Content Distribution using Network Coding in Vehicular Ad-Hoc Network (VANET)

BY

Ajaykumar M. Patel

(10MCES04)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AHMEDABAD 382481

May 2013

Content Distribution using Network Coding in Vehicular Ad-Hoc Network (VANET)

Major Project

Submitted in partial fulfillment of the requirements

For the degree of

Master of Technology in Computer Science and Engineering

By

Ajaykumar M. Patel
(10MCES04)

Guide

[Prof. Zunnun Narmawala]

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
AHMEDABAD 382481

May 2013

Declaration

I, **Ajaykumar M. Patel**, Reg.No. **10MCES04** , give undertaking that the Major Project entitled **"Content Distribution using Network Coding in Vehicular Ad-Hoc Network (VANET)"** submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in Computer Science and Engineering of Nirma University, Ahmedabad, is the original work carried out by me and I give assurance that no attempt of plagiarism has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

Ajaykumar M. Patel (10MCES04)
Date: May 18, 2013
Place: Ahmedabad

Endorsed By:
Signature of Guide:

Certificate

This is to certify that the Major Project entitled "Content Distribution using Network Coding in Vehicular Ad-Hoc Network (VANET)" submitted by Ajaykumar M. Patel (10MCES04), towards the partial fulfillment of the requirements for the degree of Master of Technology in Computer Science and Engineering of Nirma University, Ahmedabad is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof. Zunnun Narmawala
Guide, Assistant Professor,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad

Prof. Vijay Ukani
Associate Professor and PG-Coordinator,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad

Dr. Sanjay Garg
Professor and Head,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad

Dr K Kotecha
Director,
Institute of Technology,
Nirma University,
Ahmedabad

Abstract

Vehicular Ad-Hoc Network (VANET) is a form of Mobile ad-hoc network that provides communication between vehicles and between infrastructure (Road Side Unit) and vehicles. Various approaches of data dissemination is used to distribute data among vehicles in vehicular network. Here, data dissemination approaches, mechanisms, models are extensively reviewed and concluded that data dissemination techniques depends on the type of applications and data to be transmitted among vehicles. However, type of VANET applications and inherent characteristics such as different network density, fast movement of vehicles make data dissemination quite challenging. Network coding is a mechanism in which nodes encode two or more incoming packets and forward encoded packets instead of forwarding them as it is. In the network coding, the reception of data does not depend on receiving specific packets but on receiving sufficient number of independent packets. We use network coding in which packets are grouped into generations. The goal of the project is to create protocol that distribute content using network coding to improve the performance. The propose protocol works in two modes: Adhoc mode and Infrastructure mode. In Adhoc mode, vehicles are communicating with each other to transfer different contents. While in Infrastructure mode, vehicles are communicating with Road Side Units for different contents. By using simulation we compared the proposed protocol in terms of delay and delivery ratio using conventional scheme and same protocol using network coding. Simulation results shows that Generation by Generation Random Linear Network Coding (G-by-G RLNC) increases the delivery ratio as compared to conventional scheme without using network coding.

Acknowledgements

With immense pleasure, I would like to present this report on the dissertation work related to "Content Distribution using Network Coding in Vehicular Ad-Hoc Network (VANET)".

I would first of all like to offer thanks to **Prof. Zunnun Narmawala**, Guide & Asst. Prof., CSE Department, Institute of Technology, Nirma University, Ahmedabad whose keen interest and excellent knowledge base helped me to finalize the topic of the dissertation work. His constant support and interest in the subject equipped me with a great understanding of different aspects of the required architecture for the project work. He has been a great motivating factor in outlining the flow of my work.

I like to give my special thanks to **Prof. Vijay Ukani**, Programme Co-ordinator M.Tech. CSE, Institute of Technology, Nirma University, Ahmedabad for his continual kind words of encouragement and motivation throughout the Major Project.

My sincere thanks and gratitude to **Dr. Sanjay Garg**, Professor and Head, Computer Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his continual kind words of encouragement and motivation throughout the Dissertation work.

I am also thankful to **Dr K Kotecha**, Director, Institute of Technology for his kind support in all respect during my study.

I would like to thank The Almighty, my family, especially my wife, for supporting and encouraging me in all possible ways. I would also like to thank all my friends who have directly or indirectly helped in making this dissertation work successful.

- Ajaykumar M. Patel (10MCES04)

Abbreviation Notation

VANET .... Vehicular Ad-Hoc Network
OBU .... On Board Unit
RSU .... Road Side Unit
V2I .... Vehicle to Infrastructure
V2V .... Vehicle to Vehicle
NC .... Network Coding
LNC .... Linear Network Coding
RLNC .... Random Linear Network Coding
ACK .... Acknowledgement
PNCO .... Partial Network Coding
AF .... Amplify-and-Forward
DF .... Decode-and-Forward
ONC .... Opportunistic Network Coding
G-by-G .... Generation-by-Generation
GenSize .... Generation Size
GF .... Galois Field

Contents

List of Tables

List of Figures

Chapter 1

Motivation

1.1 Motivation

The growth of the increased number of vehicles are equipped with wireless transceivers to communicate with other vehicles to form a special class of wireless networks, known as vehicular ad hoc networks or VANETs [8]. Generally communications in wireless networks can be classified as infrastructure-less, infrastructure-based a wired backbone with wireless last-hop, or hybrid architecture using vehicle-to-vehicle (V2V) or inter-vehicle communication (IVC) that does not rely on a fixed infrastructure, but can exploit it for improved performance and functionality when it is available. V2V based communication provides data dissemination among vehicles And V2I(Vehicle-to-Infrastructure) /I2V(Infrastructure-to-Vehicle) based communication can able to provide data dissemination service like Internet,etc. Furthermore, they prevent the hosts from exchanging position and routing table updates throughout the network owing to their limited bandwidth, frequent topology changes, and large-scale coverage[9]. Two main applications of vehicular Ad hoc networks are Safety Applications and Comfort Applications and detail list is mention in Introduction of VANET section. VANET can be used for safety applications (this is the primary goal), for driving support information services (information about parking places, points-of-interest, etc.)

and, in some concepts, it can offer classic Internet services including high quality media streaming. Of course, the above mentioned VANET problem with highly unreliable communication will limit the service portfolio in the same way as it is observed in other networks. In VANET content distribution is done with help of broadcasting. Simple broadcasting mechanism is flooding. In dense scenario this technique is not scalable because huge amount of messages are redundant in the network. Additionally, if a simple flooding is used, a data item may not be delivered to all vehicles in the data effective area because the connectivity between vehicles is not guaranteed when the vehicle density is low [5]. Several techniques to avoid the broadcast problem have been proposed like Timer based, Hop limited simple forwarding, Map based or geographic forwarding, opportunistic forwarding are the examples. Three basic data dissemination models are same-direction, opposite direction and bi directional. According to different scenarios appropriate model must be selected. Different routing strategies are used in VANET for data dissemination like redundant packets without affecting performance. Another approach is network coding where information is not routed but coded.

In Network coding [1], the basic underlying idea is that one can approach the broadcast capacity by allowing the intermediate nodes within a network to intelligently code and decode the information carried by the different flows. The idea was originally meant for wired networks with special focus on exploiting the multicast nature of transmissions, as described using the famous butterfly network example. Later the idea was adopted for wireless networks by exploiting the inherent broadcast nature of the medium. Network coding can improve throughput, robustness,complexity, and security. Consider the example of butterfly network, there are two source nodes has some value A and B (topmost nodes in figure). There are two destination nodes (bottom in figure), they want to receive both A and B. Each edge in figure can handle only a single value at a time. Using routing, central line can carry A or B, but not both at same time. Suppose we send value A through the center line, then the left destination node would receive A twice and not know value B at all. Sending B poses a same problem for the right destination node. Routing is not sufficient because no routing technique can transmit both value A and B simultaneously to both destination nodes.

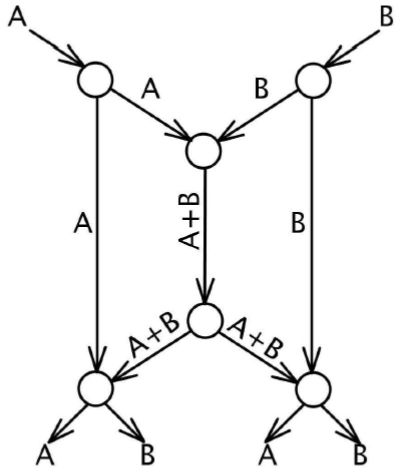


Figure 1.1: Example of Butterfly Network [1]

With the help of simple code, as shown in figure, we get both A and B to both destinations simultaneously by sending the summation of the symbols through the center line (means, we have to encode A and B using the formula "A + B"). The left destination node receives A and "A + B", and can find B by making subtraction the two values. This is the linear operation.

Now throughput: Here destination nodes can receive four messages in the cost of two messages. So, 100% improvement is achieved.

Multicast networks are much simpler for using network coding because of the extra introduced complexity for intelligent decoding at the receiver in unicast transmissions.

Apart from a broad classification of wired and wireless networks, network coding has manifold applications in various kinds of networks, such as P2P networks, sensor networks, mesh networks, vehicular networks, delay-tolerant networks, multimedia services and even in network security.

1.2 Thesis Outline

In Chapter 2, literature survey of related work and identified open issues are presented. In Chapter 3, problem definition is given. In Chapter 4, simulation system is explained. In Chapter 5, Problem, Objective, Assumptions taken while solving problem and Intended Outcomes are mentioned. We present and discuss simulation results and analysis in Chapter 6. Finally, in chapter 7 concluding remarks and future work is presented.

