in the network. The thrashing condition comes when mobile nodes waste resources for re-allocation of similar data rather than replying to the other nodes requests.

The replication decision for replicating the data items in MANETs was mostly done depending upon the condition that how frequently that data items were accessed. Thus those data items were replicated more times that were accessed very frequently even by the lesser number of nodes in the network. But the criteria should be such that the replication decisions which help to serve the requirements of the whole nodes in the network should be avoided. But CADRE did not relate to scalability and the performance results were limited in this to 50 nodes only in an area.

CADRE formed the network like hierarchical structure that consisted of the mobile hosts some of which acted like the Gateway Nodes (GNs). These nodes were used basically for the purpose of replication and passing queries in the network. These nodes are selected from among the other nodes having large bandwidth, higher energy and higher capacity. In case the nodes in the network were similar in context to the

capabilities like energy, bandwidth and memory then the gateway nodes could be such that they were being affected by these capabilities. Due to this very reason as there was much load on these nodes the network delay occurs in the network while processing the requests.

Mondal *et al.* [17] proposed another dynamic replica allocation scheme called Consistency and Load Based Efficient Allocation of Replicas (CLEAR), that enhanced data availability in the MANETs. CLEAR assumed that the change in value of the attribute of data item helped to determine the estimated cost of maintaining the required consistency for the replicas of the data item.

CLEAR similar to the CLEAR was formed depending upon the hierarchical structure. In this instead of GNs Cluster Head Nodes (CHNs) were present that helped to take replication decision for their individual clusters and also helped to validate the data. The CHNs were elected from among the other nodes in the cluster based upon the various features such as memory, energy and bandwidth. The CHNs were responsible for managing all the activities to be carried by the nodes. So for this very reason every node in the particular cluster send the replica order and update the other information to the CHNs of their clusters. These nodes after getting all the data, reviewed it to consider that which data items required to be replicated. The CHNs job after reviewing the data is that select the particular data item that is to be replicated and also the mobile hosts over which it is to be replicated. In this method replication decision was taken based on the access frequency and data size of an item. On the other hand, the parameters used for replica node selection were load, energy constraints and memory of the mobile host. This method used to calculate the effectiveness of the cost-function parameter of the replication for a data item for the purpose of giving final replication decision.

Mondal *et al.* [18] proposed Economic model for efficient dynamic Replication (EcoRep), a dynamic data replication scheme for mobile peer-to-peer networks which was depended upon the economical model estimated while query requests and allocation of the replicas. Thus, for this purpose the price for each data request is set by the EcoRep system in the network and if the requester needs this request to be served they must firstly pay this price to the node by which the request is to be served.

Depending upon the parameters like number of users accessed, number of replicas existed of that item, consistency and the access frequency the cost of particular data item was calculated.

This scheme minimized the free riding hosts in the network by imposing the query pricing criteria for each query entered in the system. The nodes that do not participate in the replication of data items are known as the free-riding nodes. But these kinds of nodes were not present in the EcoRep system as each node required currency for requesting regarding any query to the network. Usually the nodes then replicate those data items that save them the most currency after analysing the system. There are various other metrics over which the replication decision depends like the load, energy and network topology, mainly the factors those factors due to which the price of data item is affected. The case considered was that when the request was sent to the node that going to serve the request has a high load, then serving node could not give the good quality service in terms of response time as the price of that request would decrease. But the scheme did not consider the scalability in means of overhead created by the message.

Thanedar *et al.* [19] proposed the expanding replication ring technique for replication of data items. This technique was used to improve the data availability in pull dependant information dissemination systems. In the pull dependant systems the mobile hosts act as the server for its support to provide data that they require. In this approach the server is supposed to be a static entity and was known by every node in the system. Thus in this approach the network type formed and the types of data request sent were different from the previously discussed techniques. The central server in this managed the job of taking decision for replication of data items. Server managed the replicas of requested data items in such a way that the energy that was spent by nodes in the packet transmissions got reduced. This was because the server helped to increase probability of locating the data items by the node in its close vicinity. In this technique, for every data item the set of hop count values were maintained by the data server (i.e. two sets [1,3,5] and [2,4,6] ). The server chooses the set of hop count values (i.e. [1,3,5] ) for a particular data item, every time the value of access frequency for that particular data item goes beyond the threshold value. Afterwards it started to replication on every node in the chosen hop set.

Chen *et al.* [20] proposed an integrated data lookup and replication scheme for the purpose of group data access. Group data access means the collection of nodes serves the set of data items for the usage of all the other nodes or members in the group. The members of a group exchange information with the help of underlying routing protocols. Each node flooded the periodic advertisement message, which usually carried the sequence of data items replicated in that node. Local data lookup tables were used to perform data lookup which was filled with the advertisement messages.

A predictive scheme for data replication was used in [20], which predicted the network partitioning and from each partition replicates the important data items. It was assumed in this that the movement of the nodes were predictable and with the help of speed of nodes and their present designation, the next partition could be predicted. It was supposed that this information was taken from the routing layer that performs the location based routing protocol and the information required was taken from this routing protocol. In [20], the cross layer design was described and routing was also described as the part of it.

The nodes with more memory space, energy and bandwidth were used by this scheme to place large number of replicas over them so that the energy was utilized properly without any wastage. In this scheme rate of sending the message was adopted over the ongoing messages in order to ensure data availability in the large scale network. Like all other approaches this approach was also the proactive in which the replication of data is done on the nodes only. This approach also provided the data lookup and replication schemes but was not scalable solution due to broadcasting of the alternate advertisement messages. Moreover, it had the complicated cross layer paradigms which may be difficult to employ in the real case environments.

Yu *et al.* [21] proposed a scheme of data replication by grouping the nodes into hierarchical groups, an optimal replication method for maintaining consistency in the large networks. In this nodes were grouped to form clusters in a hierarchical order. A Distributed Hash Table Replication (DHTR) was used for locating and accessing data items. The cluster based network structure helped to decrease the communication overhead of the network. In larger networks communication problem was one of the main constraints. DHTR formed a network structure of non-

overlapping cluster groups and there was a Cluster Head (CH) for managing each cluster. It was supposed that the number of CH in the group were familiar to the nodes in the group and these were constant in number. Over the CH each mobile host in a cluster was registered in its group. In these structures there were present Replica Managers (RMs) and Replica Keepers (RKs). The responsibility of the RMs was to store the replicas, response to the replica requests and inform about the replicas status to the CH. Also RKs were maintained for each data item present in the system. A common hash function was used by each CH to correspond the cluster id of that data item's RKs. One of the main advantages of this scheme in case of flooding based query and update propagation method was that it formed tree like exchanging architecture where each update and propagated message was passed to CH and RKs.

This method targeted scalable query but did not provide the replica management or any constraint related to which data was to be replicated and where. The target of this scheme was basically the shortcomings of the cluster-based replication algorithm. Moreover the minimum memory space for keeping the replication nodes was also not defined and the data accessibility of the system was also not surveyed. But this scheme shows better performance in case of the consumed energy and delay in propagation. There were some other critical situation in this like the clusters formed may not be same in size and the other nodes in the cluster other than CH can be present in the other clusters. This method required the support of underlying routing protocols, usually that of AODV routing protocol. But it caused problems related to scalability in larger networks. Their performance results were limited to 250 nodes in a region.

## **2.2 Cooperative caching schemes for MANETs**

In Internet for the better performance of the web services cooperative caching schemes were used widely [22]. These schemes helped to use the cached data in a better way by cooperatively using the cached data by all nodes. The cache based systems in MANETs allow the nodes to share the cache data or to show the path to the data item for providing the better data availability and lesser the delay rates. Caching techniques helped to reduce the cost involved exchanging the information between the nodes. Thus in turn of which the required bandwidth and the energy of

the system is not consumed much. In the rest of the section various caching schemes were reviewed.

Yin *et al.* [22] proposed the three simple caching techniques that could be employed as the support in the middle of routing and application protocols. The three techniques proposed were CachePath, CacheData and HybridCache.

#### *CachePath*

In this technique for providing cache data the transport helps to provide the cache path to the data item on its neighbour cache. The node need not require every data of cache on the path from all the data going through that particular path. As the node did not required caching all passing data but in place of that it just stores the sequence of data paths when it is nearer to the caching hosts rather than the data source. This technique generally saved the caching memory instead of caching the data items itself.

#### *CacheData*

In the CacheData technique the path was not cached but instead of that the data was cached by the transporter node. In case the passing by data was accessed frequently by the nearer nodes then the transporter node chooses that data to be cached. This technique was not used widely as it required large memory space for processing in comparison to the previous schemes. There were certain rules imposed in this CacheData technique over all the nodes for removing the duplicate caching of data by the various nodes coming in the way from node that request data to the source node. The rules imposed are as described: If the same node request for some data items many a times then the corresponding node did not cache the passing by data so that the bandwidth of the path could be saved and it did not get wasted. The requests were passed to the source and validated only when no new cache of the similar data item comes into, until the end of the Time-To-Live (TTL). The TTL was the value given to each cached data. It was basically the time for which the data was there to be cached.

