Table of Contents

Chapter-1

VEHICULAR AD-HOC NETWORKS

1.1 Introduction

Vehicular Ad-Hoc Networking (VANETs) is a modern concept, emerging as new research area which integrates three prior Communication research areas namely Ad-Hoc networks, Wireless LAN, Cellular telephony. The main purpose of the VANETs is to improve the traffic safety and provides number of comfort application to traveling people. VANETs is different from other kind of Ad-Hoc networks by their hybrid network architectures, high speed node movement characteristics and wide range of new comfort application possibilities. Therefore VANETs poses a very unique networking research challenges and the design of routing protocols for VANETs becomes a very crucial and important issue.

VANETs enable new generation wireless capabilities to vehicles. The objective of VANETs is to provide two key features:

1. Efficient vehicle to vehicle communication for providing IST (Intelligent transport system) which includes Cooperative traffic monitoring, Control of traffic flows, Blind crossing, Prevention of collisions, real time rout computation.
2. Effective connectivity while on the road to high speed mobile users for providing various comfort like internet connectivity to download movies, play videos, and songs.

One of the earliest studies on VANETs was started by Association of electronic Technology for automobile traffic and driving of Japan in the early 1980s.In 2000, the European project car-TALK tried to find out the problems related to the safe and comfort driving based on inter vehicle communication. Since 2002, with the rapid development of Wireless Technology, VANETs emerged as a new research area. Various new workshops were organized to address research issues in this emerging area like ACM international workshop on Vehicular Ad-Hoc

Networks from 2004 and International workshop on Intelligent Transportation from 2003.

On the other hand, several major automobile manufacturers have started investing in this research field. Audi, BMW, Daimler Chrysler, Fiat, Renault and Volkswagen have united to create a non -profit organization called Car2Car Communication Consortium (C2CCC) with the objective of further increasing the road safety and efficiency by means of inter vehicle communications. IEEE has also formed the new IEEE 802.11p task group which focuses on providing wireless access for the vehicle environment.

Due to high nodes mobility and unreliable channel conditions, VANETs has its unique characteristics which pose many challenging research problems, such as data dissemination, data sharing and security issues. Data routing in VANETs is a key networking problem because of high mobility of vehicles while most traditional sensor networks routing protocols focus on the stationary nodes. The main requirement of VANETs routing protocol are to achieve dynamic changing topology and minimal communication delay with minimum consumption of network resources. Here I have discussed most recent research progress of routing protocols and their pros and cons are compared with the key performance matrices [1].

1.2 Network Architectures of VANETs:

Network architecture of VANETs falls into three categories:

1.2.1 Pure Cellular/WLAN:

VANETs uses fixed cellular gateways or WLAN access point at traffic intersection or besides road. Access point also plays the role of sinks for sensed information. Vehicles moves and connect to sinks when they enter into sink’s signal transmission range. Moving nodes didn’t have gateways capabilities [1].

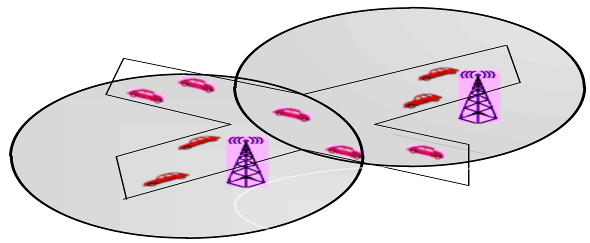


Figure 1: Pure Cellular Architecture of VANETs

1.2.2 Pure Ad-hoc:

All vehicles and road side wireless devices can form a mobile ad-hoc network to perform vehicle-to-vehicle communications and achieve certain goals such as blind crossing. In this category vehicles can work as gateways also [1].

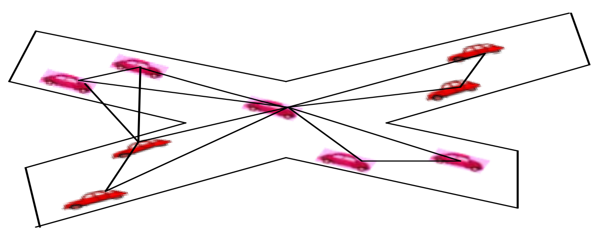


Figure 2*: Pure Ad-hoc Architecture of VANETs*

1.2.3 Hybrid:

It combines cellular, WLAN and Ad-Hoc networks together. It uses some vehicles with both WLAN and cellular capabilities as the gateways and mobile network routers so that vehicles with only WLAN capabilities can communicate with them through multi hop links, to remain connected to the world [2].

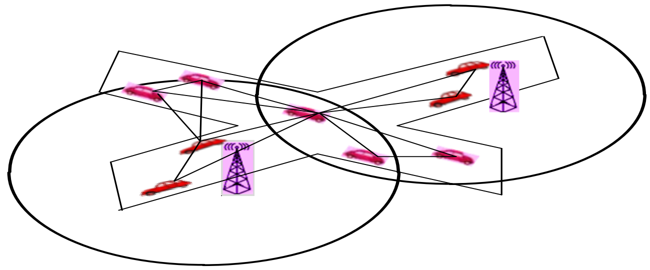


Figure 3*: Hybrid Architecture of VANETs*

1.3 Characteristics of VANETs:

VANETs can be distinguished from other kinds of Ad-Hoc networks by following unique properties [1]:

1.3.1 Highly Dynamic Topology:

Due to high speed movement of vehicles, the topology of VANETs frequently changes. For example, vehicles having a transmission range of 300m, in the worst case if two cars travelling at the speed 30 m/s then their links will be lost in every 10 seconds.

1.3.2 Frequently Disconnected networks:

In case of low vehicle density as in rural areas, VANETs have high probability of disconnection. In this environment, transmitting packets to neighbors or multi hop packets to sinks is a major problem due to lack of vehicular connectivity.

1.3.3 Sufficient Energy and Storage capability:

The nodes in VANETs have ample energy and computing power for processing and storage because nodes are supported by vehicles instead of hand held devices.

1.3.4 Different Environments and Application:

Nodes in VANETs have to be in various environments like highway scenario, urban areas, rural areas, battle fields. Number of protocols has been designed for these complicated sensing environments. At the same time these protocols should be flexible for the application related with VANETs such as traffic management, environment monitoring, vehicle tracking and traffic safety.

1.3.5 Mobility Prediction:

Because of high speed movement of vehicle nodes, the prediction of future position of node becomes very important in the design of VANETs routing protocols. This prediction can be made possible by GPS navigation system.

1.3.6 Geographical Communication:

In case of unicast or multicast, communication end points are defined as ID or group of IDs (IP address or group of IP address). In case of VANETs most of the application needs geographical based communications in which we need to communicate with all the nodes that belong to particular geographical area.

1.4 Issues in the design of routing protocols for VANETs:

The special and unique characteristics of VANETs raise various challenging issues. We need to consider the following during design of routing protocols [2].

1.4.1 Asymmetrical Transmission Range:

The transmission range of vehicle sensors and cellular gateways or WLAN access point is different. Vehicle sensors have smaller coverage than access point.

1.4.2 Hard Delay Constraints:

For some VANETs messages, high data rate is not required; instead real time processing is required. For example, in accident or break fail messages.

1.4.3 Reliability and Quality of Service:

For reliable delivery of data in VANETs, we need some acknowledgement. Sometimes delivery of time is also closely associated with data.

1.4.4 Flexibility:

Routing protocols should be flexible in different environments of VANETs and various kinds of applications which require different kinds of services.

1.4.5 Scalability and Fault Tolerance:

Generally VANETs are large scale networks facing unpredictable obstacles such as buildings and mountains. Therefore routing schemes should have low influence on increase of the network area and should be able to form new links and routes to destination in case of failure.

1.4.6 Low Cost:

Most of the current solutions of VANETs needs heavy infrastructure. Therefore proposed protocols should be economical as well as efficient.

1.4.7 Security:

Especially in commercial and military applications, security becomes an important issue. Therefore routing should be secure and message transfer should be authenticated.

Chapter-2

ROUTING PROTOCOLS IN VANETs

Routing in VANETs has been studied recently and many different protocols are proposed. We classify them into the following three categories:

1. Broadcast Routing
2. Geo-cast Routing
3. Unicast Routing

2.1 Broadcast Routing

Broadcasting is a basic scheme for routing in VANETs. Specially used for short distance scenarios such as a segment of road where the sensed data can be traffic jam video, weather, emergency, road condition, and delivering advertisements and announcements.

- **Flooding:** It broadcasts message to all of its neighbors except the one from where it has got its message. It performs well for limited number of sensors but with the increasing number of nodes performance drops quickly [3].

- **BROAD COMM**: Durresi et al. proposed emergency broadcast protocol for highways based on hierarchical structure. Highways are divided into cells where vehicles move along. Nodes are divided into two levels of hierarchy. The first level includes all the nodes. The second level is represented by cell reflectors. Cell reflectors are the nodes located at geographical center of cells. Cell reflectors work as base station and handles in/out emergency messages from neighboring cells [4].

- **Vector Based Tracking Detection (V-TRADE)/ History Enhanced-V-TRADE:** GPS based broadcast protocols. This scheme classifies the neighbors into different forwarding groups. For each group only a small subset of vehicles is selected to rebroadcast the message. V-TRADE uses vehicle movement and position vectors Whereas, HV-TRADE uses vehicle history position information for subset selection [5].

- **Urban Multi-hop broadcast (UBM)**: In UBM the sender node tries to select the furthest node in broadcast direction for forwarding and acknowledging packets without any prior topology information. It is specially designed to overcome interference, packet collision and hidden node problems. RTB, CTB and ACK are used for reliability [6].

2.2 Geo-cast Routing

It is basically location based multicast routing. The objective of geo-cast routing is to deliver packets from source node to all other nodes within a specified geographical region, also called Zone of Relevance (ZOR).

It is based on direct flooding which limits the message forwarding overhead and network congestion by defining a forwarding zone [1].

- **Linda bariesemeister et al's Protocol:** In this protocol when a node receives a packet, it does not forward it immediately but waits for some duration according to following formula.

$$WT(d) = -\frac{MaxWT}{Range}.\hat{d}+MaxWT$$

Where, *MaxWT* = Maximum Waiting Time

Range = Transmission range

d= Distance from sender

$\hat{d}$= min {d, range}

The waiting time is inversely proportional to the sender's distance from current node. When, this waiting time expires and the current node does not receive same message from other neighboring nodes then the current node rebroadcasts the message. It improves the system reliability but increases system delay [7].

- **Cached Geo-cast:** Maihofer and Eberhard's scheme add a small cache in routing layer which is used to hold those packets that can't be forwarded instantly due to local minimum. When a new neighbor comes into reach or known neighbors change their positions, the cached message is forwarded to newly discovered nodes. It always chooses greedy distance approach to forward packet towards destination. It is good at abating system latency but with the increase of network topology, data forwarding loop creates problems [8].