Chapter 2

Literature Survey

2.1 Introduction

2.1.1 Introduction to VANET

Vehicular Ad-Hoc Network (VANET) is a form of Mobile ad-hoc network that provides communication between vehicles and between infrastructure (Road Side Unit) and vehicles. Up to now, Vehicular Ad Hoc Network (VANET) becomes a popular research topic in the area of wireless network. The ITS (Intelligent Transportation System)is one important application of VANET among other application. In a VANET environment this devices are either stations (STAs) installed in vehicles as on board units (OBUs) or access points (APs) strategically located in fixed points along the road, and hence usually referred to as road side units (RSUs). [10] The most important problem is the intermittent connectivity of VANET nodes caused by high mobility of vehicles. The high dynamics of vehicles combined with usage of short range communications make the connectivity among the vehicles very unstable, so even the best effort service cannot be guaranteed. Of course, the communication quality differs in cities during traffic jams (where we may obtain dense and stable networks) from one in highways (where the VANET network is sparse and the topology is highly dynamic). VANET can be used for safety applications (this is the primary goal), for driving support information services (information about parking places, points-of-interest, etc.) and, in some concepts, it can offer classic Internet services including high quality media streaming. Some of the VANET projects [11] are PRECIOSA, Fleetnet, WILL WARN, Cartalk 2000, SAFESPOT - "Smart Vehicles on Smart Roads", Watch-Over, aktiv, SIM-TD, PreDRIVE C2X, AutoNomos, Drive-In, EVITA - "E-Safety Vehicle Intrusion Protected Applications", COOPERS - "CO-OPerative SystEms for Intelligent Road Safety", CVIS - "Cooperative Vehicle-Infrastructure Systems", COMeSafety, Ertico, eSafetySupport, MARTA "Mobility and Automotion through Advanced Transport Networks", GeoNet - b"Geographic addressing and routing for vehicular communications", PReVENT - "PReVENTive and Active Safety Applications", SEVECOM - "SEcure VEhicle COmmunication", Safe Intelligent Mobility Test Area Germany (SIM-TD) and CARLINK Consortium.

2.1.2 VANET Application Areas

Following are the different application areas where VANET can be used [12][13].

a. Active Safety

	Situation/Purpose	Application Examples
Active Safety	1. Dangerous road features	1. Curve speed warning 2. Low bridge warning 3. Warning about violated traffic lights or stop signals
	2. Abnormal traffic and road conditions	1. Vehicle-based road condition warning 2. Infrastructure-based road condition warning 3. Visibility enhancer 4. Work zone warning
	3. Danger of collision	1. Blind spot warning 2. Lane change warning 3. Intersection collision warning, 4. Forward/Rear collision warning, 5. Emergency electronic brake lights, 6. Rail collision warning, 7. Warning about pedestrians crossing
	4. Crash imminent	1. Pre-crash sensing
	5. Incident occurred	1. Post-crash warning 2. Breakdown warning 3. SOS service

Figure 2.1: VANET Active Safety Applications

b. Public Service

	Situation/Purpose	Application Examples
Public Service	1. Emergency response	1. Approaching emergency vehicle warning 2. Emergency vehicle signal preemption 3. Emergency vehicle at scene warning
	2. Support for authorities	1. Electronic license plate, 2. Electronic drivers license, 3. Vehicle safety inspection, 4. Stolen vehicles tracking

Figure 2.2: VANET Public Service Applications

c. Improved driving

	Situation/Purpose	Application Examples
Improved driving	1. Enhanced Driving	1. Highway merge assistant 2. Left turn assistant 3. Cooperative adaptive cruise control 4. Cooperative glare reduction 5. In-vehicle signage 6. Adaptive drive train management
	2. Traffic Efficiency	1. Notification of crash or road surface conditions to a traffic operation center 2. Intelligent traffic flow control 3. Enhanced route guidance and navigation 4. Map download/update 5. Parking spot locator service

Figure 2.3: VANET Improved driving Applications

d. Business/Entertainment

	Situation/Purpose	Application Examples
Business/ Entertainment	1. Vehicle Maintenance	1. Wireless diagnostics, 2. Software update/flashing 3. Safety recall notice 4. Just-in-time repair notification
	2. Mobile Services	1. Internet service provisioning 2. Instant Messaging 3. Point-of-interest notification
	3. Enterprise solutions	1. Fleet management 2. Rental car processing 3. Area access control 4. Hazardous material cargo tracking
	4. E-Payment	1. Toll collection 2. Parking payment 3. Gas payment

Figure 2.4: VANET Business/Entertainment Applications

2.1.3 Characteristics of VANET

When deploying of a vehicular networking system, several issues have to be resolved, ranging from applications development up to economical issues.
These unique characteristics of these networks are as follows [6][14][15]:

- **Dynamic, large-scale, and rapidly changing topology:** The speed and choice of path defines the dynamic topology of VANET. If we assume two vehicles moving away from each other with a speed of 60 mph (25m/sec) and if the transmission range is about 250m, then the link between these two vehicles will last for only 5 seconds. This defines its highly dynamic topology.

- **Frequent fragmentation resulting in small effective network diameter:** The above feature necessitates that in about every 5 seconds or so, the nodes needed another link with nearby vehicle to maintain seamless connectivity. But in case of such failure, particularly in case of low vehicle density zone, frequent disruption of network connectivity will occur. Such problems are at times addressed by road-side deployment of relay nodes.

- **Hard Delay Constraints:** The safety aspect (such as accidents, brake event) of VANET application warrants on time delivery of message to relevant nodes. It simply cannot compromise with any hard data delay in this regard. Therefore high data rates are not as important an issue for VANET as overcoming the issues of hard delay constraints.

- **Variable, highly dynamic scale and network density :** Due to high mobility vehicles network density change drastically.

- **Interaction with onboard sensors:** This sensors helps in providing node location and their movement nature that are used for effective communication link and routing purposes.

- **Mobility Modeling and Prediction:** The above features for connectivity therefore needed the knowledge of node positions and their movements which as such is very difficult to predict keeping in view the nature and pattern of movement of each vehicle. Nonetheless, a mobility model and node prediction based on study of predefined roadways model and vehicle speed is of paramount importance for effective network design.

- **Communication Environment:** The mobility model highly varies from highways to that of city environment. The node prediction design and routing algorithm also therefore need to adapt for these changes. Highway mobility model, which is essentially a one-dimensional model, is rather simple and easy to predict. But for city mobility model, street structure, variable node density, presence of buildings and trees that behave as obstacles to even small distance communication make the model application that very complex and difficult.

2.2 Conventional Content distribution Approaches

Broadcasting appears to be an attractive solution due to its low cost and large potential volumes of data. Location-aware broadcasting would limit the broadcast range only to the site of interest, thus reducing overhead (avoiding the broadcast storm problem). Clustering is another approach to optimize the message dissemination process: neighbor nodes form clusters, manageable units that limit the broadcasting range. E.g. in a clustering method called Local Peer Groups (LPGs) is proposed, where nodes can either form static or dynamic clusters.

As defined in VANET architecture Content distribution between components is categorized as V2I/I2V and V2V [14][16][6]. Content distribution among vehicles depends on the type of assumed network architecture.

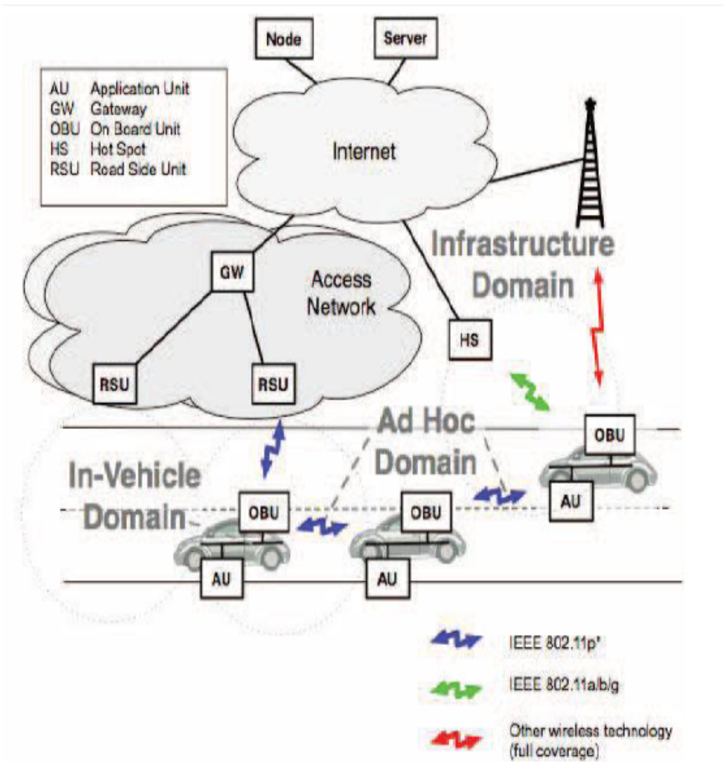


Figure 2.5: VANET Architecture [2]

2.2.1 V2I(Vehicle-to-Infrastructure) /I2V(Infrastructure-to-Vehicle) dissemination

In V2I and I2V dissemination, two approaches are considered as mention below:

- **Push based (broadcast)**
  - Data delivery from source to many vehicles.
  - Push based station pushes out the data to everyone. This is useful for the popular data like Traffic alerts, Weather alerts and here is no cross traffic resulting in low contention. Its drawback is that everyone might not be interested in the same data.
- **Pull based (on-demand)**
  - Data query from one vehicle to specific targets.
  - In Pull based communication, a Request Response model is followed. This is useful for unpopular / user-specific data like Email, Webpage requests. Its drawback lays in lots of cross traffic that result in Contention, Interference, and Collisions.