#### *HybridCache*

This technique was combination of both the CacheData and CachePath. This technique could use the CacheData or the CachePath schemes to perform its own

working criteria depending upon the certain parameters like TTL value and the size of the data item. So it could be decided upon which to be used at which condition. The CacheData technique was employed in this in case the data item to be cached was small in size enough to be cached instead it could cache the path. Generally the decision was taken depending upon the parameters.

In the caching techniques one of the important thing to be considered was that what were the considerable items to be cached when the cache is full, rather than ignoring the important items also. In this technique on employing either of the two techniques, the cached items to be replaced were selected by the mobiles nodes based on the two metrics i.e. distance and frequency with which the data items were accessed.

### **2.2.1 Cooperative Caching Service (CCS)**

CCS [23], in the same way as in [22], supports the middleware level in between the routing layer and the application layer protocols. This service used the so called cache resolution strategy, the cocktail scheme. This scheme was used to found the requested items. Whereas on the other hand the cache management strategy was used to decide that which data was to be cached by using some simple rules to handle the situation.

The Recent Request Table (RRT) is maintained that keeps in it the previously requests that were received. In the cache resolution strategy this table was used. In it when the request comes to the node, at first the node checks its own memory and in case the data was not present in the memory then the RRT was checked. This table was used frequently as it helped searching the data items and ignored searching of the similar data items that were searched previously by avoiding the duplicated broadcasting for the similar data item which was searched previously. After searching in the memory and the RRT table even if no matching was found then the adaptive broadcasting was started by the CCS, which also reduce the number of nodes that the flooded packets could travel which in turn reduce the wastage of energy in the system.

For searching match the adaptive flooding technique was used to find the requested item's cache in the neighbour. Even after if it was not able to found the

cache of the requested item, then employing the usage of underlying routing protocols for sending request to the source of the requested data item or to the server. But on the way to source if the request found the requested entity in the intermediate nodes cache, it sends the item to the requester and stops forwarding the request. In [23] these three steps strategy was the cocktail scheme.

CCS also worked on how to solve the problem of deciding that in the limited cache which data items were to be preserved. Depending upon the current position in the neighbourhood CCS classified the items as the primary items or the secondary items. Thus the existence matters much in these services. The item had priority in the cache depending upon whether it was primary item or the secondary item. In case the data item was primary, it had higher priority in the cache. The Least Recently Used (LRU) algorithm was also employed. In case if the data item was secondary, then the LRU algorithm which was used for management of the cache was applied to the previously existing cache. CCS similar to [22], was also used for cached data items the TTL-based consistency control mechanism.

### **2.2.2 Data caching for benefit**

Tang *et al.* [24] considered the optimizing problem of reducing the total cost involved in accessing the data items in case of ad hoc networks. The problem is applied only to the networks that were having limited resources or memory space and the multiple data items. It was usually investigated that this problem was NP-hard. So for this reason the polynomial-time approximation algorithm was proposed. This algorithm helped to achieve reducing of the total cost of accessing the data item by at least one-fourth of the minimal solution point. They had also given another version that provided the results closer to the central one. That version was the distributed version of their approximation algorithm. So the approximation version gives the better performance approximate to the solution. The greedy approach was followed in the proposed approximation algorithm. In the approximation algorithm the approach followed was that the data items were cached in the memory maximizing the reduction in the total cost involved in accessing the data items in a greedy manner.

## 2.3 Comparison of Data Replication and Caching properties

Replication was referred to the scheme of forming replicas or the carbon copies of the data items of the nearly present useful data items on the neighbour nodes. The replication was done to reduce the overhead involved in communication between the nodes present at the farther distance from the source nodes and also to later avoid unavailability from their current position in case of the On-demand retrieval [25]. Whereas caching on the other part involves the strategy of storing or maintaining record of the previous data items by saving them or when the data items requested by the nodes comes to the nodes itself it saves the path only. The difference between replication scheme and caching lied in the way the data items were dealt. Thus the main point of difference between them was that caching techniques usually act after the data items were retrieved whereas the data item was replicated even before the request for that data item was received on the requested node.

The replication scheme was employed with the common decision of all the members present in the large group. The common decision in the group was of nodes, that the individual node helped in deciding over the matter to form replicas of the original data item. Thus this scheme involved the large set of nodes. Generally the replication technique depended upon the statistics of accessing the data items either of node statistics or the other statistics.

There were some of the properties that supposed to be critical for both the replication and caching techniques. On the basis of these properties the data replication and caching techniques reviewed were compared in Table 2.1. These common properties were listed below as:

- Centralized structure: This property related that there was need for the source or the central server for handling each and every request.
- Consistency mechanism: This property related that in the system the technique was having the updated or without updated propagation consistency control mechanism.
- Cluster based hierarchy: This property issues over the scheme that does it required the cluster dependant hierarchy.

- Dependency upon routing protocol: This property described that the technique required the underlying routing protocol for its working or not.
- Lesser memory space: This property related to the facts that in particular technique for the purpose of replication there exists some type of memory constraints or not.
- Fairness: This property described whether the technique aims at improving the load balancing or the fairness feature among the moving nodes in the network.
- Scalability: This factor described that whether the technique designed was suitable to be employed in case of the scalable networks or not.

**Table 2.1:** Comparison between various properties of caching and replication

	SAF,DAF, DCG	DREAM	CLEAR	Expanding Ring	Chen <i>et al.</i>	DHTR	Yin <i>et al.</i>	CCS	Caching for benefit
Centralized structure	No	No	No	Yes	No	No	No	No	No
Consistency mechanism	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
Cluster based	No	No	Yes	No	Yes	Yes	No	No	No
Routing protocol	No	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
Lesser memory	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
Fairness	No	No	Yes	No	Yes	No	No	No	No
Scalability	No	No	No	No	No	Yes	No	No	

## 2.4 Algorithms for Connected Dominating Set (CDS) formation

CDS was the problem in graph theory which was used for forming the virtual backbone structure in MANETs. There was dominating set present in the graph consisting of dominator nodes. The nodes present in the dominating set were from those nodes present in the graph but all the other remaining nodes were such that they were either the one hop neighbours of one of the node in the dominating set or the adjacent nodes of the nodes in the dominating set. The performance of CDS algorithm could depend upon the approximation factor that could guarantee that the formed set was the optimal set and must not exceed that factor. Time and message complexity was considered the cost involved in formation of the CDS algorithm [26]. There were various algorithms for constructing the CDS. In this section, various algorithms for CDS formation in ad hoc networks were reviewed.

Chen *et al.* [27] proposed that the large dominating set could be expected using the strict connectivity rule even than the CDS which basically gave the obvious structure for forwarding the messages. Thus in their algorithm they did not considered this feature and formed it as a new system which was named as Weekly Connected Dominating Set (WCDS). As in the CDS dominator nodes were connected but here this assumption was relaxed and the direct connection between them was not required further. Because of this reason only the size of WCDS was smaller. Moreover, as a result of this even though the network did not remained complicated anymore but there come the problem while taking routing decisions. One of the main disadvantages of this algorithm was that the neighbours exchange messages while changing their state [28] due to which the overhead created in this algorithm becomes much larger.

Stojmenovic *et al.* [29] had given various algorithms for CDS construction in MANETs. The CDS they had created, in its two kinds of nodes were present in the network mainly the CH and the Border Node (BN). CH in each group was used to form maximal independent set in MANETs of the given graph. BN was the node which was not the CH and in its two hops connectivity it must had atleast two CH present. CH was assigned order depending upon their position and rank, so as to fulfil the total ordering of all the nodes [29]. Rank of a CH was the total number of its 1-

hop neighbours. During the start of algorithm for selecting the CH firstly all nodes were colored white. Then correspondingly in a sequence the white node was colored red. The nodes colored red was that nodes which were ranked lowest among all the other nodes. After that in the next step all the nodes adjacent to red nodes that were white in color were colored yellow. At the last step in the algorithm each node was red or yellow and the red nodes turned to be the CH. In [26] and [30] two similar algorithms for CDS formation was formed that resulted in message optimality. It was claimed that the approximation factor for first was around 8 and for second one the approximation factor was constant and also the message and time complexity was linear.

The above mentioned both of the algorithms started with the formation of maximal independent set. The independent set formed in this ensured that the minimal distance between the complementary subsets was generally two hops only. The two were different in sense that in [26] rank of nodes were given by the leader electing strategy whereas in [30] there was no such strategy followed for ranking the nodes. After this complex analysis the construction phase was started. This was the complex CDS formation algorithm. The reason by which it was proved was that each node had to handle fifteen to twenty different types of variables. Also the nodes must be able to send seven to ten different types of messages for the formation of virtual backbone structure for the network.