ALGORITHM-I

Forward Packet

(used for forwarding packets by node)

```
N Forward Packet (P)
// P.d.c…..packet's destination region center
// P.fwdRetries…packet's forward attempt at current node
// MaxFwdRetries…Maximum forward attempts defined
  1.   nh=FindNearestNeighbor(P.d.c); //Find neighbor closest to destination
  2.   if (nh!=NULL ) then //suitable neighbor found
  3.   send packet P tonh
  4.   else
  5.   if (P.fwdRetries<MaxFwdRetries) then
  6.   Delay P and Recall Forward Packet(p)
  7.   else
  8.        Put P into cache
```

- **Abiding Geo-cast:** In this protocol, packets need to deliver all nodes that are inside the geocast destination region during the geocast lifetime (a certain period of time) [9].

  It uses three approaches for forwarding messages: Server approach**,** election approach and neighbor approach.

2.3 Unicast Routing

It works efficiently for avoiding data collisions. It is divided into two categories.

1. Cluster based unicast.
2. Position based unicast

2.3.1 Cluster-Based Unicast Routing

A virtual network infrastructure is created through the clustering of node in order to provide scalability [1].

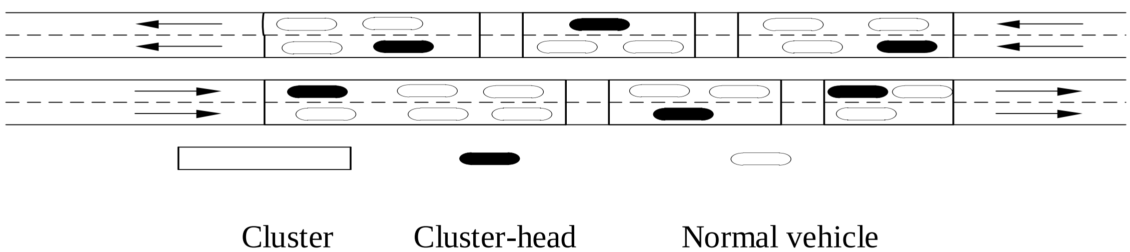


Figure 4 : *Cluster Based Unicast Routing in VANETs*

Each cluster has a cluster head which is responsible for intra & inter-cluster coordination. Inside nodes communicates though direct links and inter-cluster communication is done through cluster heads.

- **Clustering for open IVC Network Algorithm (COIN)**: In this protocol cluster election is based on vehicular dynamics and driver intentions. Here oscillatory nature of inter-vehicle distances is also considered [10].

- **Location based Relative Algorithm- Cluster Based Flooding (LORA-CBF)**: Each node can be the cluster head, gateway or cluster member. Greedy routing is used to forward packets. LREQ and LREP are used to find unknown routes [11].

2.3.2 Position-Based Unicast Routing

Node movement in VANETs is usually restricted in just bidirectional movements constrained along roads and streets. Therefore geographic routing based on position has been identified as a more promising routing technique for VANETs [1].

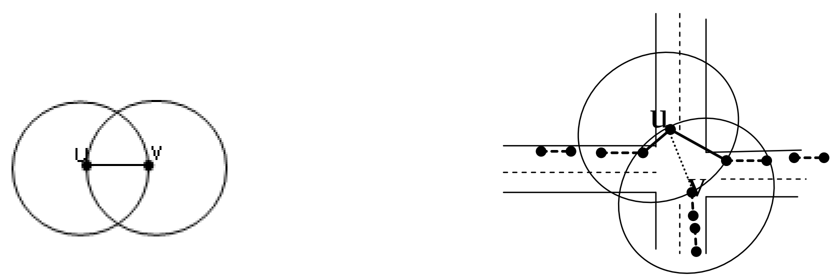
- **Greedy Routing (GR):** always forwards the packets to the node that is geographically closest to the destination.

- **Greedy Perimeter Stateless Routing (GPSR):** it combines greedy routing with face routing to get out of the local minimum where greedy routing fails [12].

Problems of GPSR:

- ➢ Direct communication between nodes may not exist due to obstacles such as buildings and trees.

- ➢ If we first apply the planar graph to build routing topology and then run greedy or face routing on it then routing performance will degrade because of longer path



(a) RNG **(b) Disconnection b/w UV**

Figure 5 *: Disconnection Problem in GPSR in VANETs*

Figure Explanation: Disconnected link between UV

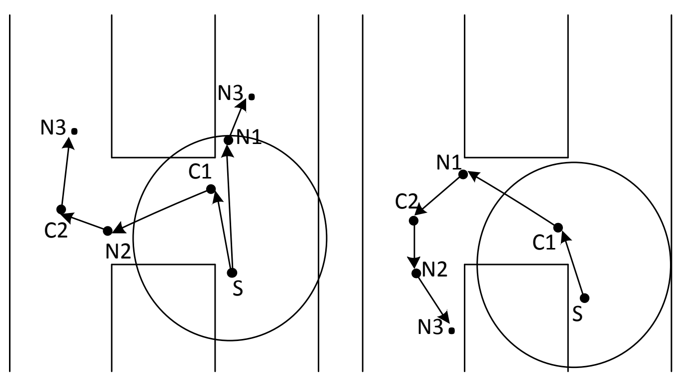
a. Relative neighborhood Graph (RNG) is planar topology used by GPSR. It tells that if the intersection of two circles centered at U&V with radius ||UV|| does not contain any other node then there will be a direct connection between U&V.

b. In GPSR, link UV is removed because node a & b exist inside the intersection area but because of obstacles like buildings, trees link ua and ub are not available. Thus U and V becomes disconnected which cause GPSR failure.

Various techniques have been proposed to deal with these challenges.

- **Greedy Perimeter Coordinator Routing (GPCR)**: It does not require street map information. It uses the fact that junctions are the only place where critical routing decisions are need to be taken [13].

  Therefore packet should always forward to a coordinator node on a junction rather than being forwarded across the junction.



(a) Restricted Greedy (b) Right-hand rule

Figure 6 *Restricted Greedy and Right-hand Rule of GPCR in VANETs*

To get out of local minimum problem GPCR uses repair strategy:

- Right hand rule is used to forward packet at junction by the coordinator.
- Apply greedy forwarding in between junctions

2.4 Comparison of Routing Protocols:

TABLE I

COMPARISION OF ROUTING PROTOCOLS

Routing Protocol	**Routing Type**	**Position information? (how to use)**	**Hierarchical Structure?**	**Simulation Scenario**	**Delay**
Flooding	Broadcast	No	No	------	Long
BROADCOMMM	Broadcast	Formation of cell	Yes	Highway Model	Short
V/HV-TRADE	Broadcast	Classify forwarding group	No	Simple Intersection	Short
L.Briesemeiser et al	Geocast	Packet forwarding	No	Highway model	Normal
Cached Geocast	Geocast	Packet forwarding	No	Quadratic Network Model	Short
Abiding	Geocast	Packet forwarding	No	--------	Short
COIN	Unicast	Cluster Formation	Yes	Highway	Short
LORA_CBF	Unicast	Packet forwarding and location prediction	Yes	Road Model	Short

Above table summarizes the characteristics of routing protocols of VANETs with following key properties like routing types, usage of position Information, hierarchical, simulation scenario, delay. In general, position based routing and geo-casting is more promising than other routing protocols for VANETs because of geographical constrains. However the performance of routing protocol depends heavily on mobility model, driving environment, vehicular density and many other facts. Therefore having the universal routing solution for all application of VANETs is extremely hard. In other words for specific VANETs application, we need specific routing protocol and mobility model to fulfill its requirements.

Security is also important for routing in VANETs, because many applications will affect life-or-death decisions and illicit tampering can have devastating consequences. This characteristic makes secure routing in VANETs more challenging and novel then it is in other communication network.

Chapter-3

GEOCASTING IN VANETs

3.1 Introduction

The first idea of geocast goes back to attempt to relate IP address to geographic locations in the UUMAP project. The project maintained a database in which geographic locations of internet host were stored. Later two similar projects tried to relate DNS names to geographic locations. They extended the DNS data structure with geographic longitude and latitude information, which makes it possible to return geographic location of a host based on IP address or DNS names. However, both these approaches were notable to support the reverse function, that is, they were not able to return the IP address or DNS name based on geographic information. Therefore, such systems made it possible to relate data flows with geographic areas, but they were unsuitable to direct data flow to a given geographic area.

Routing packets to a geographic destination location was first presented in Cartesian Routing. Cartesian Routing uses latitude-based and longitude-based addresses. Each network node, that is, source node, destination node, or intermediate node, knows its geographic address and the geographic addresses of its directly connected routers. Based on this information, geographic routing is possible where packets are forwarded to the neighboring node that is closer to the destination node than any other neighboring node or the forwarding node itself. If no neighboring node is closer to the destination than the forwarding node, the search space is enlarged by considering all nodes with n-hops distance to the forwarding node, using a flooding mechanism. Note that this approach is the basic algorithm for several later protocols. A restriction of Cartesian Routing is that only unicast is considered [14].

Current VANETs applications focus on three main areas: improving road safety (accident avoidance), enhancing driver convenience by providing alternative routes based on the road condition and finally in-vehicle VANETs applications like those for entertainment and inter-vehicle social chatting and cooperative gaming. Current road safety application focuses on warning systems for the purpose of road safety improvements [15]. A vehicle-to-vehicle live video streaming architecture is proposed in [16]. In VANETs, a vehicle can communicate with other vehicles directly through a wireless connection if they exist within its transmission range. In [17], the authors proposed a method to collect traffic information to estimate arrival time from destination using IEEE 802.11b based communication. It divides the road map into areas and each vehicle measures the time required to pass each area. Each vehicle periodically broadcasts their passage time to neighboring vehicles. The vehicles average area passage time is calculated by creating statistics data when number of area passage time records reaches a predetermined threshold. Another system for discovering and disseminating traffic congestion is proposed in [17]. Where vehicles build their own local traffic maps of speeds experienced on visited roads and shares this information with other vehicles. This allows a vehicle to build a map of expected speeds even on non-visited roads.

3.2 Geocast Routing Motivation

Emerging inter vehicles communication based on mobile networks has sparked considerable curiosity about the intelligent transport systems (ITSs). Wireless ad-hoc network communication plays a vital role in ITS. By using the global positioning system (GPS), VANETs have overcome the limitation of traditional systems, like radar and video cameras and make possible more advanced services in ITS.

Most of the ITS services require sending messages to all nodes in a certain geographical area, called geocasting, a subclass of multicasting. Unlike multicast, which sends a packet to arbitrary nodes, geocast enables transmission of a packet to all nodes within a pre-defined geographical region. The goal of geocasting is to guarantee delivery while maintaining a low cost [18].

In vehicular ad-hoc networks, the major challenge for routing protocol is to find a route from the sender to the destination without any preconfigured information and under constantly varying link circumstances. Topology based routing is strictly avoided because of dynamic changes in topology. The approach of position based routing relies only on geographic position information to deal with the problem of dynamic topology changes. This means, that all routing decisions, to which node packet should be forwarded, are based on the geographic destination data that is included in the packet. Position based routing is a suitable candidate for vehicular ad-hoc networks since position information is already available from navigation systems [19].