2.2.2 V2V (Vehicle-to-Vehicle) dissemination

In V2V dissemination, two approaches are considered as mention below:

- **Flooding**
  - In the this technique, each individual vehicle broadcasts its own information periodically. Upon receiving messages by vehicle, it stores it then rebroadcasts the message.
  - This is useful for delay sensitive applications.
  - In dense scenario this technique is not scalable because huge amount of messages are redundant in the network. Additionally, if a simple flooding is used, a data item may not be delivered to all vehicles in the data effective area because the connectivity between vehicles is not guaranteed when the vehicle density is low. Several techniques to avoid the broadcast problem have been proposed like Timer based, Hop limited simple forwarding, Map based or geographic forwarding, opportunistic forwarding are the examples. Several Network Coding protocol has been proposed to solve the problem of scalability related to flooding. Network coding improves the performance of content distribution by mitigating the scheduling problem given only a local knowledge of the network.
- **Relaying**
  - Stored data about the other vehicles ahead and it is propagated it with every broadcast period.
  - The relay node is responsible to forward the packet further and so on. As it is clear contention is less in compared with the flooding approach and it is scalable for dense networks. This is due to the less of number of the nodes participating in forwarding message and consequently generated overhead is less. However, selecting relay node and ensure reliability are two challenges that need to be addressed.

Table I: Comparison of Content Distribution Approaches

Dissemination approach	Mechanism	Pros	Cons
Push based	Data delivery from source to many vehicles.Push based station pushes out the data to everyone.	Suitable for popular data like Traffic alerts, Weather alerts.	Not suitable for unpopular data
Pull based	Data query from one vehicle to specific targets. In Pull based communication, a Request Response model is followed.	Suitable for non-popular data, user specific data	Cross traffic incurs heavy interferences, collisions, contention.
Flooding	In the this technique, each individual vehicle broadcasts its own information periodically. Upon receiving messages by vehicle, it stores it then rebroadcasts the message.	Reliably and quickly distribute data	Not suitable for dense networks.
Relaying	Stored data about the other vehicles ahead and it is propagated it with every broadcast period.	Works well in dense networks	Selecting best next hop reliability is difficult

2.3 Broadcasting mechanisms and suggested approaches

Broadcast mechanisms for data dissemination are as follows [17]:

2.3.1 Simple Flooding

In this method, a source vehicle broadcasts a message to all of its neighboring vehicles. Each neighbor vehicle that receives this message further broadcasts this message to all its neighbors unless all the vehicles receives this message. High amount of redundancy and bandwidth consumption is the problem with this approach. Contention and collision is done due to redundant broadcast. This problem is referred as broadcast storm problem. The approaches to overcome from it is mention in next subsections.

2.3.2 Probability Based Methods

Probability based methods are as follows:

Table II: Probability Based Methods

Method Name	Approach
Probability based approach	It solves the problem of flooding. To solve the problem of re-broadcasting, congestion and collision, each node uses predetermined probability for re-broadcasting. The disadvantage of method is that some node may not receive message.
Counter based scheme	It uses three variables: a counter (k) , a random assessment delay (RAD) and a threshold (K). k increases the number of times a node receives redundant message. The message is dropped when $k > K$ and RAD expires. Otherwise, the message is rebroadcast. Some node do not broadcast message, when network density is high.

2.3.3 Area Based Methods

Area based methods are as follows:

Table III: Area Based Methods

Method Name	**Approach**
Distance based approach	It is similar to counter based approach. It uses the distance - d between source and receiving node to decide whether to rebroadcast or drop the message. A threshold value, D is set and RAD is checked constantly. If $d > D$ and RAD expires then the message will dropped, otherwise it will rebroadcasted.
Location based approach	Each node adds its own location in he header of the message it sends or rebroadcast. The additional coverage area is calculated based on location of sender for rebroadcasting. If value is lower than defined threshold and RAD does not expires, the message is rebroadcast. Otherwise it is dropped. The disadvantage of this approach is cost od calculating additional coverage areas.

2.3.4 Neighbour Knowledge Methods

Neighbour based methods are as follows:

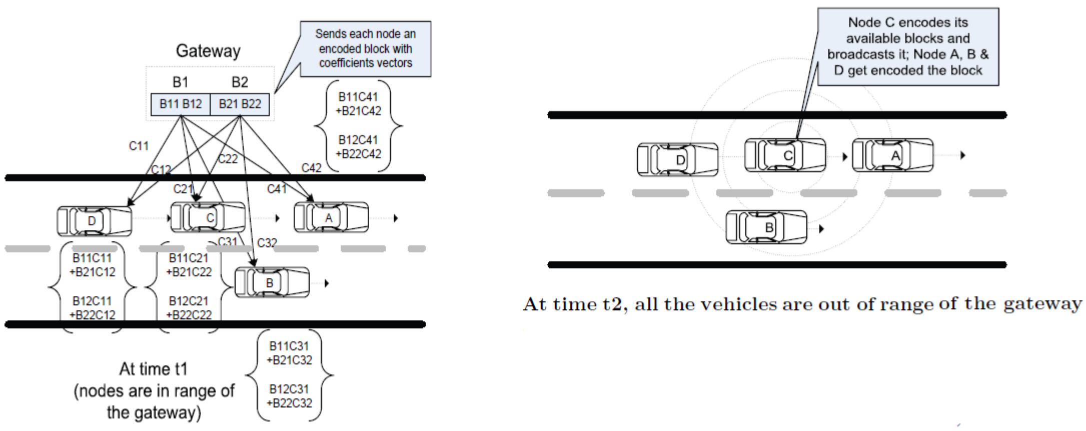
Table IV: Neighbour Based Methods

Method Name	**Approach**
Self Pruning	Each node has its neighbours knowledge. The receiving node compares its neighbours list to the senders neighbour list. If additional node could not be reached, the message is dropped. Otherwise the node will rebroadcast. It shows that even with this method a situation can come which causes message redundancy.
Scalable Broadcasting Approach	It is similar to self pruning but knowledge upto two hop distance is required. So the probability of each node receive the message is increase as compare to self pruning.
Ad Hoc Broadcasting Approach	This approach attempts to find its neighbours that can cover the most the network nodes. This node is then given the designation of Broadcast Relay Gateway (BRG) and only this node is allowed to broadcast the message. Due to lack of local information of whether to rebroadcast or not, it is not useful in highly mobile VANET network.

2.4 Content Distribution Mechanisms

Several content distribution mechanisms are explained as mention below:

- **VANETCODE - Network Coding to Enhance Cooperative Downloading in VANET:** [3] In VANETCODE, encoding and distribution data is done by the gateway and straight freeway (highway) is considered. All gateways have all data blocks of file. All nodes are moving in the same direction. The priority based mechanism (more blocks higher priority) is used to transfer the data. Dissemination of the encoded blocks amongst the one-hop neighbours only and is entirely independent of routing.



Encoding and distribution by the gateway

At time t2, all the vehicles are out of range of the gateway

Figure 2.6: VANETCODE mechanism [3]

Advantages:

- No need of block selection and peer selection as compare to SPAWN protocol.
- No need to find who are the neighbours.
- No relay on routing protocol due to one-hop communication.

Future Scope:

- An important problem of peer-to-peer content sharing schemes is free riding, which can seriously degrade the performance.
- Incentive mechanisms play an important part in countering this.

- **CodeTorrent - Content Distribution using Network Coding in Vehicular Ad-Hoc Network:** [**18**] CodeTorrent is network coding based file swarming protocol. AP contains entire file at the beginning of the simulation. In CodeTorrent, a node which intends to share a file, a seed node, creates and broadcasts (at very low rate) to its 1-hop neighbour the description of the file. A description contains information about file like, Descriptor ID, Name, Size, number of pieces, File ID, etc. If node wants to share multiple files than it creates & broadcasts multiple file descriptors. Once any node receive descriptor of file and it found interesting than it sends request to the node. After receiving the request from other node, node send newly generated coded frame. To reduce the collision broadcast jitter is applied. It uses User Datagram Protocol (UDP) to transfer packets to its neighbour nodes. CodeTorrent does not use routing protocol, because it only depends on one hop unicast with overhearing capability.

  Advantages:

  - Shortens downloading time.

  Future Scope:

  - Simplest form, so different parameters and Optimization can be done.

- **Efficient Data Dissemination in Vehicular Ad Hoc Networks:** [4] In this method, vertical downloading with horizontal dissemination technique is used. It is push based model and only highway scenario is considered. File is downloaded from the infrastructure. Then source node transfer that file to next node in dissemination direction. Once next node after source node download file completely, handoff occur and that node transfer that data now.

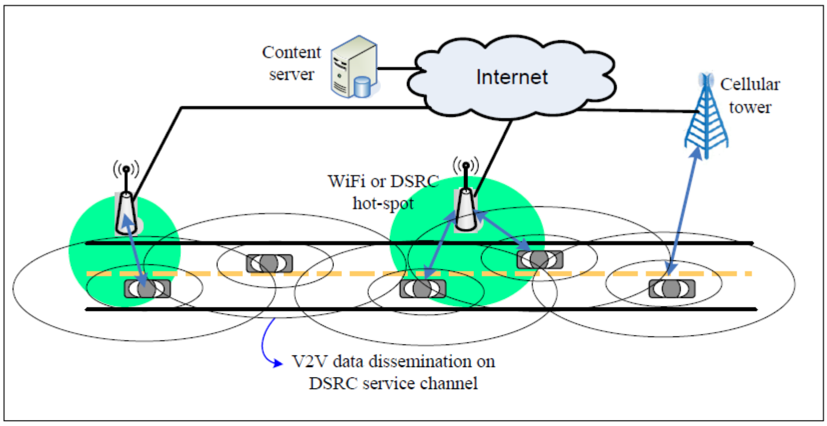


Figure: Vertical downloading with horizontal dissemination of data

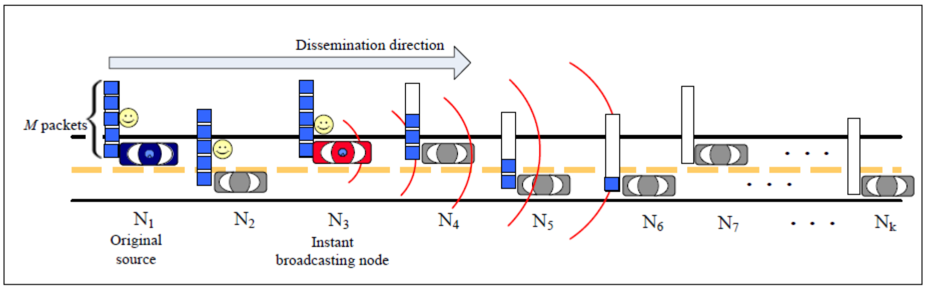


Figure: Data dissemination from one source vehicle to all others in the $+x$ direction in a 1-D VANET.