Wu *et al.* [31] proposed CDS algorithm that was simple and worked very effectively. In this algorithm certain rules were followed while construction of the CDS algorithm. Firstly, the CDS was found in this and then the duplicate nodes were pruned to form the final structure applying some of the rules. At the starting point all the nodes were present in the network. Then the duplicate copies are pruned or removed in case id of dominator node is large than their id. Each node here was given the specific node id.

## **2.5 Congestion aware routing protocols**

In MANETs the nodes move randomly without depending upon any central authority for routing. The packets in these types of networks were broadcasted to the nodes for sending them to the desired destination. Suppose that in case the large number of

packets were flooded to the particular node. Then due to more packets coming at the node sometimes congestion at that node may occur or also due to failure of link to the particular node over which the packet was to be forwarded congestion could occur. Thus there must be congestion detection strategies employed that could help in detecting the congestion and in turn also helps to avoid that congestion.

Narasimhan *et al.* [32] used the weight value as the combined parameter and proposed the hop-by-hop congestion aware routing protocol. The other few metrics considered were delay, data transfer rate and link quality. The route chosen to forward the packets was such that it had minimum cost index among the other route to the same destination. The cost index depends upon the weight of the nodes over that path. The underlying protocol used was the multipath routing protocol that discovered various paths to the destination. Among the multiple paths the one with minimum cost index was chosen.

Fard *et al.* [33] proposed the algorithm for controlling congestion that occurred due to the network or link failure. It was such that the metric chosen for determining was the threshold value. There were two phases in this algorithm according to which the algorithm worked. In the first phase, losses that occurred as a result of network or link failure were determined. The amount of loss occurred was known by determining the queue usage which depends upon the relative 1-way trip time. In the second phase usually the comparison between the relative trip time of the new route and of the old one was done by using the current values like relative trip time and hop count present in the transport layer. Then correspondingly the value of relative trip time was adjusted.

Yi *et al.* [34] proposed the optimized multipath optimized link state routing protocol. The multiple routes to the destination were created using the dijkstra algorithm for the multiple paths. The different parameters used in this protocol were basically the cost function and the link stability. These parameters were used as they provide greater flexibility and extensibility to attain the usefulness of the given metrics for designing the protocol. The quality of service was also taken care of while developing this protocol as occurrence of loop was firstly detected. Also the recovery of routes was done in case they fail due to some reason. This protocol worked in a

better way even than the Optimized Link State Routing (OLSR) protocol. In case of large network traffic and high mobility scenarios this protocol was employed to give better performance.

Liu *et al.* [35] proposed the delay aware multipath routing protocol. This protocol was used to fulfil the requirements of end-to-end delay. The protocol performed its working in the two phases. In the first phase the delay of accumulating the data packets was taken as the admission metric to found the path between the source and destination nodes. In the second phase to estimate the delay and to found the best routing path the node delay was taken as parameter basically. The parameters chosen take into account the number of forwarding nodes nearby nodes and the amount of time for which the channel was busy and also the amount of packets to be forwarded present in the sending buffer.

## Chapter 3

### System Model and Problem Formulation

---

---

The system is assumed to be a medium to large scale mobile ad hoc network. Every node in the assumed environment has been assigned a unique host id. The specific or unique id has been assigned to each node so that it can be distinguished from the other nodes present in the network. Here the set of nodes present in the network is defined by  $N = \{N_1, N_2, N_3, \dots, N_P\}$ , where  $P$  is the total number of nodes in the network. Every node  $N_i$  where  $\{i = 1, 2, \dots, P\}$  can hold initially its own data item  $d_i$ . The set of all data items are denoted by  $D = \{d_1, d_2, d_3, \dots, d_P\}$ .

The assumptions are made for making the network simpler, such that all the data items taken should be of the similar size and weight. Such data items are taken so that all the data items can be put in the single packet on the network layer. Thus for this very purpose the size of the data items are kept small. So their size should be lesser than one transmission unit, the maximum size of packet which can be sent over the path on the network layer.

Capacity of each and every node is kept equal and of fixed space. It is assumed that each node  $N_i$ , can store the replica of any data item but only of those present in the set  $D$ . The replicas of data items are stored only if there is memory space and capacity present to store them. Each node  $N_i$  is having prior knowledge of the existing data set  $D$  and can attempt to have any data item  $d_i$  at any time. Moreover, it is also assumed that all nodes are having the same memory capacity. As each node is assumed to be similar therefore, even it cannot be considered that there can be some nodes with special software or hardware characteristics. The assumed model is such that in it every node takes participation and can share the data items with the other nodes without displaying any kind of partiality.

In the network assumed two kinds of nodes are considered depending upon the type of service given to each. Both the type of nodes performs their services purely in a dynamic environment. Here each mobile node sends the request for the item. In case the requested node is having the data item itself or the replica of that item then it is

consider that the request has been completed. Also it is assumed while constructing the network model that each data item is such that it cannot be changed and for every single the data of that node is replicated only if required.

In this chapter, the problem defined is that how to overcome the fault which occurs in the network due to congestion. Usually fault occurs in the system due link or network failure and also due to node failure. If congestion occurs there may also be the chance of link failure and node as it is not able to forward packets. In that case choose another path having replicated data items over the nodes on that path. In larger networks usually the problem of data availability occurs. So, one solution to this problem is replication of the main data items over the best hosts in the network. The protocol is such that it provides solution for fault tolerance by replication in MANETs.

The problem formulation therefore involves firstly the creation of virtual backbone structure. So that number of nodes over which the process is to be employed decreases. Next the way of locating data items is being described. Then comes the problem when more packets are transmitted towards the backbone node due to which congestion may occur. So some technique is to be employed to detect the congestion. Thereafter, to control the congestion some method is to be proposed such that the alternate path can be chosen to transmit the request packets. The path having more number of hops to reach destination can even be taken. Actually the replica allocation decision is based on the parameters such as access frequencies for the particular data item and number of hops it is far away from the source node. So when the alternate route is chosen and the request for the data item is sent then it is seen that its cost function is large. Then that particular data item is being replicated over the nearest backbone node. Thus in this way the data becomes available in the network without involving much overhead.

In the previous works the replication of data items are done for improving data availability considering various factors in the network. But it is worthwhile to answer this question that in this defined work the problem considered is of increasing the data availability in terms of fault tolerance. It is likely in case the congestion occurs data availability decreases. So for that purpose firstly it is determined that whether the

congestion has occurred and then to provide the particular data item the same are replicated over the nearby nodes.

## **Chapter 4**

### **Implementation**

---

---

In this chapter, the details of SCAR framework are described. The proposed solution provides better data availability by tolerating fault and reduces the message overhead created in the network having large number of nodes. It also helps in congestion detection and to resolve the same by moving over to the alternate path having node over which the replicated data can be found for the request packets sent.

These properties can be employed with the idea of forming virtual and dynamic backbone structure which helps to lessen the number of nodes involved in locating the data item in the network. The idea of formation of virtual backbone can be supported depending upon one of the problem in graph theory, the minimum connected dominating set problem. In this framework it is easy to locate data by sending request randomly to any of the backbone node only and receiving data from there. It also helps in case congestion occurs over the shortest path by choosing alternate path to destination. On the alternate backbone node path it is determined that the data item is far and cost functions involved is also more and see various other parameters. Then data item is replicated over that node itself.

It actually construct distributed replication scheme over the data locating step to increase the data availability. It does not uses any predefined protocol specific control packets, which means it perform the whole procedure passively. Thus it can help to reduce the problem related to congestion and control the overhead that can be caused. The basic parameters involved in making the decision whether received data item is to be replicated or not are distance of that data item from the requester node, battery power consumed in reaching that node and the frequency of accesses made to that particular data item. Usually the criterion is that the nodes replicate the data items that are distant from them and have high request frequencies.

#### **4.1 Virtual backbone formation**

The way in which the virtual backbone support is created is discussed here. The Minimum Connected Dominating Set (MCDS) is problem in graph theory that can be

employed to form virtual backbone for the assumed graph of network topology. MCDS is the induced subgraph of the Dominating set (D). The dominating set is the subset of vertices of the graph  $G(V, E)$  having vertices that belongs to  $V$  in  $G$  or the vertices  $v \in V$  are adjacent to vertices present in  $D$ . The MCDS of graph formed in case of MANETs helps in reducing overhead created by packets as the space to search data is reduced. But the formation of MCDS over a graph is NP-complete problem as approximations are used in practice [36]. The simple algorithm to form MCDS has been proposed in [31]. Here the MCDS formation has been inspired by this related work.