The motivation for using a geocast protocol is three folds. First, in vehicular ad-hoc networks we are only interested in communication between vehicles within the same geographical region in which the vehicle belongs. We refer this geographic region in which messages are transmitted to as the zone of relevance (ZOR). The boundaries of this area are determined by the GPS coordinates. This allows the information exchanged between vehicles to be flooded in a limited fashion. Second, geocast protocols have this nice property of using a multicast tree for broadcasting the requests from the vehicles in need of information, to the potential vehicle in the same ZOR. The multicast tree is used to route the responses back to the source using the same communication tree in a unicast way. Finally, geocast protocols being not necessarily a point-to-point protocols can be easily customized to piggyback application specific information such as those related to the road [20].

3.3 Different Techniques of Geocast Routing

There are three different techniques of geocast routing. Routing with simple flooding, Direct flooding and no flooding [21].

3.3.1 Routing with simple flooding

In this protocol, the source node delivers the information to all the nodes inside its transmission range. This protocol is directly not used in VANETs, but indirectly used for comparison of protocols. Performance of this protocol is very low because with the increase of number of nodes in the networks, load and looping of information becomes an unmanageable problem for the system.

3.3.2 Routing with directional flooding

In this protocol source node delivers the information to the forwarding zone nodes which are inside the transmission range. Here different types of forwarding zone can be used. There are two popular forwarding zones, Rectangular forwarding zone and distance based forwarding zone.

In the rectangular forwarding zone technique, source node sends the information to all nodes in its transmission range but only those nodes that belongs to the forwarding zone forwards the information to the next level. Nodes that do not belong to forwarding zone will simply discard the information. This is shown in following diagram very clearly.

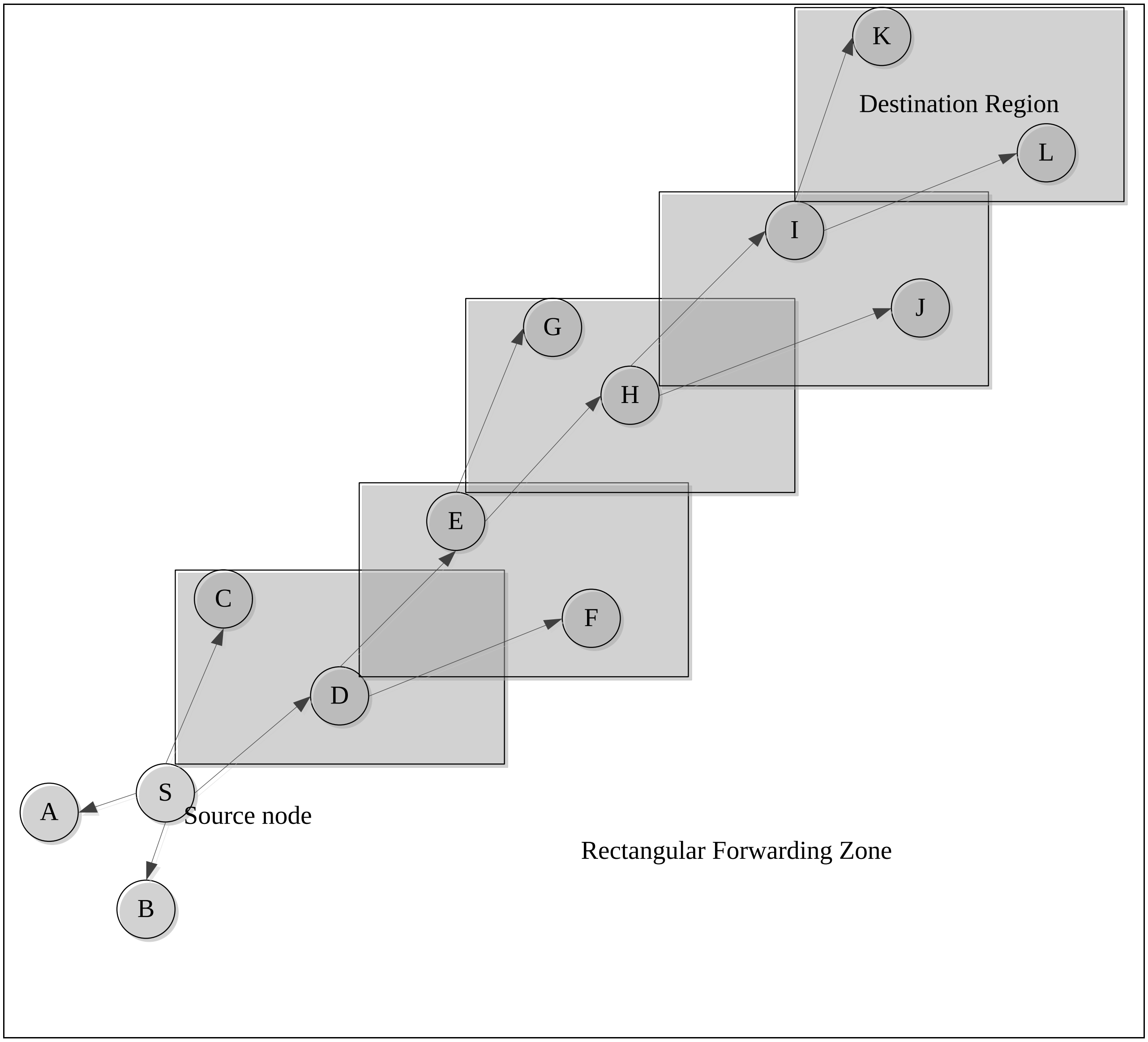


Figure 7: *Rectangular Forwarding Zone in VANETs*

In distance based forwarding technique, source node and each current sender, forwards information packet to the node which is nearest to destination region and inside its transmission range. At the end when a node sees that destination region is within my transmission range it forwards information to all the nodes of the destination region. This technique is also explained clearly in the following diagram.

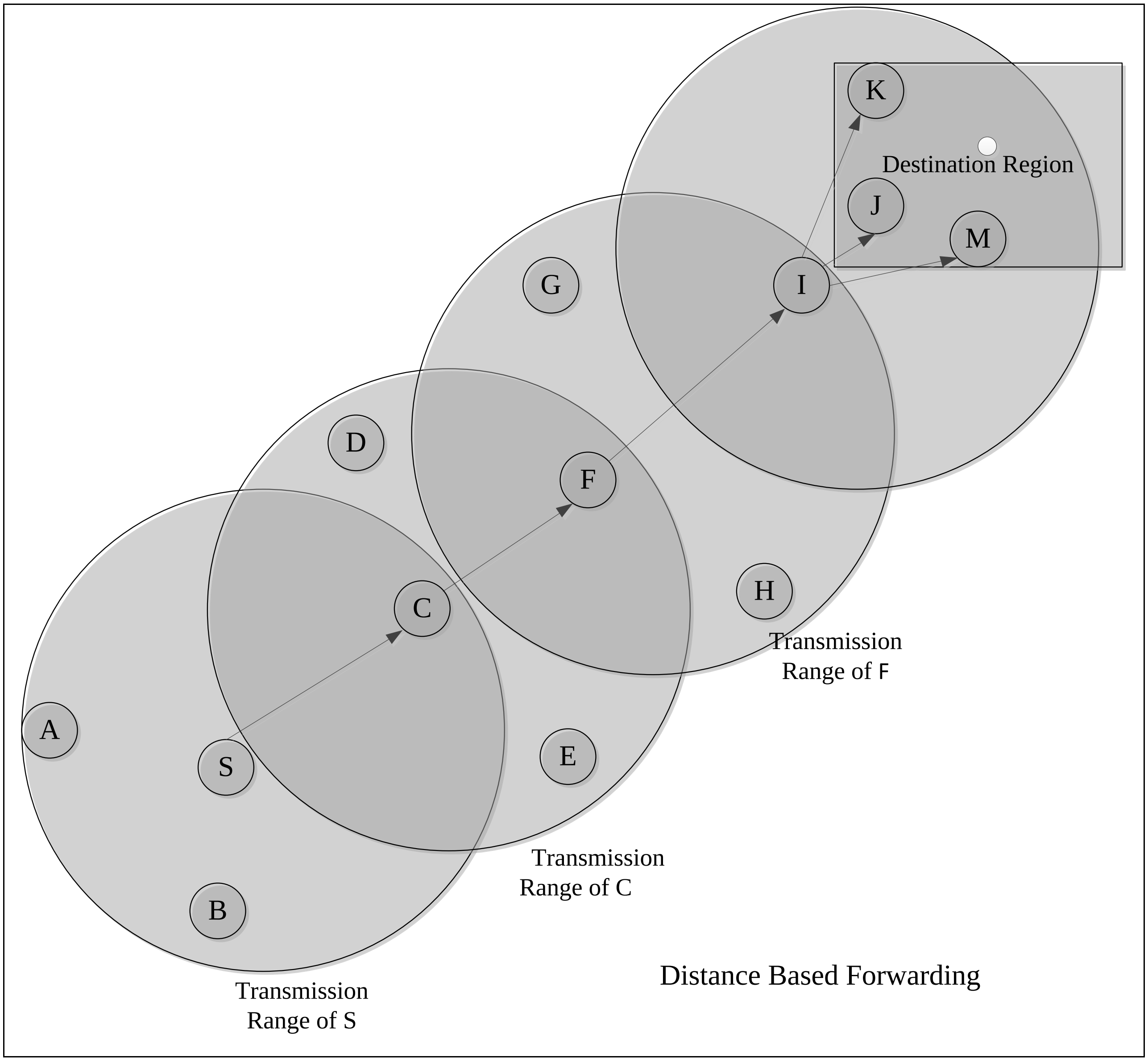


Figure 8: *Distance Based Forwarding in VANETs*

3.3.3 Routing without flooding

In this Technique there are three basic components Geo-Node, Geo-Router and Geo-Host. Each Geo-Router has circular range, which is covered by Geo-Node. Routers are connected with each other. Source node (any Geo-Host) sends information packet to its Geo-Node which automatically forwards it to its Geo-Router. Geo-Router finds an appropriate Geo-Router according to the destination region. The last Geo-Router forwards information packet to all Geo-Node that are covering the destination region's Geo-Hosts.

Chapter-4

RELATED WORK

4.1 Detection and dissemination of road traffic congestion

This protocol is an integrated solution having two main components:

- A flooding based geocast protocol for disseminating and exchanging road traffic information in a real time.
- A modified version Dijkstra algorithm which is used to dynamically recalculate the vehicle's route to a given destination by finding the least congested route

The operation of this protocol is based on simple request-response messages. The multicast tree is built by vehicles broadcasting the request to those vehicles within the transmission range. A request and response message is of the following form:

Req ID	Src ID	ZOR

Request

Resp ID	Src ID	Dest ID	Location	CongIndex

Response

Figure 9: *Request and Response Message format*

This request message is transmitted to the next vehicles in the transmission neighborhood. Upon receiving a request message the vehicles first check whether it has already received this message from different route and whether it's current GPS position is within the ZOR area indicated in the ZOR field of the message. The current message is ignored if the vehicle is outside ZOR or it has been already received. Otherwise the vehicle broadcasts the request message again to the next neighboring vehicle and sends a response message back to the source vehicle [20].