Figure 2.7: Efficient Data Dissemination in VANET mechanism [4]

Advantages:

– Network coding based broadcast (NC) is performed better than Random broadcast (RND) and Perfect feedback scheme.

Future Scope:

– Considering the factors that are assumed for improvement like, equal inter-vehicle spacing, corresponding to motion with identical velocity.

- **Abiding Regional Data Distribution using Relay and Random Network Coding on VANETs:** [5] In this method, it provides continuously location-dependent information generated by vehicles to other vehicles being driven near the birthplace of a data item with low data delivery traffic and short delays. In R2D2V (RNC-based Regional Data Distribution on VANETs), Hello message (contains ID, current time, current location, vehicles speed) is broadcasted by every vehicle. Using hello message vehicle maintains list of neighbouring vehicles and update also. Here data is divided into linear combinations of N divided data pieces than encoding is performed and broadcasts that encoded packets to the neighbour. If same neighbour and no new data then dont send encoded packets. Re-encoding is performed by intermitted node. Changing Hello message sending intervals. Changing the probability of replying to Hello messages in accordance with vehicle density.

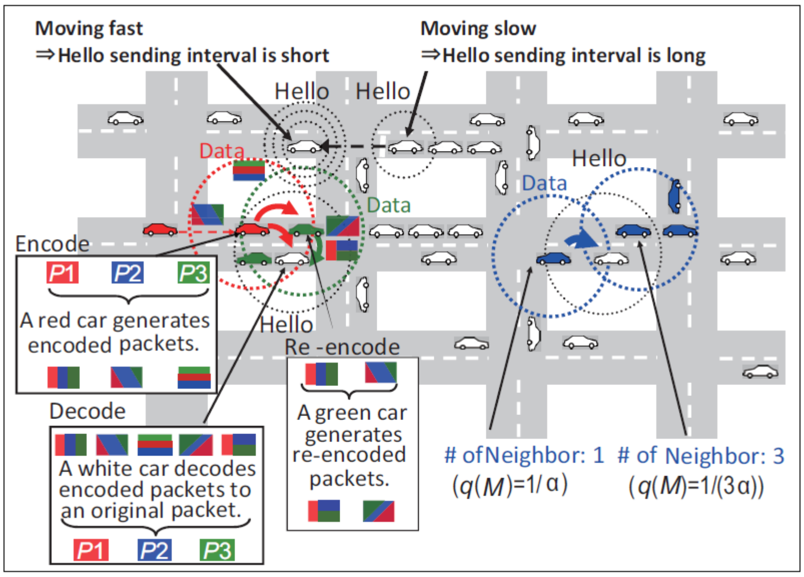


Figure 2.8: R2D2V mechanism [5]

Advantages:

- R2D2V can achieve a high data delivery ratio for vehicles driving near the birthplace of the information with low data delivery traffic and short delays when the vehicle density is high.

Future Scope:

- R2D2V cannot disseminate the data items rapidly to vehicles newly coming to the data effective area after the data item has been generated.

To overcome the problem of R2D2V, R2D2V-D and R2D2V-TD is proposed by authors [5]. In R2D2V-D, vehicles moving direction is also considered. When a vehicle holding a data item receives a Hello message from another vehicle, it checks whether its current position is closer to the birthplace of the data item than a Hello-sending vehicles current position. In R2D2V-TD, it uses vehicles moving direction and elapsed time for the same. The advantage of both above method is "Data also deliver to the newly admitted vehicle with low data delivery traffic and short delivery delay" and the future scope is scalability.

2.5 Data Dissemination Models

Three data dissemination models [14] [6] are same-direction, opposite-direction and Bi-direction.

a. **Same direction model**

In the this model, each vehicle broadcasts periodically both its generated and the store relayed data in the same data packet. Only vehicle in the same direction are responsible for broadcasting of data packets.

When a vehicle vehicle1 broadcasts a data packet, vehicle vehicle2 will accept this packet and propagate this packet later if :

(1) vehicle 1 and vehicle2 are in transmission range of each other, and

(2) vehicle 1 and vehicle2 are moving in the same direction, and

(3) vehicle1 is in front of vehicle2 with respect to their directions.

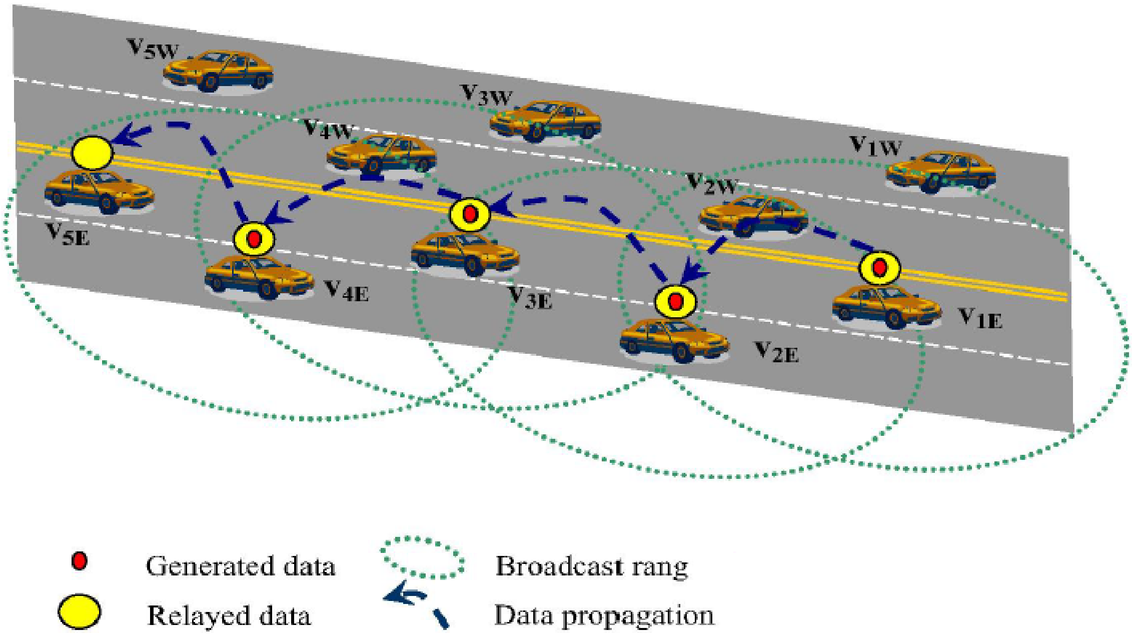


Figure 2.9: Data dissemination same direction model [6]

b. **Opposite direction model**

Here generated and relayed data are not broadcasted together. Instead of that, vehicles in same direction only broadcast its generated data.

Here these generated data are aggregated and propagated backwards by the vehicles in the opposite direction. When vehicle vehicle1 broadcasts a packet, vehicle2 will receive of this packet, if it is in transmission range of vehicle1, as given blow:

(1) If vehicle1 and vehicle2 are moving East direction, vehicle2 will accept packet if vehicle1 is in East of vehicle2. This is the case when vehicle1 broadcasts its own generated data.

(2) If vehicle1 and vehicle2 are moving West direction, vehicle2 will accept packet if vehicle2 is in West of vehicle1. This is the case when vehicle1 will relay a packet.

(3) If vehicle1 is moving East direction (or West direction) and vehicle2 is

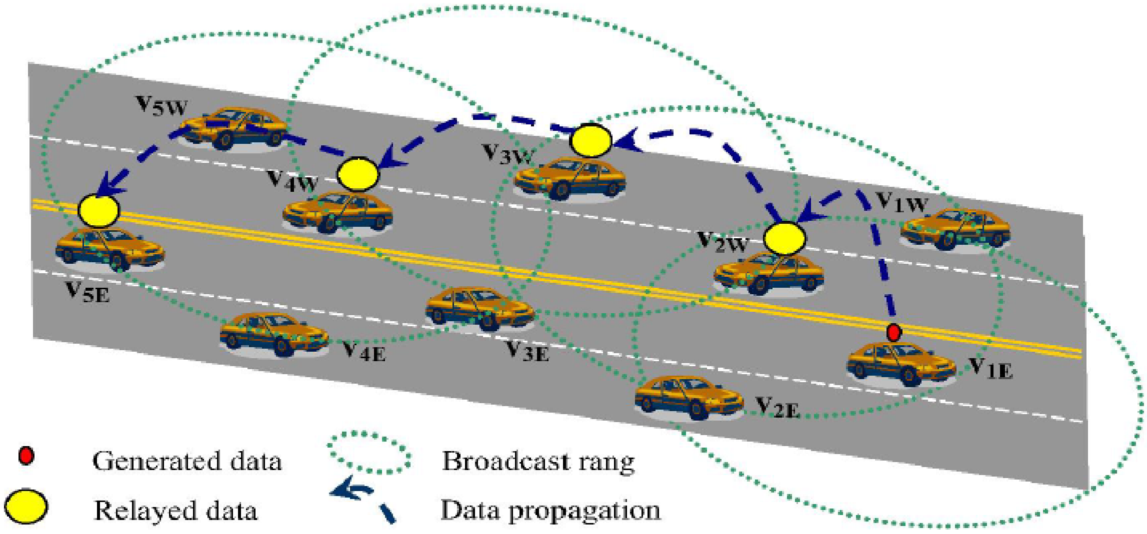


Figure 2.10: Data dissemination opposite direction model [6]

moving West direction (or East direction), vehicle2 will accept the packet according to relative position of the vehicles.