In the process of formation of MCDS, firstly nodes exchange their neighbour lists among other nodes. Then in next phase the nodes are marked true or false. In the starting all nodes are marked false. If a node is having in its lists atleast two non-adjacent nodes then only that node is marked true. If nodes are marked true it means they are dominator or members of MCDS. Next is the pruning phase, in which some nodes drop themselves from initial MCDS. In pruning phase conditions are applied and the nodes are removed which are out of the required scenario. Usually the set is formed having nodes that are marked true. To explain the criteria let us suppose that there are two nodes are in that set i.e.  $x$  and  $y$ . The condition is that if  $x$  and  $y$  are connected with each other with id value of  $x$  lower than that of  $y$ , any neighbour of  $x$  is also neighbour of  $y$ . Then pruning of  $x$  is done because  $y$  will cover the node  $x$  then.

## **4.2 Process of locating data items**

When the virtual backbone structure is formed afterwards the nodes send requests to the owner node for data items on the backbone nodes. The way in which the requests are sent and data is received is described here. It is considered that during the process of backbone formation, two types of nodes are created in the system, the backbone nodes and the initial nodes. In this framework the request packets are not broadcasted rather the requests are sent only to the backbone nodes or the dominator nodes present in the network. The dominator node is one to which every other node present in the network is attached directly. The initial nodes can only send their request for the data to the backbone nodes present in their vicinity. The process of locating the data items in the virtual backbone structure is such that when backbone node receive request for

data item in case it is having that data item, it will directly send it to the requester node and if it is not having that item then it will forward the request to its nearest backbone nodes. In the process of receiving data it is seen that if the data item received in backbone node is requested by the requester node, it will transmit the same to it. The backbone node will check its hash table entries to check the unique parameters of the received data with the requester node id.

### **4.3 Congestion detection**

During the data locating process the requests are sent to the backbone nodes and in case congestion occurs congestion detection strategy are employed at intermediate nodes to detect congestion. This strategy is buffer based. The weight is assigned to the current length of the buffer which is represented by the Weight parameter ( $W$ ). Each intermediate node on receiving the data request packets checks their buffer length ( $C_{bl}$ ). Then each node evaluates the Average Value ( $AV_{bl}$ ) with the help of applying exponential weighted moving formula [37]. Here the threshold value is specified to be near to 90 percent. In case this average value comes to be more than the predefined threshold value then it means that congestion has occurred.

### **4.4 On-demand replication**

In case congestion is detected on the shortest path then the request is made on the other backbone node through which the distance to the owner node is more. To overcome this, replication scheme is applied to replicate distant data on the backbone node itself. So even if the shortest path to destination fails, the other route can be taken over which the probability of getting the data item even closer becomes more. It is also discussed that under which circumstances the replication can be done. It is the On-demand replication process as decision for replication is made on receiving data item [38]. Three parameters are taken over which the replication decision depends: the number of nodes present between the requester node and owner node of data item, the battery power consumed in reaching the owner node and lastly the request frequency that how frequently data item is requested within a specified period of time. The replication decision depends commutatively on cost function given in equation 4.1 which is dependent upon the above described three parameters. The cost function

at node  $n_y$  for data item  $d_x$  is given as:

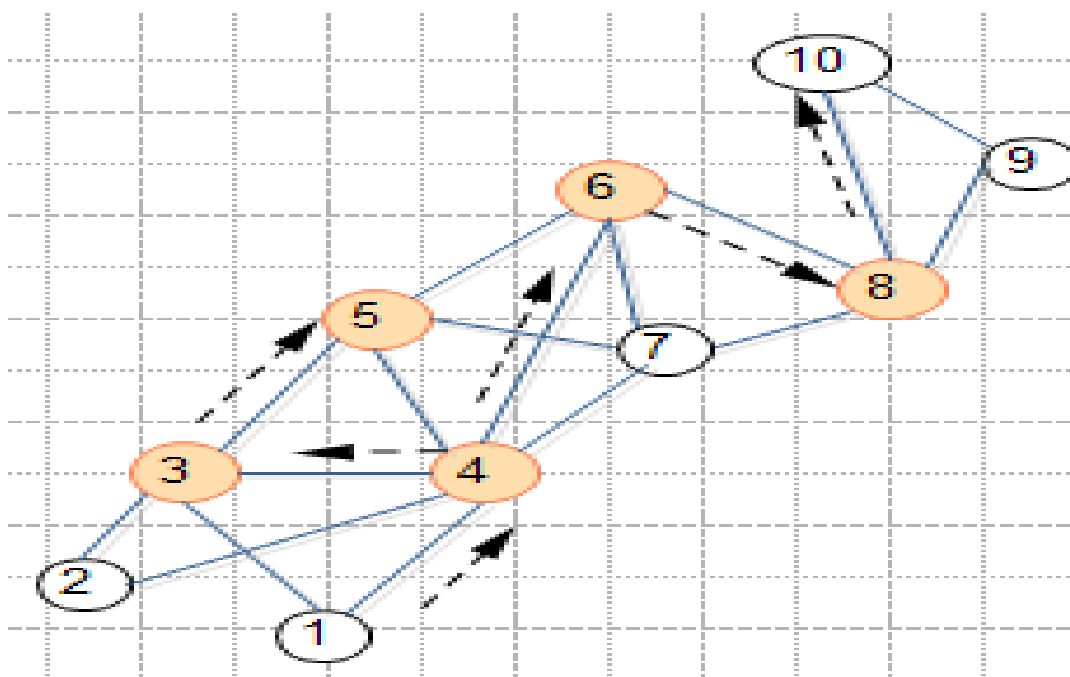
$$cost (f_x, n_{xy}, p_{xy}) = f_x * n_{xy} * p_{xy} \dots\dots\dots (4.1)$$

Where  $f_x$  is the history of request frequencies for  $d_x$ ,  $n_{xy}$  is the number of nodes between node  $n_y$  and the node from which  $d_x$  is received.  $p_{xy}$  is the amount of battery power consumed. Usually this function gives higher cost to the data items requested more frequently and in reaching which more power is consumed and also are far from the requester node. The decision for replication will be made based on this cost function. If at a particular node the cost of received data item is more than the lowest cost item in that node's cache. Then the received data item is replicated and the lowest cost item present firstly is replaced with it. This way the replica of the data item at distant comes on the closer node to requester node and probability of getting that item soon also increases.

#### **4.5 Scenario in case congestion occurs**

The working of the framework designed is explained with the help of scenario generated in the Figure 4.1. The phases explained above describes the process of formation of virtual backbone structure and routing of request packets to randomly chosen node having shortest path to the owner node of the data item. On it if congestion occurs then choose the alternate path by sending request to another backbone node. On the other backbone node it is determined that data item is far and various other parameters are seen and as a result the replication decision is made to replicate the data on that node itself. In this way the next time data is requested from that particular will be available easily and at shorter distance. The highlighted nodes in Figure 4.1 represent the backbone nodes while other ones are the initial nodes. The scenario discusses the case in which node 1 requests data item at node 10. Now node 1 see its neighbouring list and choose randomly any backbone node, say node 4 here to send request on it. The node 4 now checks its 2-hop connectivity and finds that the data item 10 is not present near it, so it sends the request to farther nodes 3 and 6. Now as node 6 find node 10 in vicinity of its neighbour node 9 so it sends request to it this sends it further to node 10. On receiving request, node 10 packs data and sends it back on same path to node 1.

The replication of data items is done in this as when data is sent back to node 1, the backbone node 4 in the path on receiving data item compares the cost function of the received data item with that of the lowest cost data item say at that time it is data of node 5 with that of node 10. If the cost function of node 10 comes to be more than that of node 5 the data item 10 is replicated over node 4 itself. So next time when request for data item of node 10 comes at 4 it will receive the data from neighbour node only.



**Figure 4.1:** Scenario in case congestion occurs

Now in case congestion occurs on the shortest path to the destination node choose an alternate path. Consider that route from node 4 is the shortest route and if on node 4 congestion occurs. Then the packets will be sent to other nodes, like here they are sent to the backbone node 3. Next the similar data items locating phase is employed to send the requests to node 10 and then correspondingly data items are received at node 3. Then at node 3 the decision for data replication is made and the data item at node 10 is replicated at node 3. It can be seen now that next time when the requests are sent to get data item at node 10 over the shortest path, here which is through node 4 but

congestion occurs at that node. Then the alternate path having backbone node is followed. So then the request will be sent at node 3. Now if on node 3 the replica of data item of node 10 is present the data will be become available on closer node even though that route is not the shortest route.