How it decides ZOR and Congestion Index:

4.1.1 Zone of Relevance:

The ZOR coordinates consists of the coordinates of the angle representing the geometrical region covering a given area. Typically geometrical region is of rectangle shape with each angle is identified by its GPS coordinates.

4.1.2 Congestion Index:

Congestion index is calculated by following formula:

$$CI = \frac{T - T_0}{T_0}$$

Where T is time actual travel time and T_0 is the ideal travel time when no congestion is experienced.

4.1.3 Least Congested route calculation:

It uses modified Dijkstra algorithm to dynamically calculate least congested route. It has modified Dijkstra Algorithm because of two reasons, first congestion indexes are dynamic. It changes continuously. Therefore while a vehicle is driving towards its destination, the least congested route may become congested one, which requires vehicles to regularly re-calculate their route. Second, a least congested route may not be the best one in terms of travel distance and cost. Therefore we design following factor based formula to calculate Dynamic distance or weight of the route.

$$D=\alpha.d + \beta.CI$$

Where α and β are the importance factor that the driver wants to give respectively to distance and congestion index. α and β are bound by the following relationship.

$$\alpha=1-\beta$$

Here β actually represents the point up to which driver cares about least connectedness. The vehicles use least congested route algorithm every time they approach towards a junction which may lead to more than one direction. Therefore the number of times the newer route is recalculated is bound by the number of junction a vehicle crosses. Assuming a road section between two connected junctions i and j, an $D_{i,j}$ can be calculated as:

$$D_{i,j} = \alpha . d_{i,j} + \beta . CI_{i,j}$$

Here, $D_{i,j}$ is the dynamic distance used in the modified Dijkstra Algorithm. $d_{i,j}$ is the actual distance and $CI_{i,j}$ is the congestion index between junction i and j. Following is the complete flowchart of the discussed technique.

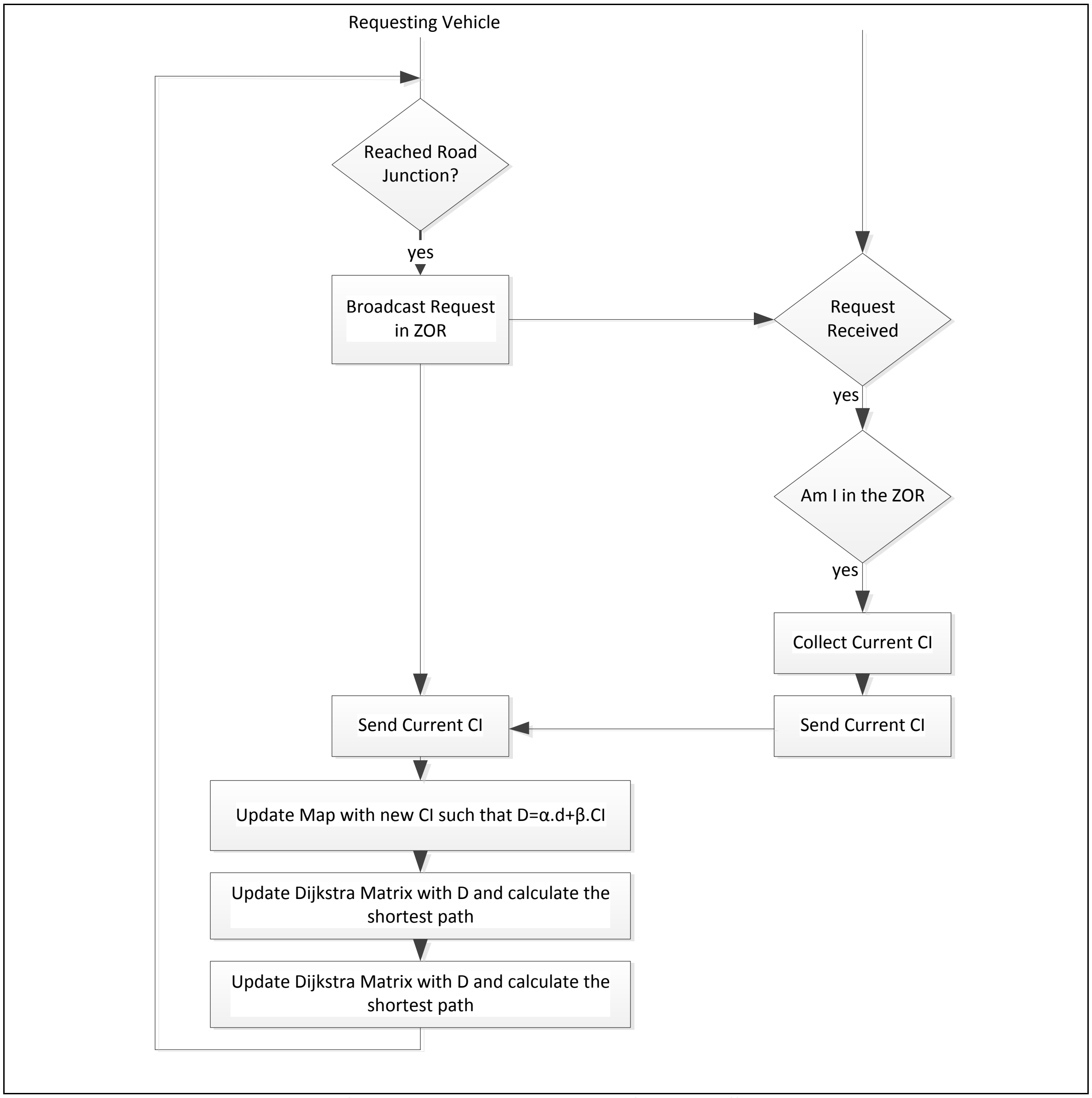


Figure 10: *Flowchart of Detection and Dissemination of Road Traffic Congestion Protocol*

4.2 Geocast for Commercial AD dissemination

This Geocast technique disseminates commercial advertisements to as many vehicles as possible with a smaller number of message forwarding.

4.2.1 System Architecture

Vehicles are GPS enabled and also maintain its neighbor information (address and location). The road side unit (RSU) initiates the dissemination of an advertisement message (AD message) during the time interval I_{AP}. Among the vehicles which are receiving AD message from RSU two are selected as next forwarders. Each vehicle chosen as forwarders selects one or more next forwarders to disseminate AD message to other vehicles [22].

RSU Addr	Sender Addr	Sender's Forwarder List	Coordinates of DR	Time Interval I_{AP}	AD

Figure 11: *Format of Advertisement Message*

4.2.2 Operation

It has divided its operation in following two sub operations:

- Geocast AD message from RSU to entire DR with smaller number of message forwarding.
- Make the Ad message live in DR during I_{AP}.

For the first operation each vehicle selects its next forwarders and includes this information in the forwarded AD message. When a vehicle receives an AD message from RSU, it sets its timer $T_s$proportional to the distance from itself to the center of DR. Now if it receives an AD message from the other vehicles ahead in its moving directions before the expiration of timer T_s then it assumes that another vehicle has already forwarded the message and cancels its timer. Otherwise it forwards the Ad message with the list of next forwarders, which lets its neighbors know that whether they should forward the AD message or not. Number of next forwarder n is defined as:

$$N=R/\delta$$

Here R is transmission range and $0\leq\delta\leq R$. Next forwarder is always selected from the vehicles which are moving ahead in its moving direction.

For the second operation, a vehicle having received an AD message from other vehicle uses a timer T_D to make alive the AD message. The vehicle estimates the time to reach to center of DR by itself based on its average speed, current position and distance from the center. This estimated time is used to set the timer T_D. Now if the vehicle doesn’t receive AD message from other vehicle before theexpiration of T_D, it sends the AD message with next forwarder list for newly entering vehicles in the DR. Otherwise, it cancels timer T_D [22].

4.3 Transmission Range Control and Caching in Geocast

This protocol proposes transmission range control mechanism to improve neighborhood selection technique which significantly decreases network load and end to end delivery delay. It also gives a caching mechanism to deal with high velocity which results constant neighborhood change and unstable routing path [19].

4.3.1 Range Forwarding

In the default greedy forwarding scheme a node that is closest to destination region (DR), which is inside the transmission range is always selected as next forwarder. But the problem with this mechanism is that the selected next forwarder node may go out of transmission range dynamically, resulting a loss of packet.

To improve this neighborhood selection technique, a range r which is smaller than the wireless transmission range is assigned to each node. Now next forwarding node is selected from the neighbors that are:

- Closer to the packet's DR and
- Inside the preconfigured forwarding range r.

4.3.2 Caching Approach

In order to reduce the packet loss during line forwarding, a caching technique is used. It adds a small cache to the routing that keeps those packets that can't be forwarded instantly due to local minimum problem. The cache gets notified about every newly discovered node or about changes in neighbors' position. This notification is handled by a beaconing system. Each beacon used for notification contains current position of the vehicle from where beacon comes.

Whenever a cache gets a beacon, it checks whether there is a packet stored whose destination is closer to the new node. If yes then it forwards the cached packet to new node otherwise ignores the beacon. This means the cache operates on-demand and does not result in higher network overhead as in the case of blind periodical resending attempts [19].

4.3.3Pseudocode for Beaconing System

ALGORITHM-II

GETS BEACON

(Used to perform operation whenever a node gets a beacon packet)

```
N Gets Beacon (S,P)
// S…..address of beacon sender
//P……position of beacon sender
//T…..neighbor Table
1. If there exist an i such that T[i].address==S then // neighbor present
2.         T[i].position=P; // change position
3. Else
4.         Add(S,P) to T;
5. checkCache; // search cache for any possible packet forwarding
```

Chapter-5

PROPOSED GEOCAST ROUTING PROTOCOL:

IMPROVED CACHING TECHNIQUE USING FULL RADIO TRANSMISSION RANGE POWER

5.1 Protocol Design

My proposed "geocast routing protocol with improved caching technique using full transmission range power" is based on "Geocast in Vehicular Environments: Caching and Transmission Range Control for Improved Efficiency" [19]. My protocol has two main concepts that is different from [19] and which makes it novel.

1. Improved Caching Technique
2. Transmission with full Radio Transmission Range Power

5.1.1 Improved Caching Technique:

My goecast protocol improves the caching methodology of paper [19] by not only caching those information packets that can't be forwarded because there is no node in its transmission range that is nearer to the destination region from itself, But also caching those information packet that can't be sent because the node which is nearer to destination region from itself, goes out of range immediately because of high speed movements.

My geocast protocol also perform re-caching, means that it does not send a cached information packets until it can be delivered to either more appropriate node or destination with surety. For this guaranteed delivery, node uses neighborhood table information which contains following information about each neighbor node.

Position Info	Address Info	Average Speed Info	Current time when a neighbor has sent this info

Figure 12: *Format of Neighborhood Table Information*

This improved caching methodology almost eliminated the packet loss during line forwarding as well as during forwarding of cached packet. It either delivered the packet with guarantee or if it can't deliver then cached/re-cached the packet. If we compare [19]'s caching approach then we find that it only reduces packet loss during line forwarding. It does not guarantee packet delivery.