Note that the first rule gives guarantee of a fast delivery of the newly generated data to all the vehicles within single hop of source vehicle.

c. **Bi direction**

The bi-dir model combines both the same-dir and the opp-dir models. In bi direction model, propagation of generated and relayed data are in the same direction while propagates of relayed data are done among vehicles in the opposite direction. Information in the bi-direction model is propagated by vehicles moving in both directions.

2.6 Network Coding

There are two types of network coding [19] , **Linear Network Coding (LNC)** and **Random Linear Network Coding (RLNC)**. In traditional network, relay node or router simply forward the information packets destined to other node. In Linear

Network Coding (LNC), source or intermediate node allows to combine no of packets it has generated or received into one/several outgoing packets, where multiplication and addition are done over the field GF_2^s[20]. Linear combination does not mean concatenation, if we linearly combine packets where each packet has length of M, the resulting encoded packet has also size M.
In LNC , meaningful coefficients should be used for encoding and decoding of packets. LNC requires central authority to control generation of this meaningful coefficient. Algorithms employed for this should be centralized. But in wireless networks due to node's mobility and heterogeneity of network distributed approaches are suitable. So RLNC[21] [22] suggests the random generation of the encoding coefficient. In wireless networks, channels have a bigger error rate, higher interference between channels, unknown network topology. So protocols used in wireless networks should be optimized for above conditions. In RLNC, each node generates its own coding coefficient for each encoded packet. Also coefficients are sent to the destination in the packet header. So, the destination can decode the packet without knowing network topology or encoding rules, even if the topology is not fixed.

Number of successful transmission was measured for two cases in [23] , with and without random linear network coding. Simulation results indicates, there is an increase in number of successful transmissions for distance greater than 500 distance units in the case of random linear network coding.
For broadcasting there are no ACK to confirm the reception and we do not have mechanism to avoid congestion which decreases throughput. Results were collected for broadcasting scenario using network coding. Using random linear network coding the optimum result is obtained for a network with 150 nodes and congestion coefficient 0.026 (i.e. 4/150, where 146 nodes receive original packets out of 150 nodes, so 4 nodes with congestion). In RLNC, with increase in number of nodes, congestion decreases. In RLNC, for multicast scenario, probability that RLNC is valid is at least $(1 - d/q)^n$, where d is group size i.e. number of destination nodes, q=field size and n is number of links [19].

In LNC or RLNC there is higher packet delay as in that we have to delay the transmission of already arrived packets until additional packets have been collected. In opportunistic network coding, instead of selecting a particular node to be the next hop forwarder, nodes in the network coordinate with each other to select a multiple nodes which can potentially be served as next-hop forwarder. From multiple nodes, the node which is closest to the destination will forward the packet and other will drop the packets. In this scheme coordination amongst the nodes is required.

2.6.1 Variant of Network Coding

In this section, the survey of various variant of Network Coding [19] is presented with their results.

- **Partial Network Coding(PNCO) with Opportunistic Routing**
  In this scheme,source node breaks the information into n blocks of k packets and randomly mixes packets of same block before forwarding. When sender receives ACK of previously sent block from receiver, it sends next block. ACK will allow sender to send next block and hence reduces the delay. Sender also calculates the delivery probability on each link with expected cost metric and sends the forwarder list in the packet header. When forwarding nodes receive dependent packet it drops it. If it receives independent packets then packets are encoded again by using forwarders coefficient. When destination receives the encoded packets, it decodes them and gets the original packets. In above scenario, windowing scheme is used to stop retransmission of already sent packet by sender as well as to allow sender to send next packet.
  Resulsts shows that compare to path routing protocol, PNCO with opportunistic routing result in an increase in the achieved throughput. Over 90 percent of PNCO flows have throughput greater than 50 packets/sec. but path routing protocol is about 40 percent. Network coding can result in higher per-packet delay. The average delay of PNCO is higher, compared to path routing proto-

col. But it is reduced about 50 percent when compared to conventional network coding.

- **Cooperative Network Coding**

  In **Cooperative Network Coding** with cooperative communication different nodes collaboratively forward the information packets to exploit the spatial diversity. Participating nodes will be determined through upper layer protocols. Nodes in cooperative domain synchronized through synchronization technique e.g. GPS. There are two kinds of protocols for the forwarding nodes. i) Amplify-and-forward (AF) ii) decode-and-forward (DF). In AF mode, relay nodes transmit the received signal after some power normalization to amplify the information. In DF, relay nodes decode the received signal and then transmit it with its own encoding scheme to forward clean information, which reduces transmission rate. Here the information is exchanged in two phase. In phase I source transmits information to relays and in phase II relays will broadcast the combined signals of different sources to destination or another relay nodes.
  Results in [19] shows that in one relay system Decode-and-Forward outperforms the Amplify-and-Forward. However, in two relay the Amplify-and-Forward is better than Decode-and-Forward.

  Network coding allows mixing of various data packets. But in this scheme, packet has to wait to be coded with other packet until other packet arrives. It increase delay and loss rate.

- **Opportunistic Network Coding (ONC)**

  It is the approach in which whether packet is transmitted with or without network coding is decided by the status of the buffer's queue at a node. With network coding, number of packets transmitted by relay will decrease and hence increase the power efficiency of relay node. While in physical network coding, relay nodes are not decoding the received signal but it simply amplifies and

broadcasts the received signal and hence complexity of the relay node increased. This scheme is suitable for stationary traffic flow. For real time traffic, packet has to wait to network coded with other packet (delay). Also for finite buffer, packet loss rate will increased. To overcome these limitations, ONC can be used. When the probability of sending packets without encoding is fixed to be 0 then it reduces to conventional network coding. So conventional network coding is one of the cases of ONC.
Although the in term of delay and packet loss conventional network coding is not optimal. For ONC there is a delay-power tradeoff. Simulation results shows that ONC achieves lower delay compared to conventional network coding.

- In wireless digital broadcasting applications , base station (BS) broadcasts information to a user terminals (User equipment-UE) through wireless broadcasting channels. A received packet at User Equipments (UE) is either error-free or discarded as erroneous. UE will request BS to retransmit discarded packets which is automatic repeat request (ARQ) error control protocol. This strategy becomes inefficient as the number of UEs increases or number of packets increases. To improve system efficiency, use of network coding during the retransmission phase was suggested in [19].
  There are N information blocks sent by BS to $M >= 2$ UE. So there are M BS to UE block erasure channels are assumed. After the transmission of N information blocks to M UEs, each UE feeds back the indices of the lost or erased blocks. An error matrix E is generated by BS to record the block erasure status reported by UEs. The size of matrix E is MxN where ei,j =1 if the jth block is erased otherwise ei,j = 0.
  In retransmission phase the set of erased blocks is divided into subsets such that at least one erased block per UE is in any particular subset. The erased blocks in a subset are encoded into one encoded block for retransmission. In ARQ each erased block is retransmitted separately. Ex. Error matrix E for M=2 and

N=6 is as shown below.

$$E = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

As shown above with ARQ, blocks 1,2,3,4 and 6 retransmitted separately. With network coding only three blocks are retransmitted 1 xor 2, 3 xor 4, and 6. Now UE1 has received blocks 2, 3 and 5 correctly and UE2 has received blocks 1, 4 and 5 correctly in original broadcast. Now, in retransmission, UEs can retrieve the remaining blocks through simple modulo 2 additions.
Performance of proposed scheme is compared against traditional ARQ. In simulation in [19] the impact of N (Number of blocks), M (No of User Equipments) and erasure probability pi on the normalized overhead is considered two scenario, i) by keeping identical erasure rate on all links and erasure rate on link1,$p1 > pi$, ii) identical erasure rate on all other links. Proposed scheme can asymptotically achieve the lower bound on normalized overhead when the numbers of information blocks are sufficiently large.

2.7 Generation-by-Generation RLNC

In practical situation, the size of the matrices for which network coding take place has to be limited. It is difficult with random network coding. So for RLNC packets are grouped into generations [19]. Here the size of generation i.e. number of packets in one generation is fixed. Packets of same generation are encoded with each other as shown in Figure 2.11 [19]. Upon receiving encoded packets, intermediate node makes generations of received encoded packets destined to same destination node and encodes the packets of same generation using its own encoding vectors. Intermediate node sends effective coefficient generated using its own coefficients generated and received from sender along with encoded packets. This new effective coefficients received by receiver will help it to decode original packets sent by sender.

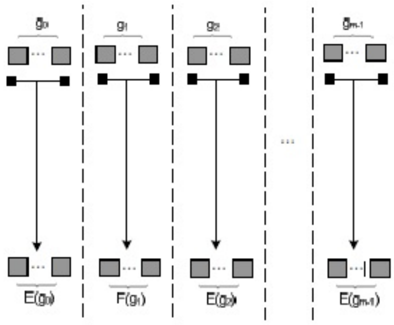


Figure 2.11: G-by-G Network Coding

As the number of originating packets (n) in a network for given destination increases, the amount of memory needed to store coefficients of encoded packets increases because these coefficients are to be remembered till a node receives at least n packets to decode all original packets. Further, till at least n encoded packets are not received, most of the n original packets can not be decoded. Thus delivery delay of a packet increases with n. To reduce memory requirement and the delay, originating packets can be grouped together into so called 'generation'. Now in the network, all nodes will encode packets of the same generation. If the generation size is k($k \ll n$) , whenever the receiver receives at least k packets of a generation, it will able to decode k original packets and the corresponding k encoded packets can be discarded, freeing the memory and the delay will also be reduced as receiver will have to now wait only for k encoded packets to be able to decode. But as packets from different generations can not be 'mixed', mixing opportunity reduces as k reduces.
Performance of Network Coding can be improved by increasing generation size (k) [19]. But on the other hand, increasing generation size after some threshold, increases overhead. In G-by-G Network coding where generation size is k, sender S generates atleast k encoded packets from k input packets of one generation. Here each generation is encoded and decoded separately of other. In G-by-G Network coding losses are expensive as the partial reception of encoded packets of same generation means a complete loss of that generation. To overcome the problem of losses, redundant packets can be sent with generation. Redundant packets enhance the reliability of

communication. But note that in G-by-G Network coding, extra packets sent with generation, protects that generation only.