#### 4.6 SCAR Algorithm

**Input:**  $G(V, E)$ , a simple unit-disk graph

**Output:** Congestion free graph with no data availability problem.

- 1:  $x \in V$ ,  $N(x)$  is neighbour set of  $x$
- 2: *if (any node has atleast two non-adjacent neighbours in its neighbour list) then*
- 3:     mark that node true
- 4: *else*
- 5:     leave marking that node left with false
- 6: *endif*
- 7: Put nodes marked true in set  $V_1$  and apply pruning to  $V_1$
- 8: For each  $x, y \in V_1$
- 9: *if ( $x, y$  are neighbours and  $id(x) < id(y)$ ) then*
- 10:      $x$  drops itself such that  $MCDS = (V_1 - x)$
- 11: *else*
- 12:     graph remains same
- 13: *endif*
- 14: Send data request on nodes left after pruning
- 15:  $cost(f_x, n_{xy}, p_{xy}) = f_x * n_{xy} * p_{xy}$
- 16: *if (cost of received data item > lowest cost item present on node) then*
- 17:     replicate received data on present node
- 18: *else*
- 19:     send data to requester node
- 20: *endif*

21: *Compute*  $AV_{bl} = (1-W) * AV_{bl} + W * C_{bl}$

22: *if* ( $AV_{bl} > \text{threshold value}$ ) *then*

23: Congestion detected as a result fault comes

24: *else*

25: no congestion occurs on backbone node

26: *if* (*congestion occurs and fault comes on the backbone node*) *then*

27: go to the other backbone node

28: *if* (*replica present*) *then*

29: data available on closer node and fault gets tolerated

30: *else*

31: route the request for data to destination node

32: *endif*

33: *else*

34: fault packets will be dropped

35: *endif*

36: *endif*

#### **4.7 Network Simulator (NS-2)**

In this section, NS-2 used for simulation is described in brief. In both the wired and wireless networks for knowing the dynamic features of the congestion and flow control mechanisms it has been supposed as the variant of the real network simulator [39]. NS-2 has developed around 1989 and is being supported by various societies and institutions to carry out their simulation works. NS-2 is used basically in network related research works and is an event driven simulator. With the help of it several protocols, algorithms, traffic sources, etc. can be implemented. NS-2 offers or also supports for the simulation of TCP, multipath protocols and routing over both the wireless and wired networks.

For the purpose of simulation NS-2 uses the programming language Object oriented Tool command language (OTcl) at the simulating layer for transcribing the

simulation scripts given by the users [39]. OTcl programming language is an objected oriented extended version of the Tcl Language only. The Tcl programming language is compatible with the C++ programming language. Thus NS is basically used for interpreting the user Tcl scripts.

During interpretation of the OTcl script two files are generated in the NS in which the analysis or results are present. These two files are the trace file and the nam file. In the nam file the visual animation of the whole simulation is being shown. On the other hand, in the trace file the output of all the objects created in the simulation is shown. Various packages like ns, nam, tcl and otcl etc. comes along with the NS project. The package is then named as all-in-one package. But these all utilities can be even separately downloaded and installed. In this work, the stable version 2.34 of ns all-in-one package is utilized which is installed in the Ubuntu environment. In this, the work is in the .tcl files which are to be written in the hex editor and then correspondingly the output is analysed in the .tr files in which the traces of the output values of each object are stored.

#### **4.8 Mobility model**

The mobility model is used for generating the detailed scenario of the motion of nodes in the area of simulation. In most of the simulations the Random Waypoint mobility model is employed. In this particular model the nodes move from one waypoint to the next with a randomly chosen speed. In each transition a specific speed and duration is chosen. After the stipulated transition time ends the node may stops for a certain time before starting its transition towards the next waypoint. The nodes in the simulation set up move randomly according to the model which is known as the random waypoint model that selects the rectangular field.

#### **4.9 Simulation Set up**

The network model is formed taking network having meso-scale (from 20 nodes) to large scale (up to 400 nodes) mobile nodes. In simulation the mobile nodes are identical and implement the User Datagram Protocol (UDP) and Internet Protocol (IP) as the transport and the network layer protocols. The packet losses at the network layer are ignored in this simulation setup. But on the other hand, the packet losses at

the media access control are still assumed. The packet size is taken here to be 512 byte. Each node is wireless communication card with the transmission range of 100 metres operating in the 2.4 GHz radio frequency band. In order to analyse the scalability property the number of nodes taken for simulation are varied from 20 to 400 nodes and the simulation time taken for which the nodes keep in motion is taken to be around 50 seconds. As the numbers of nodes are increased for determining the simulation considerably the area of simulation is also increased. At the same time the density of the whole areas is kept constant. The density of the area is kept constant so that the issues like contention and collision over the wireless sharing channel.

The simulation is set up over such an environment in which the nodes can move freely without any collision in a square type two-dimensional flat area. The motions of nodes are depended upon the mobility model described above in section 4.8. While simulation the nodes choose unusual speed upto 1 to 3 m/s. The simulation parameters taken are given in Table 4.1.

**Table 4.1:** Simulation Parameters

Interface	Wireless
MAC Protocol	IEEE 802.11b
Network Protocol	IP
Transport Protocol	UDP
Packet size	512 byte
Mobility Model	Random Waypoint
Simulation Area	1000 * 1000
Simulation Time	50 second
Number of Nodes	20 to 400
Node speed	1 to 3m/s
Transmission Range	100 m
Radio Frequency	2.4 GHz
Traffic Model	CBR
Protocol	AODV

## Chapter 5

### Results and Discussion

---

---

In this chapter, the performance of SCAR framework is analysed and using an underlying ad hoc routing protocol compared its data lookup mechanism. The underlying protocol used for the base-case data lookup simulations is AODV routing algorithm because of its on-demand property and popularity. AODV is on-demand protocol in which when a node need route to destination it initiate a route discovery process.

#### 5.1 Performance Metrics

For analysing the performance of SCAR framework simulations are done in large network, high density scenario and high load network environment. To analyse the performance of the framework, different parameters are defined which are as follows:

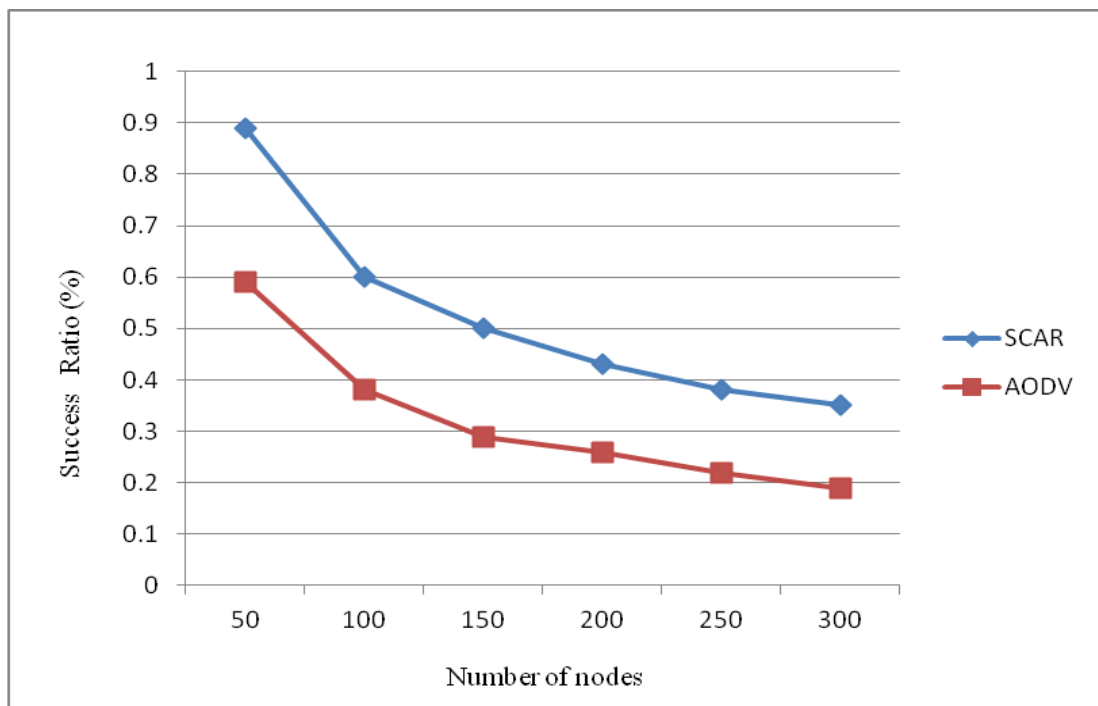
- **Data accessibility:** It is considered as an important parameter for data lookup and replication process also. It is generally described as the ratio between numbers of successful data accessibility requests made to the total access requests sent. The main criterion behind data replication scheme is to improve data accessibility in the network so that required data items can be reached. But as in MANETs the nodes moves randomly and changes the topology very frequently, it is difficult to get total data accessibility unlike as in the conventional static networks.
- **Rate of requests:** Rate of requests is defined as the number of total packets sent by the node. The packets sent include every packet sent from node, like the routing layer packet and the upper layer packet such as data and data requests packets. In this criterion the unicast and broadcast packets both are treated as single packet. Generally this parameter helps to calculate the message overhead involved in solving data lookup problem with or without replication.
- **Average query extent:** The average number of hops passed while finding the requested data item by the query is called as the average query extent. Basically, the query extent is the distance in terms of number of nodes from

requester to the requested data item on the successful route found. The SCAR decreases the average query extent by replicating the data items on the closer node so that the required data item can be found easily without traversing the large number of nodes. Thus in this way the average cost involved in finding the route to destination in the whole network will be expected to decrease as the value of average query extent decreases.