The local cache of each node is maintained by a "Beaconing System". Any change in the position of neighborhood nodes is notified to the node which updates its neighborhood information table and searches the cache for the nodes that can be forwarded using updated information of neighborhood table.

This means our cache operates on demand and doesn't increase the network load by periodically attempting to resend a cached packet.

5.1.2 Transmission with full Radio Transmission Range Power:

My geocast protocol eliminates the transmission range control mechanism of [19] which actually decreases the performance of [19] by increasing the number of forwarding needed to send a packet from source to destination.

My Geo-cast protocol performs a coverage-determination before sending each packet at each node where the packet may be coming from other node or it may be cached packet.

5.1.3 Pseudo code for Coverage Determination Algorithm

ALGORITHM-III

COVERAGE DETERMINATION

(Used to determine that whether a particular node is within my transmission range?)

```
Coverage_Determination( I,C,N)
// I is the node which is actually being tested for under coverage (for receiver)
// C is the center of destination region, N is the current node position(sender)
1.  If (average_speed(N) ≈ average_speed(I))
2.          return true;
3.  If( C is front side of N) then{
4.       If (average_speed(N) ≥average_speed(I) then
5.            return true;
6.  Elseif( ‖( position(N) + ( current_time – time_from_neiborhood_table)*
    average_speed(N) ) – ( position(I) + ( current_time – time_from_neiborhood_table)*
    average_speed(I) ) ‖ <transmission range) then
7.            return true
8.       else
9.           return false;
10. }
11. If( C is rear side of N) then{
12.      If (average_speed(N) ≤average_speed(I) then
13.           return true;
14. Elseif( ‖ (  position(I) + ( current_time – time_from_neiborhood_table)*
    average_speed(I) ) – ( position(N) + ( current_time – time_from_neiborhood_table)*
    average_speed(N) ) ‖ <transmission range) then
15.           return true
16.      else
17.          return false;
18. }
```

Explanation of Coverage Determination Algorithm:

The above algorithm determines that currently whether a node is in my transmission range or not using the information of neighborhood table information about that node. This algorithm is actually indirectly used to find a node which is nearest to the destination region and in my transmission range in a guaranteed way among high speed nodes.

In the 1st steps it compares the average speed of itself with intended receiver node, if it is same then it returns true meaning that intended node may the next receiver. In the 3rd step it checks the position of intended receiver. The intended receiver may be in front side or rear of the current sender node. If it is in front then in the 5th step it compares the average speed of the current node and intended receiver. If the average speed of the current node is greater than the average speed of the intended receiver then it returns true, meaning that intended receiver may be the next forwarder. In the 7th step it calculates the distance traveled by intended receiver during the interval when it had sent the last beacon up to the current time and compares this distance with distance travelled by current node during the same time period. If the difference between these two distances is less then radio transmission range of the node then it returns true meaning that intended receiver may be the next forwarder. In 12th step, If the intended receiver is in rear side of the current node then we calculate the distance travelled by the current node during the time period when the intended receiver had sent the last beacon up to the current time and compares this distance with the distance travelled by the intended receiver during same time period. If the difference between these two distances is less than radio transmission range of the node then it returns true meaning that intended receiver may be the next forwarder node. In the last in step 19th it returns false, if all the above condition fails meaning that the intended receiver can’t be the next forwarder.

Coverage Determination Algorithm is actually used in the modified "Find_Nearest_Neighborhood" algorithm of [19], as the 5th step as follows:

ALGORITHM-IV

MODIFIED FIND NEAREST NEIGHBOR [19]

Modified_Find_Nearest_Nieghbor (I,C,N)

// I is the node which is actually being tested for nearest neighbor
// C is the center of destination region
//N is the current node position(sender)

1. If $\exists$ I : ‖ position(I), C ‖ < ‖ position(N), C ‖
2. and
3. $\forall$ K : ‖ position(I), C ‖ < ‖ position(K), C ‖
4. and
5. Coverage_Determination(I, C, N)

This integration of 5th step in the above algorithm of [19] gives it a guarantee that the node selected by above algorithm as the next forwarder will definitely be in the radio transmission range during the transmission. This integration significantly eliminates the packet loss in the novel geocast protocol.

5.2 Performance Analysis:

The performance evaluation of our new protocol is straight forward and simple. As we have discussed, we have eliminated the concept of "range forwarding" of [19], which actually decreases the transmission range of each node directly from its actual transmission range to combat the high speed movement of nodes in VANETs.

In our new protocol, we retain the transmission range of the node as its actual transmission range. It means that node transmit packets with full transmission power. To combat the high speed movement of node we use the above novel "Coverage Determination" algorithms that instantly determines that whether a particular node will be in our transmission range or not, before transmitting any packet to that node.

It means the number of forwarding need to send a packet from source to destination in our novel protocol will be definitely less than [19] because our novel protocol is using full transmission power and [19] is using decremented transmission power which may be half or one third of the actual transmission power of the nodes.

Therefore we can conclude that our novel protocol will definitely perform better than the [19]. The overall throughput of our novel protocol will be more than [19] and the delay and packet loss will be less than [19].

We have practically proved this analysis by our simulation results which have been discussed in detail in the next chapter.

Chapter-6

IMPLEMENTATION AND SIMULATION RESULT

6.1 Introduction to the OPNET Network Simulator:

The OPNET Modeler gives the facility of the graphical user interface in which the users can model and simulate their networks. For developing different communication structures and implementing different scenarios', different hierarchal layers are present in the environment of the modeling. Users can build a detail model according to the requirement to do the analysis of the system. The systems are designed in the object oriented way, on compilation of the model its produces a discrete event simulation in the C language. After performing the simulation, the results are analyzed with the different statistics related to the performance provided by the OPNET. The following are the different layers in the OPNET which are explained below.

6.1.1 OPNET Project Editor

The Project Editor is the main staging area for creating a network simulation. From this editor, we can build a network model using models from the standard library, choose statistics about the network, run a simulation, and view the results.

We can also create node and process models, build packet formats, and create filters and parameters, using specialized editors that we can access from the Project Editor.