2.8 Galois Field Arithmetic

The operators in conventional arithmetic (divide, multiply, add, subtract etc.) deal with infinite or unbounded numbers. Computer hardware uses binary arithmetic where integer results are stored in registers, bytes or words of a finite size. Dealing with the overflows caused by very large calculations is a considerable problem if the result is to be used in bit by bit error detection.

Reed Solomon codes are created by the manipulation of finite group of numbers called a 'Galois Field' [24]. GF(256) is a field consisting of the every integer in the range 0 to 255 arranged in a particular order. If you could devise an arithmetic where the result of each operation produces another number in the field the overflow issues could be avoided. The generation (ordering) of the field is key. e.g. a simple monotonic series from 0 to 255 is a finite field but modulo 255 arithmetic fails commutative tests i.e. certain operations will not reverse.

A Galois field gf(p) is the element 0 followed by the (p-1)succeeding powers of α : 0,1,α, α^1, α^2,...,α^{p-1}

Extending the gf(2) field used in binary arithmetic (and CRC calculation) to 256 elements that fit nicely in a computer byte: gf(2^8) = gf(256). Substituting the primitive element $\alpha = 2$ in the galois field it becomes 0, 1, 2, 4, 8, 16, and so on. This series is straightforward until elements greater than 127 are created. Doubling element values 128, 129, ..., 254 will violate the range by producing a result greater than 255. Some way must be devised to "fold" the results back into the finite field range without duplicating existing elements (this lets modulo 255 arithmetic out). This requires an irreducible primitive polynomial. "Irreducible" means it cannot be factored into smaller polynomials over the field. In our implementation irreducible polynomial 285 is used.

The Galois arithmetic operations for GF(256) must be implemented as follows:
The bitwise exclusive-or of two numbers is used for ADDITION and SUBTRACTION as mention below:

```
function addition(a,b)
begin
  result:=a xor b;
end;

function subtraction(a,b)
begin
  result:=a xor b;
end;
```

The **Multiplication** of two numbers are implemented by modulo operation of the two numbers as follows.

```
function multiplication(a,b)
begin
  result:={a(x) * b(x)} mod p(x);
end;
```

The **Division** of two numbers is a multiplication of dividend and multiplicative inverse of divisor. Ex. $5 \div 2 = 5 * \text{multiplicative inverse}(2)$

```
function multi_inverse(a)
begin
  result:=b such that [{a(x) * b(x)} mod p(x)] = 1;
end;
```

Chapter 3

Problem Definition

To develop a network coding based content distribution protocol to disseminate data in VANET.

3.1 Assumptions

- All the vehicles have onboard units.
- The network is Vehicular Ad-Hoc Network. Mobility of vehicles is independent with respect to each other.
- The network scenario is considered for dense network(city scenario).
- Road Side Unit (RSU) has files and is enabled with Internet facility.
- RSU can download the file depending on vehicle request and store it in cache.

3.2 Objective

- Efficient content distribution in VANET using Network Coding.

3.3 Intended Outcomes

- Design and evaluation of proposed mechanism and protocol.
- Comparison of simulation results for file delivery ratio of proposed protocol using conventional scheme and protocol using network coding.

Chapter 4

Simulation System

VANET Simulator software [7] are classified into three categories: Vehicular mobility generators, Network simulators, and VANET simulators.

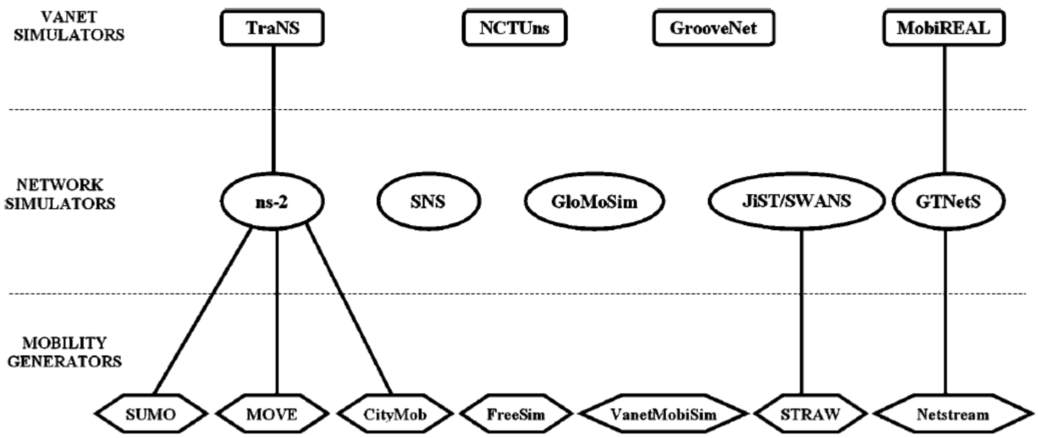


Figure 4.1: VANET Simulators [7]

- Vehicular mobility generators are needed to increase the level of realism in VANET simulations. Road model, scenario parameters are inputs of the mobility generator and the output of the trace details the location of each vehicle at every time instant for the entire simulation time and their mobility profiles. Examples are SUMO, MOVE and VanetMobiSim.

- Network simulators perform detailed packet-level simulation of source, destinations, data traffic transmission, reception, background load, route, links, and channels. Examples are ns-2, JiST/SWANS, and GloMoSim.

- VANET simulators provide both network simulation and traffic flow simulation. Examples are NCTUns, TraNS, and MobiREAL.

4.1 Mobility Generators for VANET

4.1.1 Simulation of Urban MObility

It [7] is an open source software. It is designed to handle large road networks. Its main features include different vehicle types, single-vehicle routing and dynamic routing, collision free vehicle movement , multi-lane streets, lane changing, hierarchy of junction types, an openGL graphical user interface (GUI).

Steps to generator mobility in SUMO:

a. Create a node.xml file.

b. Create a typ.xml file. Each description of an edge should include information about the number of lanes, the maximum speed allowed on this edge and optionally this edge's priority. To avoid explicit defining of each parameter for every edge, one can use edge types, which encapsulate these parameters under a given name.

c. Create an edge.xml file that provides connection between different nodes.

d. Create a net.xml file using NETCONVERT command. For example netconvert –node-files=hello.nod.xml –edge-files=hello.edg.xml –type-files=hello.typ.xml –output-file=hello.net.xml

e. Create a rou.xml file that contains details of vehicles, vehicle types and routes.

f. Create sumocfg file that contains input files and simulation parameters.

g. Open sumocfg file in SUMO simulator and run it.

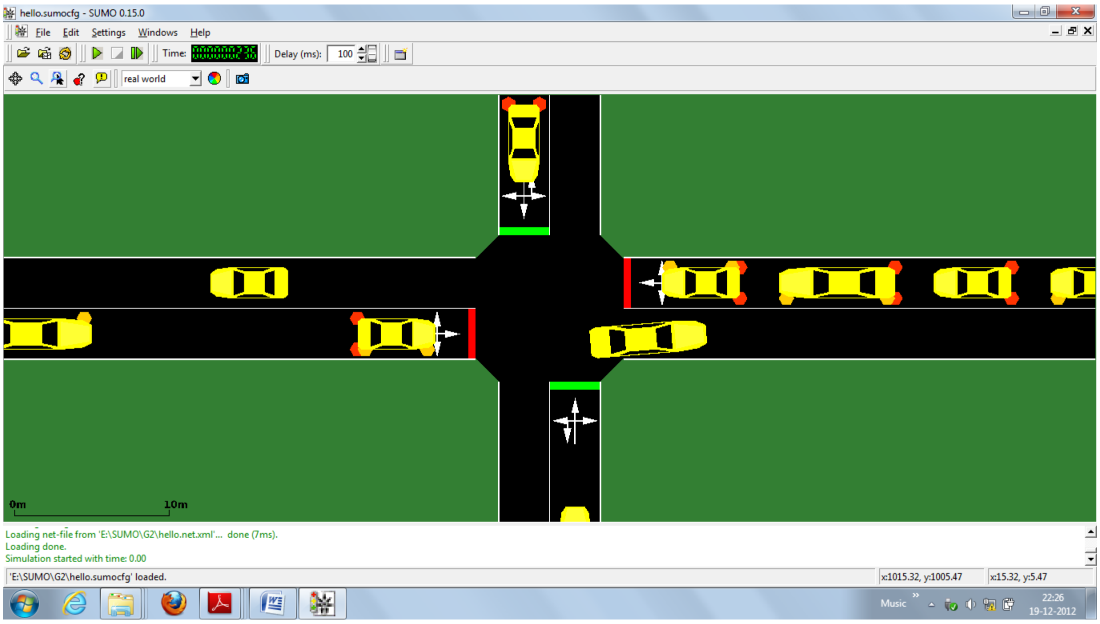


Figure 4.2: Mobility Generation using SUMO

Network Generation using NETGEN command: NETGEN allows to build three types of abstract networks: grid-networks, spider-networks and random networks.

Examples:

a. To generate Grid Network:

   netgen –grid –grid.x-number=5 –grid.y-number=5 –grid.y-length=40 –grid.x-length=200 –output-file=GridSumo.net.xml

b. To generate Spider Network:

   netgen –spider-net –spider-arm-number=10 –spider-circle-number=10 –spider-space-rad=100 –output-file=SpiderSumo.net.xml

c. To generate Random Network:

   netgen –random-net -o RandomSumo.net.xml –rand-iterations=200 –abs-rand

Importing the Road Network from OSM data file:

Command: netconvert –osm-files LargerAhmedabad.osm -o LargerAhmedabad.net.xml

4.1.2 MOVE (MObility model generator for Vehicular networks)

It rapidly generates realistic mobility models for VANET simulations [7]. MOVE is built on top of SUMO. The mobility trace file is the output of MOVE which can be used by network simulation tools immediately such as ns-2, GloMoSim, etc. In addition, MOVE provides a good GUI that allows user for quick scenarios generation without writing simulation scripts.