- Cache memory size: The memory size is defined as per the number of data items that a node can save in it. The performance of replication increases as the memory size is increased and also the large memory spaces allows to see the effect of replication better on data accessibility.

## 5.2 Results and Performance Analysis

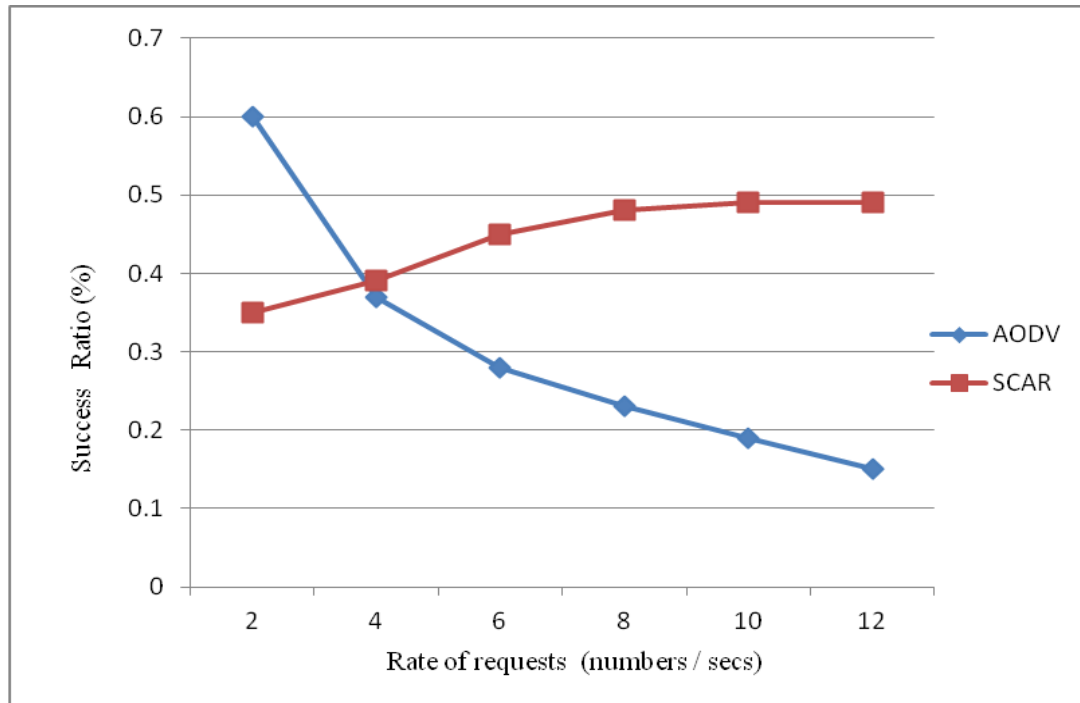
In Figure 5.1, the performance of defined metric data accessibility is analysed in terms of success ratio with respect to increase in number of nodes in the network. Other than number of nodes, rest all parameters for simulation are kept constant. It is shown that as the number of nodes in the network are increased the success ratio decreases.



**Figure 5.1:** Success Ratio comparison w.r.t number of nodes between SCAR and AODV

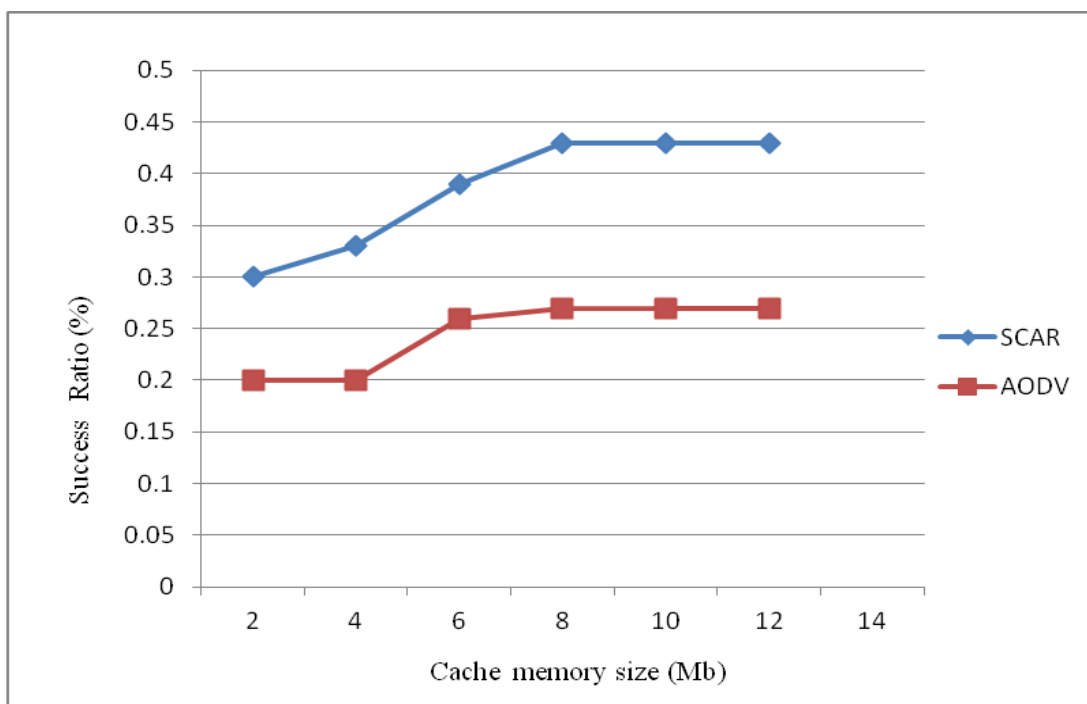
When number of nodes in network are increased, more number of nodes will be involved in transmission of data request due to which the message overhead is introduced. Moreover, the nodes are mobile due to which when nodes are increased the probability of disconnection increases. This will result in incomplete data request to system. But in case of SCAR lower message overhead is created as the request packets are sent only on the backbone nodes unlike AODV in which they are broadcasted and also disconnection does not occur. So, it is concluded that SCAR performs better than AODV in terms of data accessibility.

In Figure 5.2, the effects of varying rate of requests on the data accessibility are investigated. It is shown that AODV performs better when the rate of requests is lower. When the requests are low then the packets to be transmitted in the network are also lower due to which the packet loss and contention in the wireless channel is less. This is reason that in this case, AODV performs better. But as in the network the rate of requests increases, AODV cannot adapt itself to this increase. On the other hand, the SCAR remains at the same level of accessibility success ratio even if the rate of requests in the network increases.



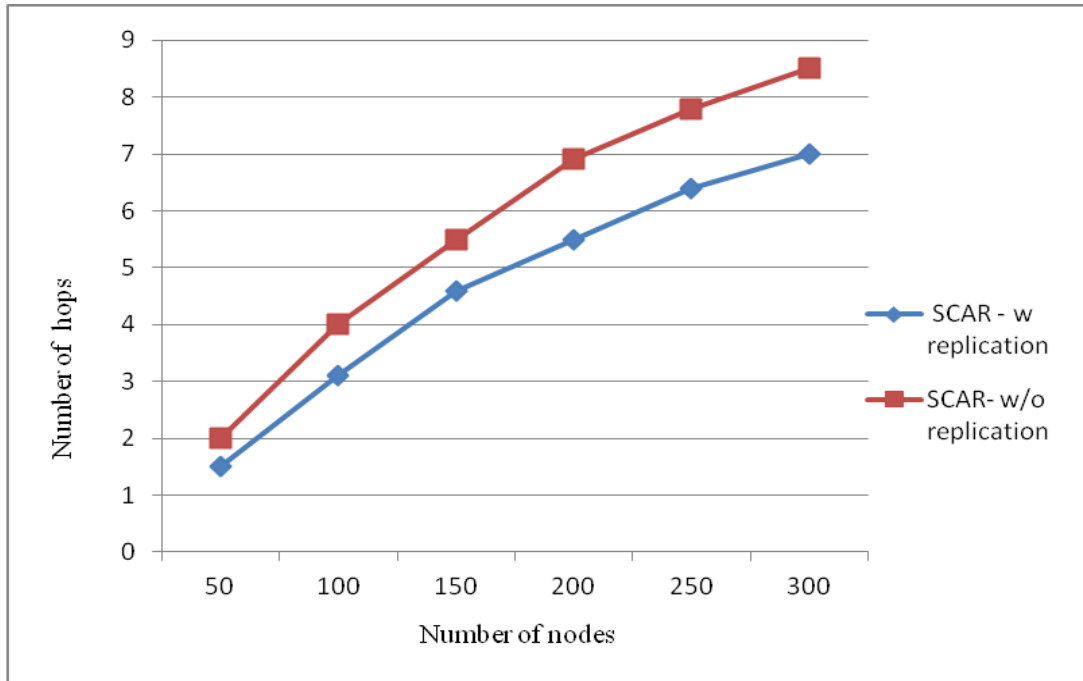
**Figure 5.2:** Success Ratio comparison w.r.t Rate of requests between SCAR and AODV