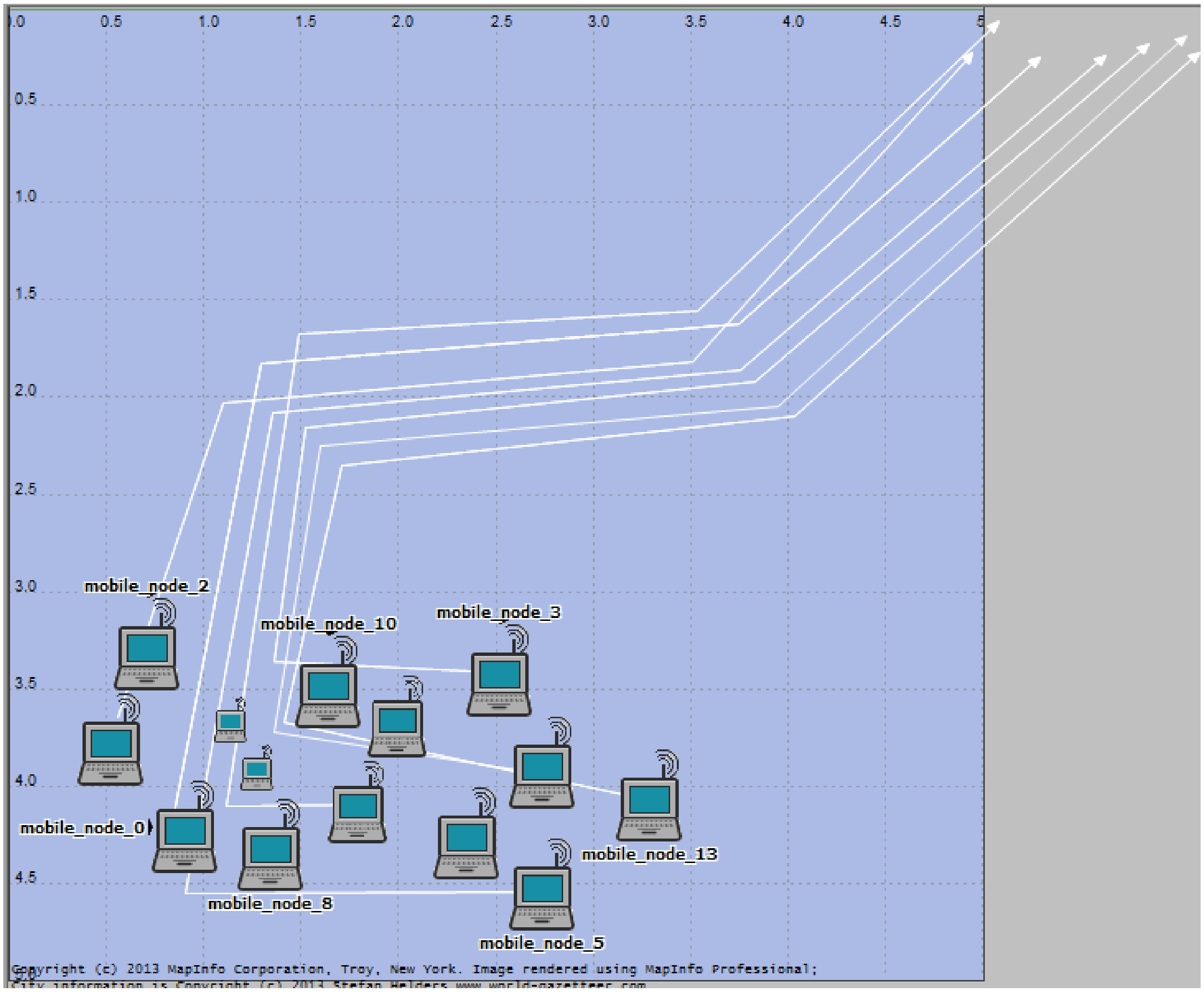


Figure 13 : (a) ***Scenario VANET- GRP with ICFTR***

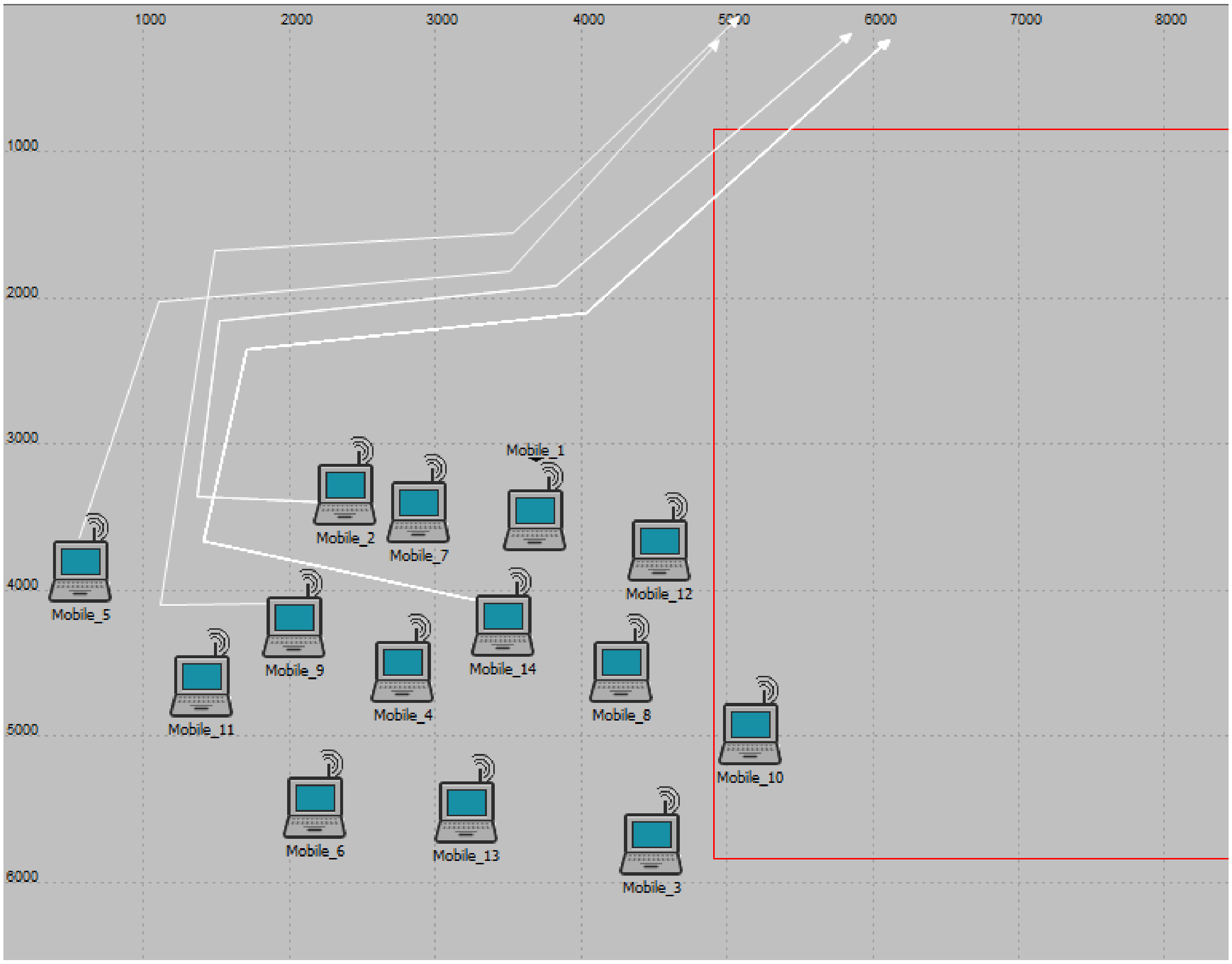


Figure 14 : (b) ***Scenario VANET- GRP***

6.1.2 The Node Editor

The Node Editor defines the behavior of each network object. Behavior is defined using different modules, each of which models some internal aspect of node behavior such as data creation, data storage, etc. Modules are connected through packet streams or statistic wires. A network object is typically made up of multiple modules that define its behavior.

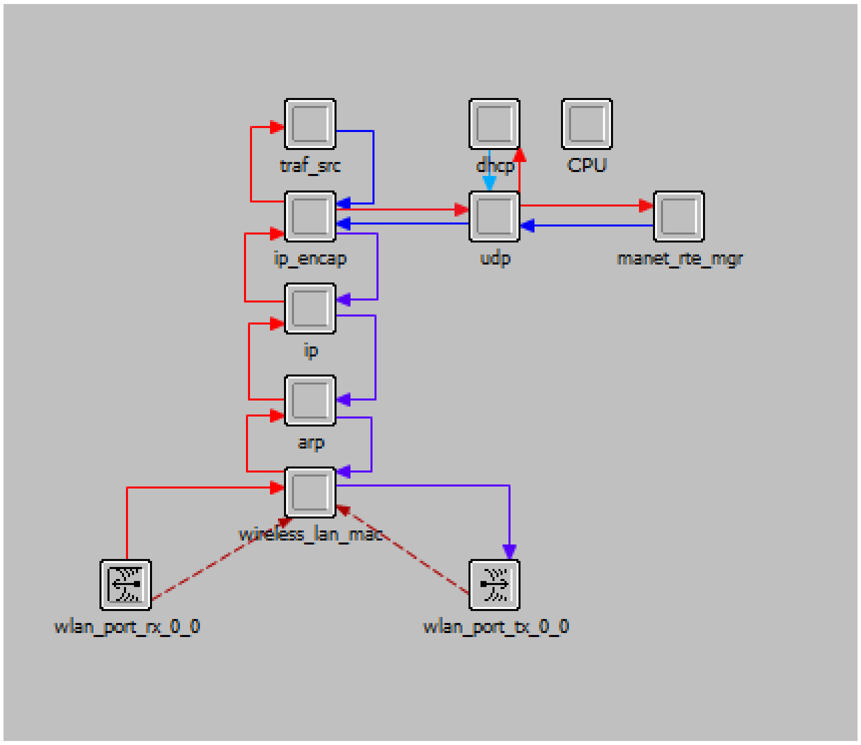


Figure 15 : ***Node Model of Mobile Node in VANET***

6.1.3The Process Model Editor

The Process Editor enables us to create process models, which control the underlying functionality of the node models created in the Node Editor. Process models are represented by finite state machines (FSMs), and are created with icons that represent states and lines that represent transitions between states. Operations performed in each state or for a transition are described in embedded C or C++ code blocks.

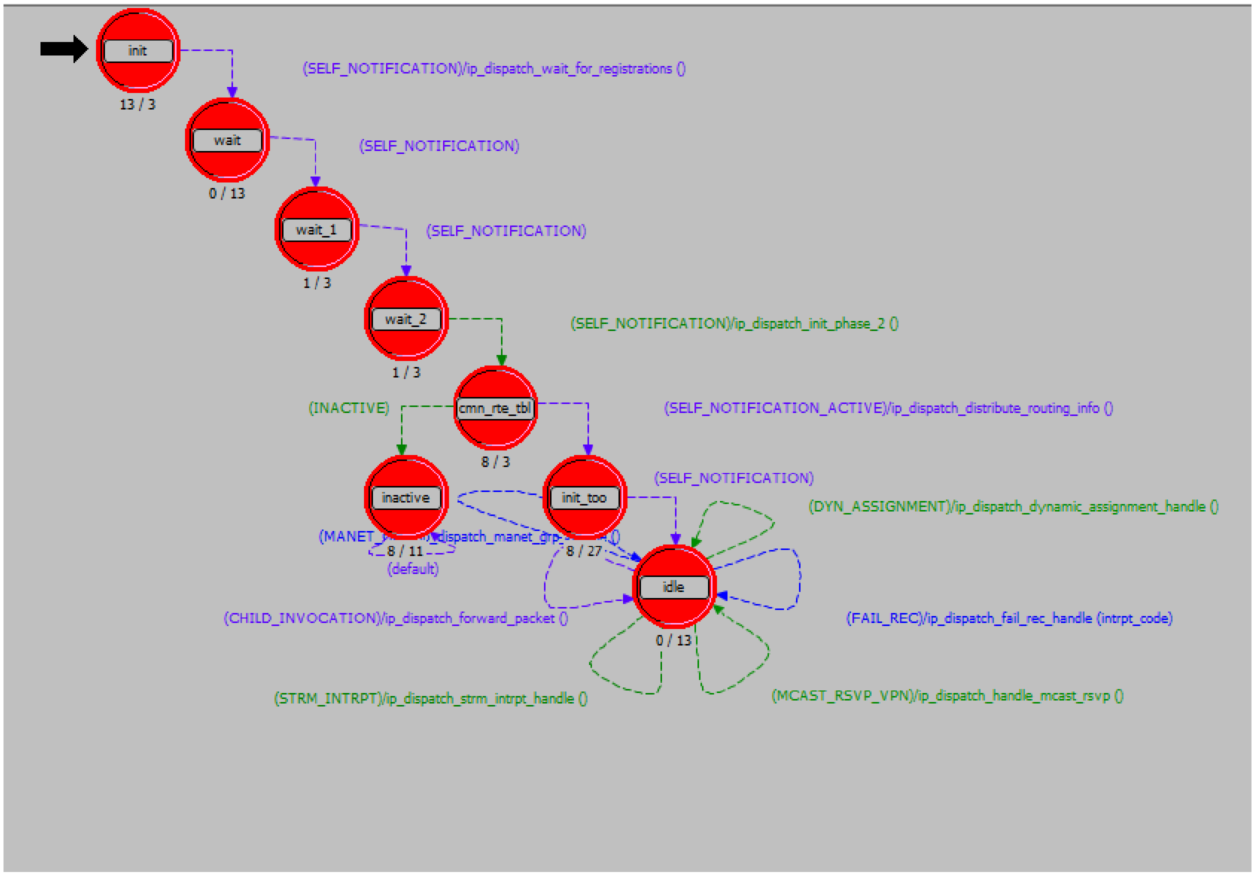


Figure 16 : ***Process Model of IP layer in VANET***

6.2Improved Caching with Full Transmission Range Power

In the process of simulation of our proposed geocast routing protocol, I modified the existing geocast routing protocol in OPNET.

Modifications for implementing "GRP- Improved Caching with Full Transmission Range" have been done in the following files:

- Header File: `grp.h`
- C File: `grp_nbr_table.ex.c`

Modifications have been done in the following functions:

- Function modified in C file `grp_rte_calculate_distance()`
- Function modified in C file `grp_rte_calculate_coordinate ()`
- Function added in C file `grp_rte_check_coverage ()`