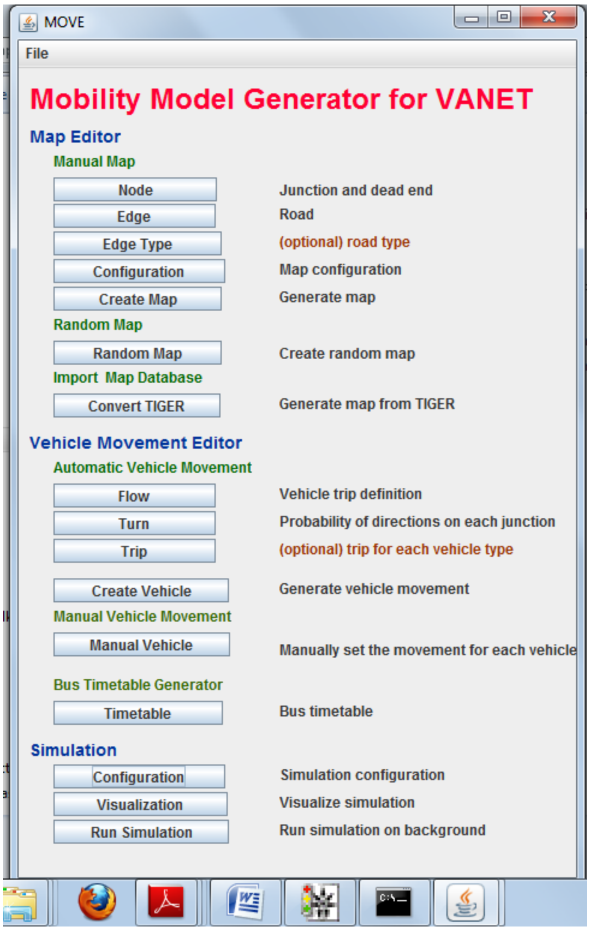


Figure 4.3: Mobility Generation using MOVE

Steps to generator mobility in MOVE:

a. Create a node.xml file using GUI.

b. Create a type.xml file using GUI.

c. Create an edge.xml file using GUI.

d. Create a net.xml file using Configuration option given in Create Map.

e. Create a rou.xml file that contains details of vehicles, vehicle types and routes.

f. Create sumo.cfg file that contains input files and simulation parameters.

g. Click on Visualization to run sumo.cfg file.

Steps to generator traffic model in MOVE:

a. Click on Traffic Model.

b. Click on Static Mobility.

c. Click in File then import MOVE Trace and .net.xml file for script generator.

d. Specify the file location for NAM and trace file.

e. Select File then Save or Save As (exNS.tcl).

f. Close Static Mobility window.

g. Then click on "Run NS-2".

h. Specify tcl script then click on Ok.

i. Close "Run NS-2".

j. To see visualization click on "Run Nam".

k. Specify NAM file then click on Ok.

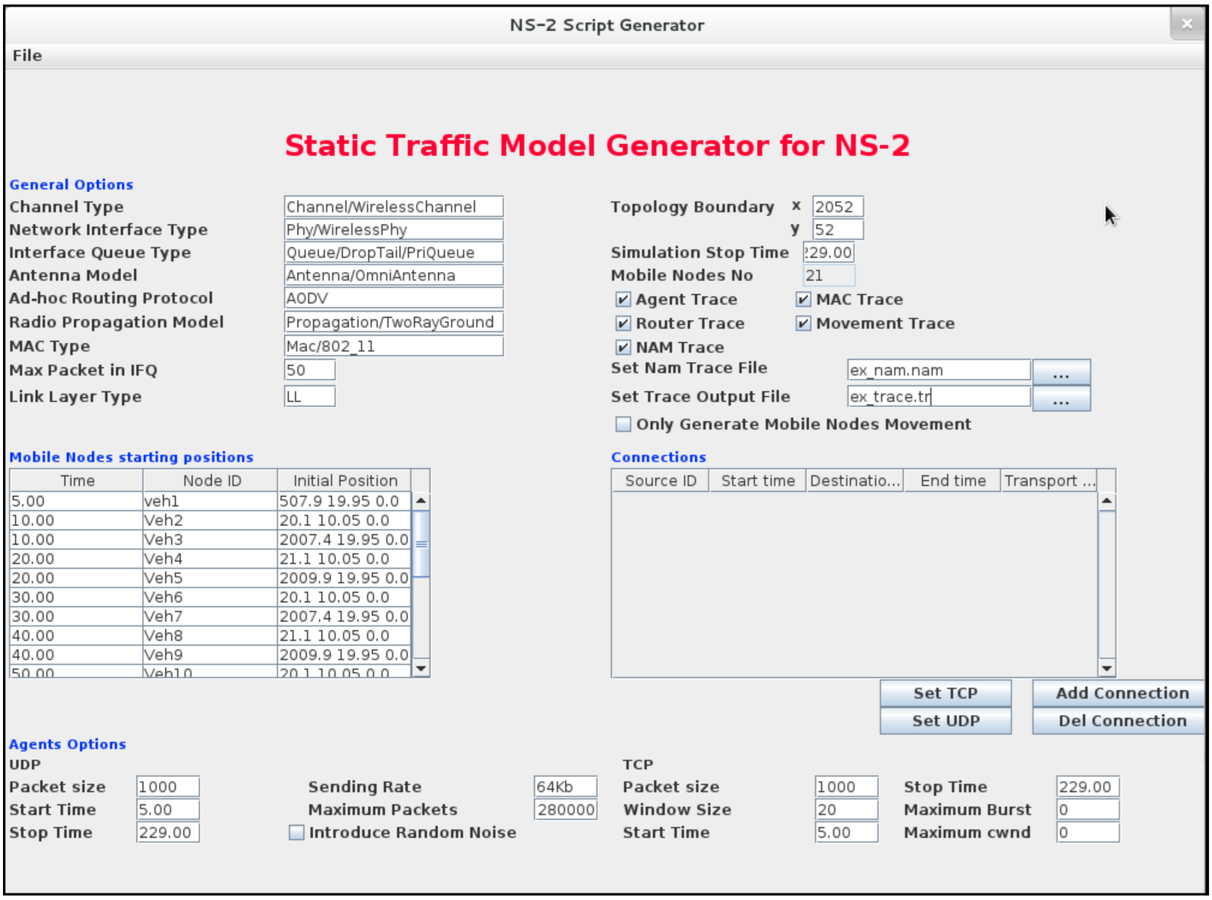


Figure 4.4: Traffic Model using MOVE

4.2 Network Simulators

4.2.1 NS-2

It is a discrete event simulator developed by the VINT project research group at the University of California at Berkeley [7]. The simulator was extended by the Monarch research group at Carnegie Mellon University to include: (a) node mobility, (b) a realistic physical layer with a radio propagation model, (c) radio network interfaces, and (d) the IEEE 802.11 Medium Access Control (MAC) protocol using the distributed coordination function (DCF).

Steps to install NS-2.34 in Fedora 17:

a. Download NS2.34.

b. Copy the file to /home/yourhome (in my case it is, /home/ajay/).

c. Open terminal.

d. Untar it using tar zxvf ns-allione-2.34.tar.gz.

e. cd ns-allione-2.34/

f. ./install

g. Once installation over, set the PATH.

4.2.2 Create a new agent in NS2

we are created a new agent in ns2. Follow the following steps to create a new agent in ns2: **Steps to create a new agent in ns2:**

a. Create a myagent.cc and myagent.h files.

b. Put both files in NS2.34 folder.

c. Add myagent object filename to the end of OBJECT CC list in Makefile.

d. Initialize packet size in ns-default.tcl.

e. Define the new packet type in packet.h.

f. Go NS2.34 folder.

g. Perform first make clean then make and make install.

h. Write MyAgent.tcl file to test newly created agent.

Chapter 5

Proposed solution

5.1 Overview

- In our protocol, data transfer takes place whenever two nodes (either RSU or OBU) come into communication range of each other. It is due to no of vehicles or speed of vehicles. So we consider network as an **opportunistic network**.
- In conventional content distribution protocol, node should forward packets such that destination node receive all required packets without receiving any redundant packet. In the **network coding (G-by-G NC)** based protocol, a node transmit coded packets that deliver all data packets to the destination with high probability. The coefficients required by the node can be generated independently by each node. Random Linear Coding is one such scheme in which coefficients are randomly decided by each node which increases overhead. RLNC does require coefficients to be carried with the packet but it does not significantly add to overhead.
- In our protocol we consider following configurable parameters:
  - The network parameter is Meeting Rate. Meeting rate of the network is the newly encountered vehicles in the network per second. It depends

on the area within which the network nodes move, their speed, mobility pattern and communication range.

– The protocol parameter is Generation Size. It denotes the size of the generation means packets are grouped into generations.

– The performance parameters of interest as mention previously are delay and delivery ratio.

5.2 Protocol Description

Based on the assumptions mentioned in 3.1, proposed solution is mention below:

- Vehicle will try to establish the connection with RSU (Road Side Unit) using HELLO message for neighbor discovery. If the vehicle successfully establish the connection with the RSU then infrastructure mode (V2I) will be enabled.

- If the OBU (On Board Unit) present in the vehicle does not receive the acknowledgement from RSU, the Ad-hoc mode (V2V) will be enabled.