In Figure 5.3, the effect of increasing the cache memory size is depicted in terms of success ratio. The number of data items that can be saved in the cache of node can increase the performance of data replication and large memory sizes help to see the effect of replication on data accessibility better. It is shown in Figure. 5.3, that in case of AODV increasing the size of cache memory does not affect the success ratio in the simulation as the AODV does not replicate the data items. But when SCAR is used it is seen that the data accessibility increases as expected because it replicates the data items and increasing the memory size helps to increase the space for storing data items. This in turn increases the data accessibility.

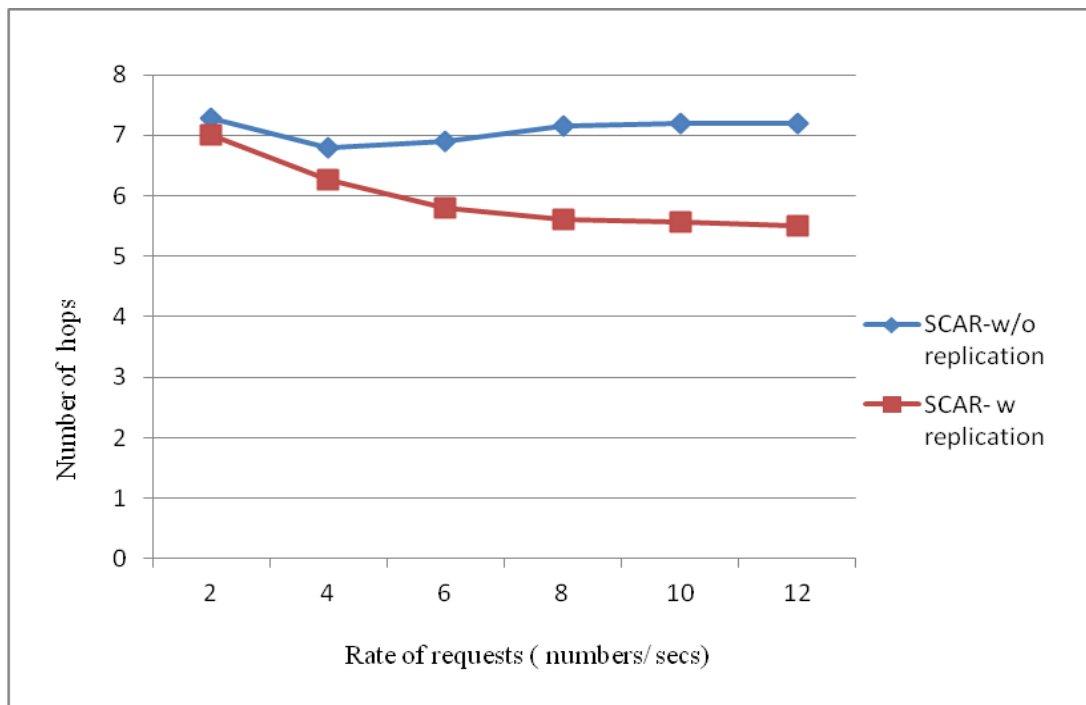


**Figure 5.3:** Success Ratio comparison w.r.t Cache memory size between SCAR and AODV

In Figure 5.4, it is shown that as the number of nodes in the network increases the average query extent increases in case of SCAR without replication. But for the SCAR with replication the results are better in terms of average query extent. This happens because even when the number of nodes increases in this case when the request is made for data item to the backbone node it can be got there only as the data item can be replicated over it. Thus the number of hops to be traversed in this case decreases.



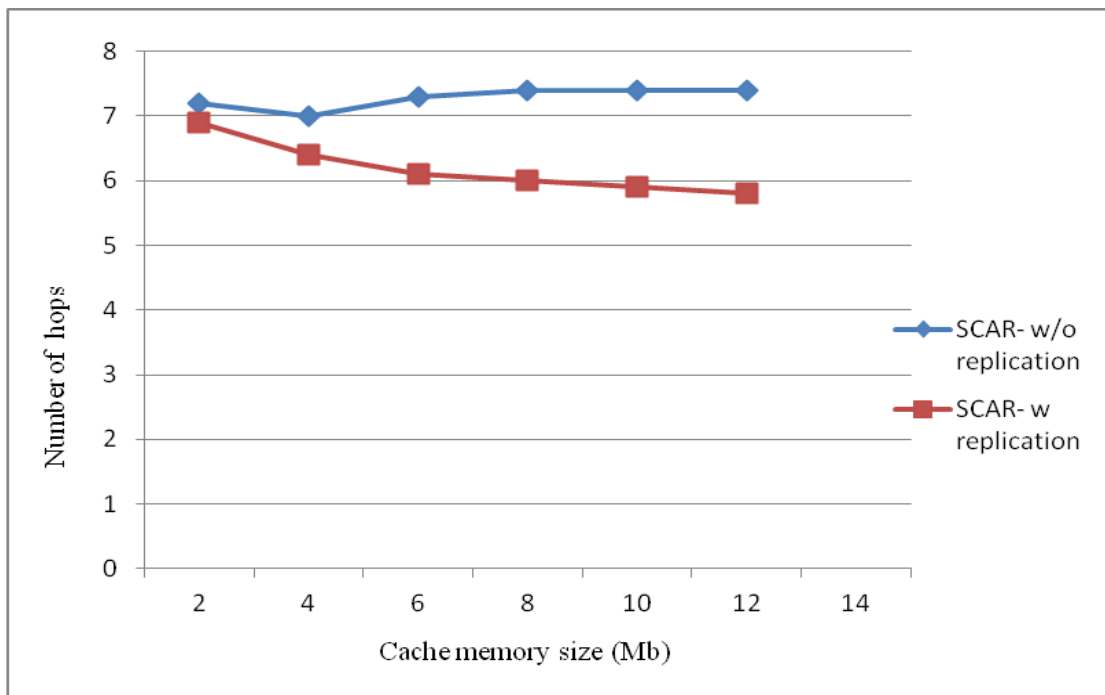
**Figure 5.4:** Number of hops traversed w.r.t number of nodes between SCAR-w and SCAR-w/o replication



**Figure 5.5:** Number of hops traversed w.r.t Rate of requests between SCAR-w and SCAR-w/o replication

In Figure 5.5, it is shown that for SCAR with replication average query extent decreases as the rate of requests is keep on increasing because the requested data items can be present on the closer nodes only so lesser hops have to be traversed. But in case of SCAR without replication the average query extent is kept stable with the rate of requests being increased.

In Figure 5.6, it is shown that as the cache memory size increases the query extent decreases in case of SCAR with replication. This happens because as the memory size increases the number of data items that a node can store increases which directly effects the data replication. So, lesser hops will be required to get the data item needed. On the other hand, in case of SCAR without replication it is seen that average query extent increases with the increase in cache memory size.



**Figure 5.6:** Number of hops traversed w.r.t Cache memory size between SCAR-w and SCAR-w/o replication

## Chapter 6

### Conclusion and Future Scope

---

#### 6.1 Conclusion

In this thesis, the SCAR framework have been discussed and analysed. This protocol helps to detect congestion because of which fault comes in the network. In case congestion occurs it helps to choose an alternate path over the virtual backbone structure. In this scheme the strategy of replication is deployed, so in case congestion is detected over the shortest routing path, it chooses an alternate path over which the request packets are sent for receiving the data item. But in case the owner node is far, the backbone node decides to replicate that item over itself. In this way, again this path is chosen in case when over shortest path congestion occurs, the data item becomes easily available and even closer to the requestor node. The performance of SCAR was compared with AODV for data locating process in MANETs. Also, the proposed approach gives replication decisions when a new data item is received by a backbone node rather than exposing any explicit control message to the system. The simulation results shown that the performance of SCAR framework with replication outperformed the other techniques in each of the performance metric defined. Thus it is concluded that backbone structure dependent SCAR framework along with the data locating process can keep the message overhead at very low level and increases the data availability.