6.3 Results

6.3.1Routing Packet Sent (pkt/sec) Analysis

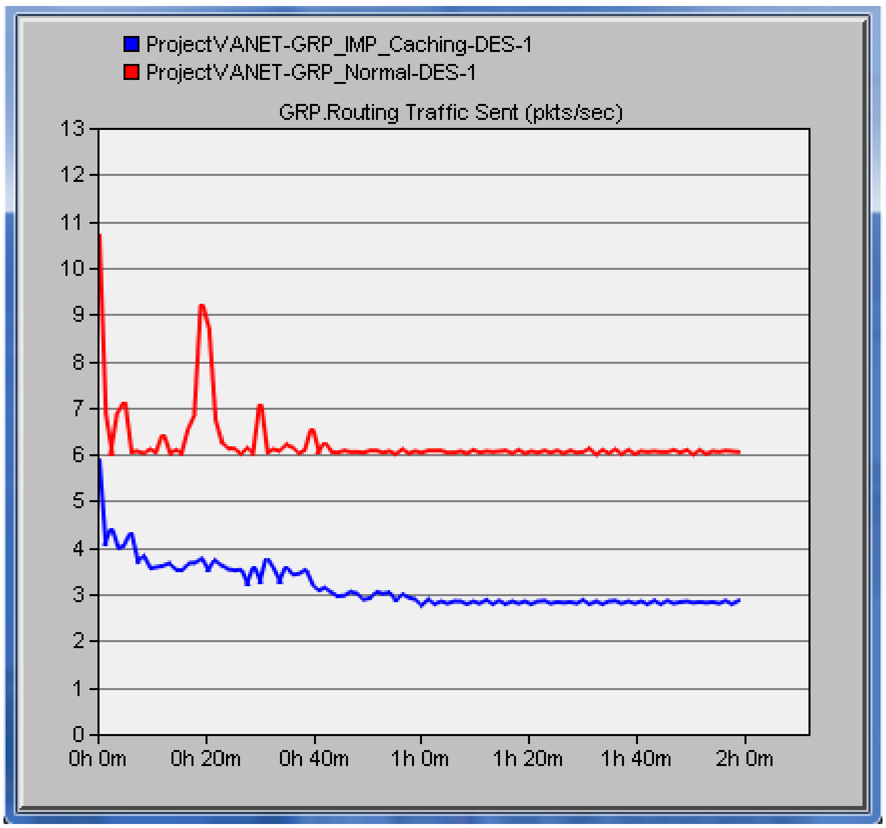


Figure 17 : ***Packet/Second Sent – GRPvsGRP-with-Improved-Caching***

6.3.2 Total Traffic Sent (bits/sec) Analysis

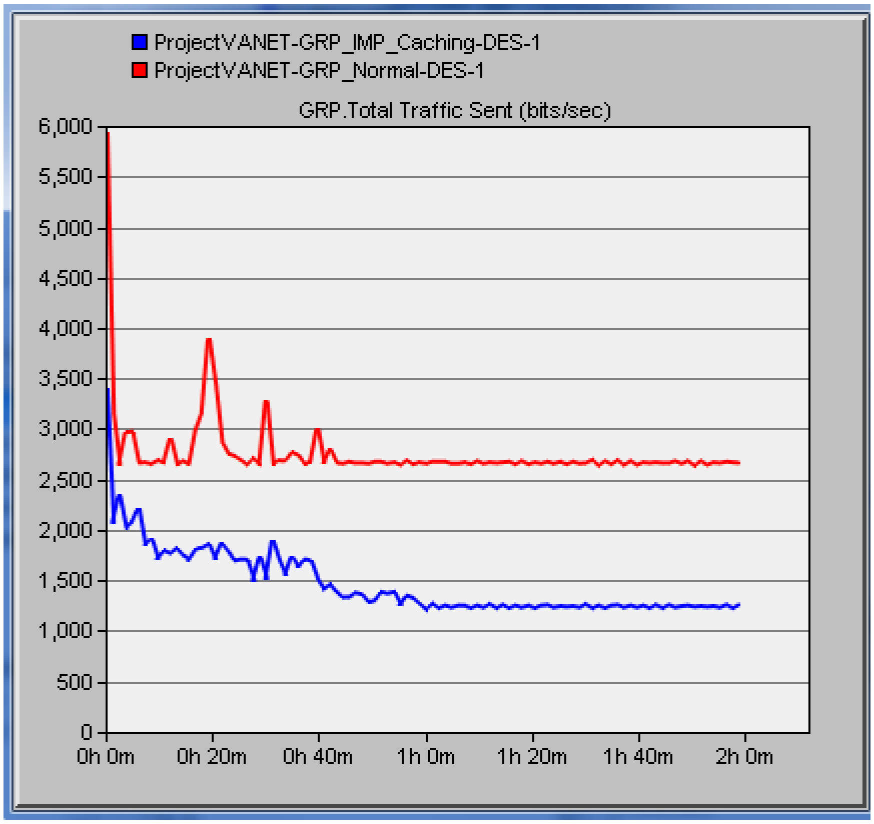


Figure 18 : ***Total Traffic Sent (bits/sec) GRPvsGRP-with-Improved-Caching***

From *Figure 16* and *Figure 17* we can analyze that total traffic sent in the network by our novel GRP-improved caching and full transmission range is less compared to the GRP protocol, however the number of nodes node traversal area and node trajectories are similar.

6.3.3Routing Packet Received (pkt/sec) Analysis

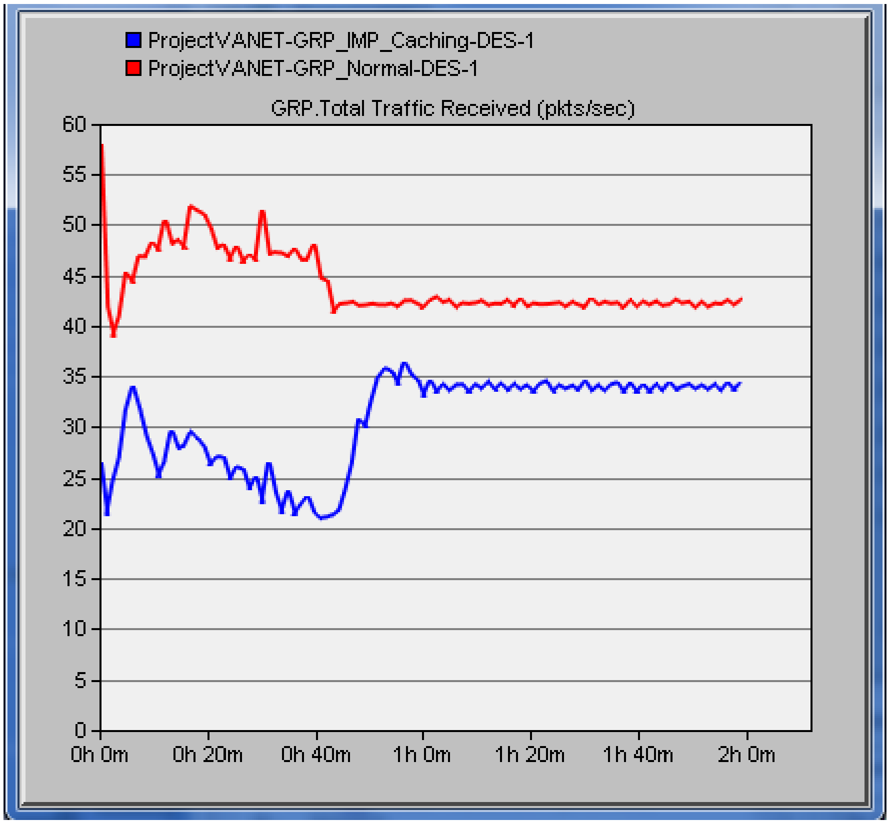


Figure 19 : ***Packet/Second Received – GRPvsGRP-with-Improved-Caching***

6.3.4 Total Traffic Received (bits/sec) Analysis

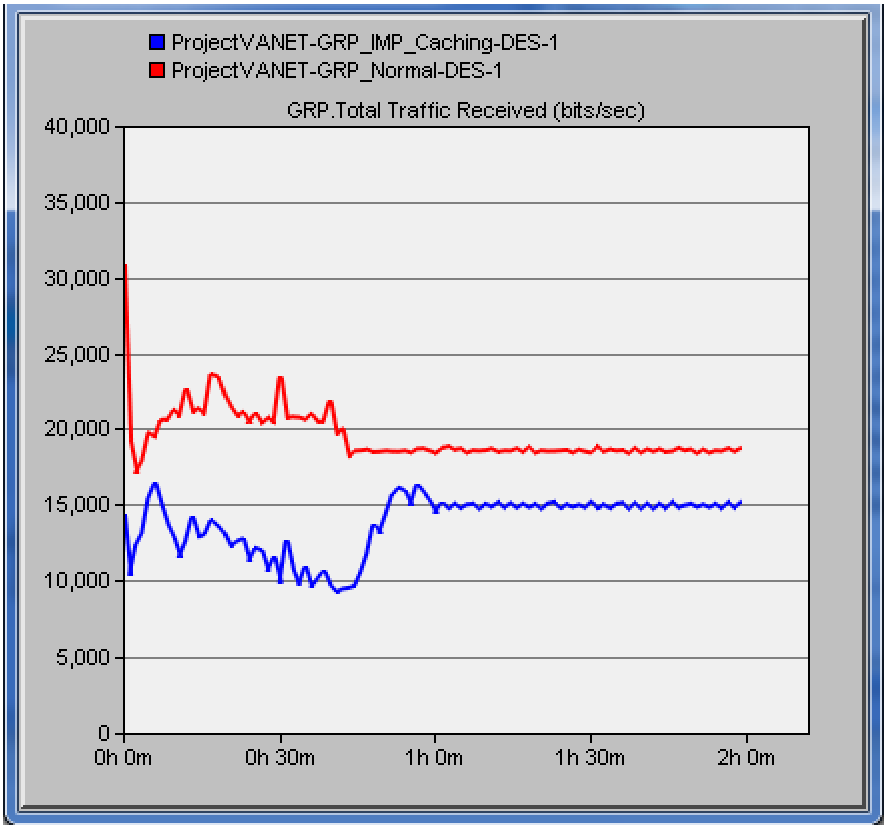


Figure 20 : ***Total Traffic Received (bits/sec) GRPvsGRP-with-Improved-Caching***

From *Figure 18* and *Figure 19* we can analyze that total traffic received in both the network scenarios is in sync with total traffic sent in the network.

Chapter-7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

Because of high speed movement of nodes in VANETs, packet loss which directly affects the routing traffic in the network becomes one of the most critical issue, in the design of vehicular ad-hoc networks (VANETs) routing protocol. Therefore, any routing protocol in VANETs must be designed by taking proper care of packet loss parameter which significantly affects network traffic as well as the overall performance of the protocol. By giving important consideration to packet loss and routing traffic, many geocast routing protocols have been designed for VANETs and many concepts and ideas are used to decrease the packet loss as well as to increase the throughput. “Geocast in Vehicular Environments: Caching and Transmission Range Control for Improved Efficiency” also tries to decrease the packet loss by caching approach and range forwarding concept. It surely decreases packet loss but its range forwarding concept negatively increase the number of hop-to-hop forwarding needed to send a packet from source to destination, which in turn increases network traffic.

Keeping the same concept ‘caching methodology’ in mind, we have designed a “Geocast routing protocol: Improved caching technique with full radio transmission range power” for VANETs. Our protocol not only almost eliminates the packet loss but also improves the throughput with respect to “Geocast in Vehicular Environments: Caching and Transmission Range Control for Improved Efficiency” by using full radio transmission range power which decreases the number of hop-to-hop forwarding needed to send a packet from source to destination. Our protocol eliminates the range forwarding concept of “Geocast in Vehicular Environments: Caching and Transmission Range Control for Improved Efficiency”. In place of range forwarding we use full radio transmission range power to forward packets which significantly improves the throughput of the system.

The work carried out in this dissertation gives insight of one of the most prominent geocast routing protocol for VANETs. It modifies the geocast routing protocol "Geocast in Vehicular Environments: Caching and Transmission Range Control for Improved Efficiency" by eliminating its main concept of range forwarding, adding a novel algorithm Coverage Determination and using full radio transmission range power. It can be concluded that our current work towards geocast routing in VANETs contributed to the knowledge in a modest way by simulating and realizing that geocast routing protocol will significantly perform better using our Coverage Determination algorithm and full radio transmission range power concept, in terms of eliminating packet loss, decreasing network traffic and significantly increasing throughput.