#### 6.2 Future Scope

A future work that can be possible on this thesis is modeling of the network parameters for better performances of the framework. One of the main work that can be inhibited is to save the route to the forwarded data items on the backbone nodes itself so as to make data items available easily without creating much overhead. The details to this work will be considered as a future work.

## References

---

- [1] W. Stallings, "Wireless LAN Technology" Wireless Communication and Networks, Pearson Education, pp. 405-418, 2002.
- [2] E.M. Royer and C.K. Toh "A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks," IEEE Conference on Personal Communications, Vol. 6, no. 2, pp. 46-55, 1999.
- [3] I. Chlamtac, M. Conti and J. J. Liu, "Mobile ad hoc networking imperatives and challenges," Ad Hoc Networks, Vol. 1, no. 1, pp. 13–64, 2003.
- [4] C. Perkins, Ad Hoc Networking, Addition-Wesley, pp. 125-160, 2001.
- [5] M.R. Pearlman, "Determining the Optimal Configuration for the Zone Routing Protocol," IEEE Journal on Selected Areas in Communications, Vol. 17, no. 8, pp. 1395-1414, 1999.
- [6] A. Valera, W.K.G. Seah and S.V. Rao, "Cooperative packet caching and shortest multipath routing in mobile ad hoc networks," In Proceedings of the IEEE INFOCOM, Vol. 1, pp. 260-269, 2003.
- [7] M.K. Marina and S.R. Das, "On-demand multi path distance vector routing in ad hock networks," In Proceedings of the IEEE ICNP, pp. 14-23, 2001.
- [8] Y. Ko and N. Vaidya, "Location-Aided Routing in mobile ad hoc networks," Wireless Networks, Vol. 6, no. 4, pp. 307-321, 2000.
- [9] J. Li and P. Mohapatra, "Location aided knowledge extraction routing for mobile ad hoc networks," In Proceedings of the IEEE WCNC, Vol. 2, pp. 1180-1184, 2003.
- [10] G. Boato and F. Granelli, "A movement-based routing algorithm for vehicle ad hoc network," IEEE workshop on Automotive Networking and Applications, 2006.

- [11] V. Gianuzzi, "Data replication effectiveness in mobile ad-hoc networks," In Proceedings of the ACM International workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks, pp. 17–22, 2004.
- [12] P. Padmanabhan, L. Gruenwald, A. Vallur and M. Atiquzzaman, "A Survey Of Data Replication Techniques For Mobile Ad-hoc Network Databases," International Journal of Very Large Data Bases, Vol. 17, no. 5, pp. 1143-1164, 2008.
- [13] T. Hara and S. K. Madria, "Data replication for improving data accessibility in ad hoc networks," IEEE Transactions on Mobile Computing, Vol. 5, no. 11, pp. 1515-1532, 2006.
- [14] Y. Tseng, S. Ni, Y. Chen and J. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," Wireless Networks, Vol. 8, no. 2-3, pp. 153-167, 2002.
- [15] P. Pabmanabhan and L. Gruenwald, " DREAM: a data replication technique for real-time mobile ad-hoc network databases," In Proceedings of the International Conference on Data Engineering, pp. 134, 2006.
- [16] A. Mondal, S. Madria and M. Kitsuregawa, "CADRE: A collaborative replica allocation and deallocation approach for mobile-P2P networks," IEEE International Database Engineering and Applications Symposium, pp. 21-28, 2006.
- [17] A. Mondal, S. Madria and M. Kitsuregawa, "CLEAR: an efficient context and location-based dynamic replication scheme for mobile-p2p networks," In Proceedings of the International Conference on Database and Expert Systems Applications, pp. 399-408, 2006.
- [18] A.Mondal, S.K. Madria and M.Kitsuregawa, "An Economic Model for Efficient Dynamic Replication in Mobile-P2P networks", International Conference on Management of Data COMAD, 2006.
- [19] V. Thanedar, K.C. Almeroth and E. Belding-Royer, "A lightweight content replication scheme for mobile ad hoc environments," Networking

- Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications, pp. 125-136, 2004.
- [20] K. Chen, S.H. Shah, and K. Nahrstedt, "Cross-Layer Design for Data Accessibility in Mobile Ad Hoc Networks," *Wireless Personal Communications*, Vol. 21, no. 1, pp. 49-76, 2002.
- [21] H. Yu, P. Martin and H. Hassanein, "Cluster-based Replication for Large scale Mobile Ad-hoc Networks," *IEEE International Conference on Wireless Networks, Communications and Mobile Computing*, Vol. 1, pp. 552-557, 2005.
- [22] L. Yin and G. Cao, "Supporting cooperative caching in ad hoc networks," *IEEE Transactions on Mobile Computing*, Vol. 5, no. 1, pp. 77-89, 2006.
- [23] Y. Du and S.K.S Gupta, "A cooperative caching service in MANETs," *ICAS-ICNS Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services*, pp. 58-58, 2005.
- [24] B. Tang, H. Gupta and S.R. Das, "Benefit-based Data Caching in Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, Vol. 7, no. 3, pp. 289-304, 2008.
- [25] K. Constantinos, "Data Management in Mobile Environments".
- [26] P.J. Wan, K.M. Alzoubi and O. Frieder, "Distributed Construction of Connected Dominating Set in Wireless Ad Hoc Networks," *Journal of Mobile Networks and Applications-Discrete algorithms and methods for mobile computing and communications*, Vol. 9, no. 2, pp. 141-149, 2004.
- [27] Y. Chen and A. Liestman, "Approximating minimum size weakly-connected dominating sets for clustering mobile ad hoc networks," In *Proceedings of the ACM International symposium on Mobile ad hoc networking & computing*, pp.165-172, 2002.

- [28] J. Yu and P.H.J. Chong, "A survey of clustering schemes for mobile ad hoc networks," *IEEE Transactions on Communications Surveys & Tutorials*, Vol. 7, no. 1-4, pp. 32-48, 2005.
- [29] I. Stojmenovic, M. Seddigh and J. Zunic, "Dominating Sets and Neighbour Elimination-Based Broadcasting Algorithms in Wireless Networks," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 13, no. 1, pp. 14-25, 2002.
- [30] K.M. Alzoubi, P.J. Wan and O. Frieder, "Message-optimal connected dominating sets in mobile ad hoc networks," In *Proceedings of the ACM International symposium on Mobile ad hoc networking & computing*, pp. 157-164, 2002.
- [31] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks," In *Proceedings of the International workshop on Discrete algorithms and methods for mobile computing and communications*, pp. 7-14, 1999.
- [32] B. Narasimhan and S.S. Baboo, "A hop-by-hop congestion-aware routing protocol for heterogeneous mobile ad-hoc networks," *International Journal of Computer Science & Information Security*, 2009.
- [33] M.A.K. Fard, S. Karamizadeh and M. Aflaki, "Enhancing congestion control to address link failure loss over mobile ad-hoc network," *International Journal of Computer Networks & Communications*, pp. 177-192, 2011.
- [34] J. Yi, A. Adnane, S. David and B. Parrein, "Multipath optimized link state routing for mobile ad hoc networks," *Ad hoc Networks*, Vol. 9, no. 1, pp. 28-47, 2011.
- [35] S. Liu and J. Liu, "Delay-aware multipath source routing protocol to providing QoS support for wireless ad hoc networks," In *Proceedings of the IEEE International Conference on Communication Technology*, pp.1340-1343, 2010.

- [36] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, Vol. 20, no. 4, pp. 374-387, 1998.
- [37] G. Rajkumar and K. Duraiswamy, "A fault tolerant congestion aware routing protocol for mobile ad hoc networks," *Journal of Computer Science*, pp. 673-680, 2012.
- [38] E. Atsan and O. Öznur, "SCALAR: Scalable data lookup and replication protocol for mobile ad hoc networks," *Computer Networks*, Vol. 57, no. 17, pp. 3654-3672, 2013.
- [39] "Ns-2," [http://nnam.isi.edu/nsnam/index.php/Contributed\\_Code#Documentation](http://nnam.isi.edu/nsnam/index.php/Contributed_Code#Documentation), 2013.

## List of Publications

---

1. Navroop Kaur, Neeraj Kumar, “Fault Tolerance in Mobile Ad hoc Networks”, International Journal of Research in Advent Technology, Vol. 2, no. 2, pp. 233-239 , 2014.
2. Navroop Kaur, Neeraj Kumar, “SCARP: Scalable Congestion Aware Replication Protocol”, IEEE International Conference on Advanced Communications Control and Computing Technologies (ICACCCT), Syed Ammal Engineering College, Ramanathapuram, Tamilnadu, India, 2014. **[Accepted and Presented].**