7.2 Future Work

We have modified a geocast routing protocol by adding a novel 'Coverage Determination' algorithm, using full radio transmission range power concept and evaluated its performance in a road environment assuming no obstacles in the environment during radio transmission. In real life road environment buildings, trees are such obstacles that can affect radio transmission. Therefore, further modification can be made to evaluate the performance in obstacle present environment. Some of the modifications are suggested bellow:

- Coverage Determination algorithm can be made adoptive to obstacle at junction point of the road environment. It is because obstacle mostly affects the radio transmission at junction points.
- We can include junction detection mechanism in our algorithm.
- Neighborhood table can be periodically scanned for stale entry information.

.

BIBLIOGRAPHY

1. Fan Li and Ya Wang, "routing in Vehicular Ad-Hoc Networks: Survey" IEEE Vehicular Technology Magazine-I June 2007.
2. V. Namboodiri, M. Agrawal, and L. Gao, "A study on the feasibility of mobile gateways for vehicular ad-Hoc networks" proceeding of the 1st Int. Workshop on vehicular Ad-Hoc Networks, 2004.
3. S.Y. Ni, Y.C. Tseng, Y.S. Chen and J.P. Sheu, "The broadcast storm problem in a mobile ad hoc networks," proceeding of 5th annual ACM/IEEE International Conference on mobile computing and Networking, 1999.
4. M. Durresi, A. Durresi, and L. Barolli, "Emergency broadcast protocol for inter vehicle communications," Proceeding of 11th Int.Con.on Parallel and Distributed Systems (ICPADS'05), 2005.
5. M. Sum, W. Feng, T.-H. Lai, K. Yamada, H. Okada, and K. Fujimura, "GPS-based message broadcasting for Inter-vehicle Communication," Proc. of 2000 Int. Conf on Parallel Processing, 2000.
6. G. Korkmaz, E. Ekici, F. Ozgiiner, and D. Ozgiiner, "Urban multihop broadcast protocol for inter-vehicle communication systems," Proc. of ACM Int. Workshop on Vehicular Ad Hoc Networks, pp. 76-85, 2004.
7. L. Briesemeister, L. Schafer, and G. Hommel, "Disseminating messages among highly mobile hosts based on iner-vehicle communication," Proceeding of IEEE Intelligent Vehicles Symposium, 2000.
8. C. Maihofer and R. Eberhardt, "Geocast in vehicular environments Caching and transmission range control for improved efficiency," Proceeding of IEEE Intelligent Vehicles Symposium, 2004.
9. C. Maihofer, T. Leinmuller, and E. Schoch, "Abiding geo-cast: time stable geo-cast for ad hoc networks," Proceeding of 2nd ACM International Workshop on Vehicular Ad Hoc Networks 2005.

10. J. Blum, A. Eskandarian, and L. Hoffman, "Mobility management in IVC networks," Proc. of IEEE Intelligent Vehicles Symp., 2003.
11. R. A. Santos, A. Edwards, R. Edwards, and L. Seed, "Performance evaluation of routing protocols in vehicular ad hoc networks," Int. Journal of Ad Hoc and Ubiquitous Computing, Vo1. l ,n o. l ,p p. 80-91,2005.
12. B. Karp and H.T. Kung, “GPSR: Greedy perimeter stateless routing for wireless networks,” in Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom), 2000.
13. C. Lochert, M. Mauve, H. Füßler, and H. Hartenstein, “Geographic routing in city scenarios,” ACM SIGMOBILE Mobile Computing and Communications Review (MC2R), vol. 9, no. 1, pp. 69–72, Jan. 2005.
14. Christian Maihofer, “A survey on geocast routing protocols,” IEEE communications Surveys and Tutorials, 2nd quarter, 2004.
15. C. F. Chaisserini, E. Fasolo, R. Furiato, R. Gaeta, M. Garetto, M. Gribaudo, M. sereno, A. Zanella, “Smart broadcast for warning messages in vehicular ad-hoc networks,” Workshop IternoProgetto NEWCOM (NoE), Turin, Itlaly, November 18, 2005.
16. M. Guo, M. H. Ammar and E. W. Zegura, “A vehicle to vehicle live vediosreaming architecture” In Pervasive Computing and Communications, Third IEEE International Conference, 2005.
17. S. Dornbush and A. Joshi, “Street Smart Traffic: Discovering and dissiminating automobile congestion using VANET,” In IEEE Vehicular Technology Conference, 2007.
18. HamidrezaRahbar, KshirasagarNaik and AmiyaNayak, “Dynamic time stable geocast routing in vehicular ad-hoc networks,” IEEE, 2010.
19. Christian Maihofer and Reinhold Eberhardt, “Geocast in vehicular environment: Caching and transmission range control for improved efficiency,” IEEE Intelligent Vehicles Symposium, 2004.

20. AbderrahmaneLakas and MoumenaCheqfah, “Detecting and dissipation of road traffic congestion using vehicular communication,” IEEE, 2009.
21. HumaGhafoor, Khurram Aziz, “Position-based and Geocast Routing Protocols in VANETs” IEEE, 2011.
22. Yujin Lim, SanghyunAhn and Kwon-Hee Cho, “Abiding geocast for commercial ad dissemination in the vehicular ad hoc network,” IEEE International Conference on Consumer Electronics, 2011.
23. Takagi, H.; Kleinrock, L. (March 1984). "Optimal transmission ranges for randomly distributed packet radio terminals". IEEE Transactions on Communications.

APPENDIX

```
// VANET.cpp : Defines the entry point for the console application.
#include "stdafx.h"
#include "math.h"
#include "VANET.h"
#include "time.h"
#define MAX_NEIGHBOURS 15
#define SEND_PACKET_TO_NEIGHBOUR() printf("Packet Sent\n")
#define TRANSMISSION_RANGE 20
double calculate_distance(COORDINATE x, COORDINATE y)
{
      double diffX = x.xPos - y.xPos;
      double diffY = x.yPos - y.yPos;
      double sum = diffX*diffX + diffY * diffY;
      return sqrt(sum);
}
int _tmain(int argc, _TCHAR* argv[])
{

      NODEINFO currentNode, destinationNode;
      NODEINFO neighbourNodeList[MAX_NEIGHBOURS];
      NODEINFO coverageDeterminationNeighbourList[MAX_NEIGHBOURS];
      double destinationDistanceFromCurrentNode;
      double destinationDistanceFromNeighbourNode;
      bool isPacketSent = false;
      while(!isPacketSent)
      {
            destinationDistanceFromCurrentNode =
            calculate_distance(currentNode.position,destinationNode.position)
            ;
            int coverageNodeCount = 0;

            for(int nodeCount = 0; nodeCount< MAX_NEIGHBOURS; nodeCount++)
            {
                  destinationDistanceFromNeighbourNode =
                  calculate_distance(neighbourNodeList[nodeCount].position,de
                  stinationNode.position);
```

```
                neighbourNodeList[nodeCount].distanceToDestination =
                destinationDistanceFromNeighbourNode;
                if((destinationDistanceFromCurrentNode -
                destinationDistanceFromNeighbourNode) > 0) // add to
                coverage determination node list
                {
                      coverageDeterminationNeighbourList[coverageNodeCount+
                      +] = neighbourNodeList[nodeCount];
                }
                sort(coverageDeterminationNeighbourList,coverageNodeCount);
          }

          for(int nodeCount = 0; nodeCount<coverageNodeCount; nodeCount++)
          {
                if(isInCoverageArea(currentNode,
                                  coverageDeterminationNeighbourList[nodeCo
                            unt],
                                  destinationNode))
                {
                      SEND_PACKET_TO_NEIGHBOUR();
                      isPacketSent = true;
                      break;
                }
          }
    }
    getchar();
    return 0;
}

void sort(NODEINFO *nodeList, int noOfNodes)
{
      NODEINFO tempNode;
      for(int nodeCount = 0; nodeCount<noOfNodes; nodeCount++)
      {
            for(int nodeCount1 = nodeCount+1; nodeCount1<noOfNodes;
            nodeCount1++)
```

```
            {
                  if(nodeList[nodeCount].distanceToDestination >
                  nodeList[nodeCount1].distanceToDestination)
                  {
                        tempNode = nodeList[nodeCount];
                        nodeList[nodeCount] = nodeList[nodeCount1];
                        nodeList[nodeCount1] = tempNode;
                  }
            }
      }
}

void calculateCurrentCoordinate(COORDINATE x, double direction, double
distance, COORDINATE *newLocation)
{
      double newX = x.xPos + distance * cos(direction);
      double newY = x.yPos + distance * sin(direction);
      newLocation->xPos = newX;
      newLocation->yPos = newY;
}
bool isInCoverageArea(NODEINFO current, NODEINFO neighbour, NODEINFO
destination)
{
      if((current.averageSpeed - neighbour.averageSpeed) <= 2)
            return true;
      if(current.direction >= 0 && current.direction < 180) //destination is
      in frontside
      {
            if(current.averageSpeed >= neighbour.averageSpeed)
                  return true;
            else
            {
                  double relativeSpeed = neighbour.averageSpeed -
                  current.averageSpeed;
                  long timeElapsedSinceBeacon = (long)time(0) -
                  neighbour.timeLastBeaconPacket;
```

```
            double distanceTravelledSinceBeacon = relativeSpeed *
            timeElapsedSinceBeacon;
            COORDINATE newLocationCurrentNode,
            newLocationNeighbourNode;
            calculateCurrentCoordinate(current.position,
            current.direction, distanceTravelledSinceBeacon,
            &newLocationCurrentNode);
            calculateCurrentCoordinate(neighbour.position,
            neighbour.direction, distanceTravelledSinceBeacon,
            &newLocationNeighbourNode);

            double distanceBetween =
            calculate_distance(newLocationCurrentNode,
            newLocationNeighbourNode);

            if(distanceBetween <= TRANSMISSION_RANGE)
            {
                  return true;
            }
            else
                  return false;
      }
}

else  //destination is in rear side
{
      if(current.averageSpeed <= neighbour.averageSpeed)
            return true;
      else
      {
            double relativeSpeed = current.averageSpeed -
            neighbour.averageSpeed;
            long timeElapsedSinceBeacon = (long)time(0) -
            neighbour.timeLastBeaconPacket;
            double distanceTravelledSinceBeacon = relativeSpeed *
            timeElapsedSinceBeacon;
```

```
                    COORDINATE newLocationCurrentNode,
                    newLocationNeighbourNode;
                    calculateCurrentCoordinate(current.position,
                    current.direction, distanceTravelledSinceBeacon,
                    &newLocationCurrentNode);
                    calculateCurrentCoordinate(neighbour.position,
                    neighbour.direction, distanceTravelledSinceBeacon,
                    &newLocationNeighbourNode);

                    double distanceBetween =
                    calculate_distance(newLocationCurrentNode,
                    newLocationNeighbourNode);

                    if(distanceBetween <= TRANSMISSION_RANGE)
                    {
                          return true;
                    }
                    else
                          return false;
              }
        }
        return true;
}
```