- In infrastructure (V2I) mode,

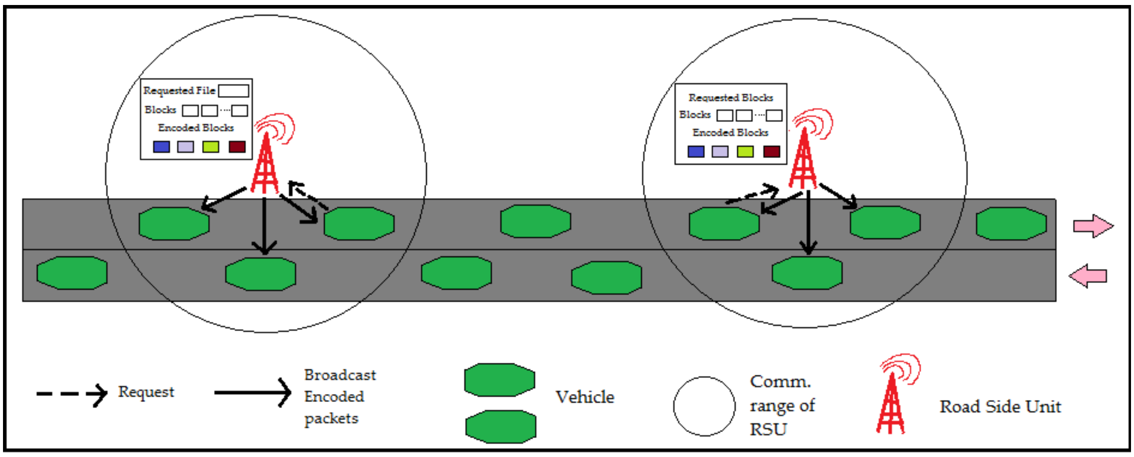


Figure 5.1: Infrastructure (V2I) Mode

a. Once RSU receives the request from vehicle for the specific file (new request) or specific blocks, RSU divides file into number of blocks and performs encoding of packets.

b. Then encoded blocks are broadcasts.

c. Vehicles receive that encoded packets for respective file and stored them (if innovative packet).

d. Only destination vehicle performs decoding of packets.

- In Ad-Hoc (V2V) mode,

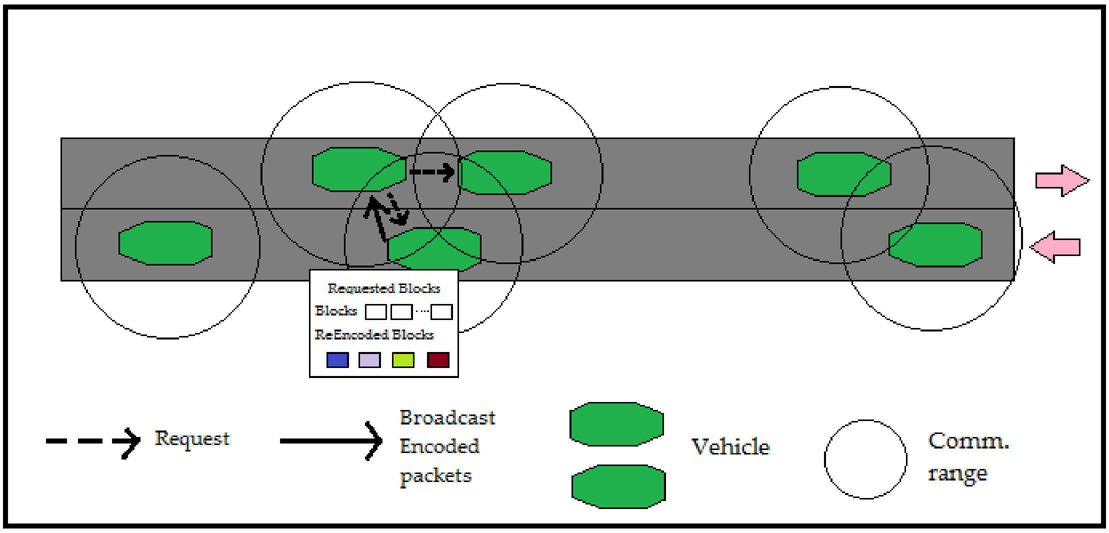


Figure 5.2: Infrastructure (V2V) Mode

a. Vehicle will send the request for specific blocks using HELLO message periodically for neighbor discovery.

b. If any neighbor vehicle has that file then neighbor vehicle divides file into number of blocks and performs encoding of packets and broadcast them.

c. Neighbor vehicles decode packets (if destination vehicle) and stored them (if innovative packets).

d. Overhearing is also done by the vehicles.

Main components of our protocol as mention in protocol are explained below:

- Neighbour discovery: *HELLO Packet* is transmitted by each node at fixed interval known as "Hello Broadcast Timer". Whenever a node receives HELLO packet from another node, it adds that node into its neighbor list if that node is not in the list already.
- If any node requires the file then send the *request* for the same. If any node contain file then divide it into blocks and encode it then broadcast it.
- Vehicles receive *replay* of encoded packets for respective file and stored them (if innovative packet).
- Only destination vehicle performs decoding of packets if particular generation packets are received.

Chapter 6

Results and Analysis

6.1 Simulation Parameters

All mobility scenarios are generated in MOVE which work on SUMO, which is open source software. We had simulated proposed protocol in NS (Network Simulator) 2.34. Simulation parameters are mentioned in the following table.

Parameters	**Simulated Values**
Antenna model	Omni-directional Antenna
Radio Propogation Model	TwoRayGround
MAC Type	IEEE 802.11p
Interface Queue Type	Priority Queue
Simulation Time	1250 second
No. of vehicles	100
Speed of vehicles	13.79 m/s, 15 m/s, 25 m/s
RSU communication range	250 m
OBU communication range	150 m
File Size	1MB to 12 MB

Source node (RSU / OBU) is sending packets to receiver node. For network coding, packets are grouped in the generations. For encoding of packets coefficients are chosen randomly and addition and multiplication operations are done over the finite field F_{2^8}.

6.2 Simulation Results

The network parameter of interest is meeting rate and number of RSUs (Road Side Units). The performance parameters of interest are file delivery ratio and file delivery delay. The protocol parameter is generation size.

We have compared our protocol with the protocol using conventional scheme which effectively disables the network coding and the protocol using network coding. We have compared file delivery ratio and file delivery delay of conventional scheme, protocol using network coding and our proposed protocol with different meeting rate, different file size and varying number of RSUs. We run the simulation for sufficient time such that each vehicle receives its requested file.

Graph in Figure 6.1 represents effect of changing network parameter i.e. Meeting Rate on performance parameter i.e. File Delivery Delay for G by G wise and Without Network Coding.

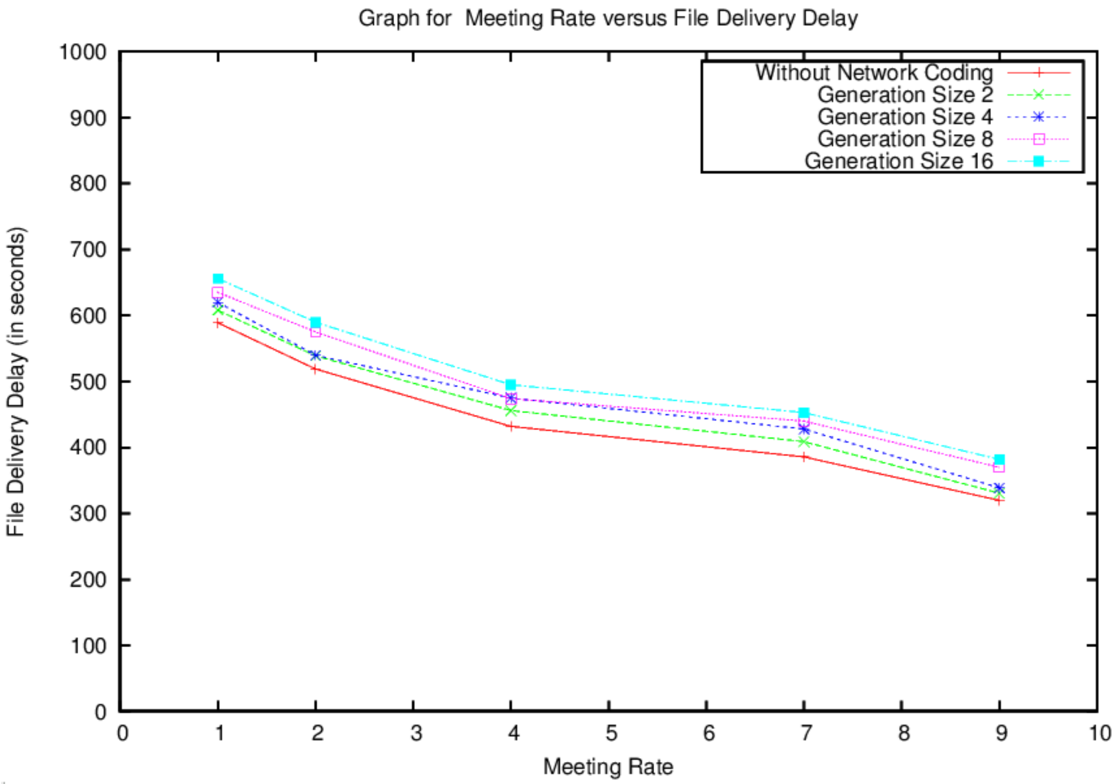


Figure 6.1: Meeting rate versus File Delivery Delay

We observe that for lower meeting rate the less chance of meeting a node with other node and hence number of packets to be forwarded by a node is less. As shown in figure 6.1 as meeting rate increase delay to deliver all sufficient packets of particular generation is decrease for all five cases. As shown in figure, network coding with generation size 2 and 4 are taking less delay to deliver packets compare to generation size 8 and 16. Delay increases because node has to wait until entire generation is decoded.

Graph in Figure 6.2 represents effect of changing network parameter i.e. Meeting Rate on performance parameter i.e. File Delivery Ratio for G by G wise and Without Network Coding.

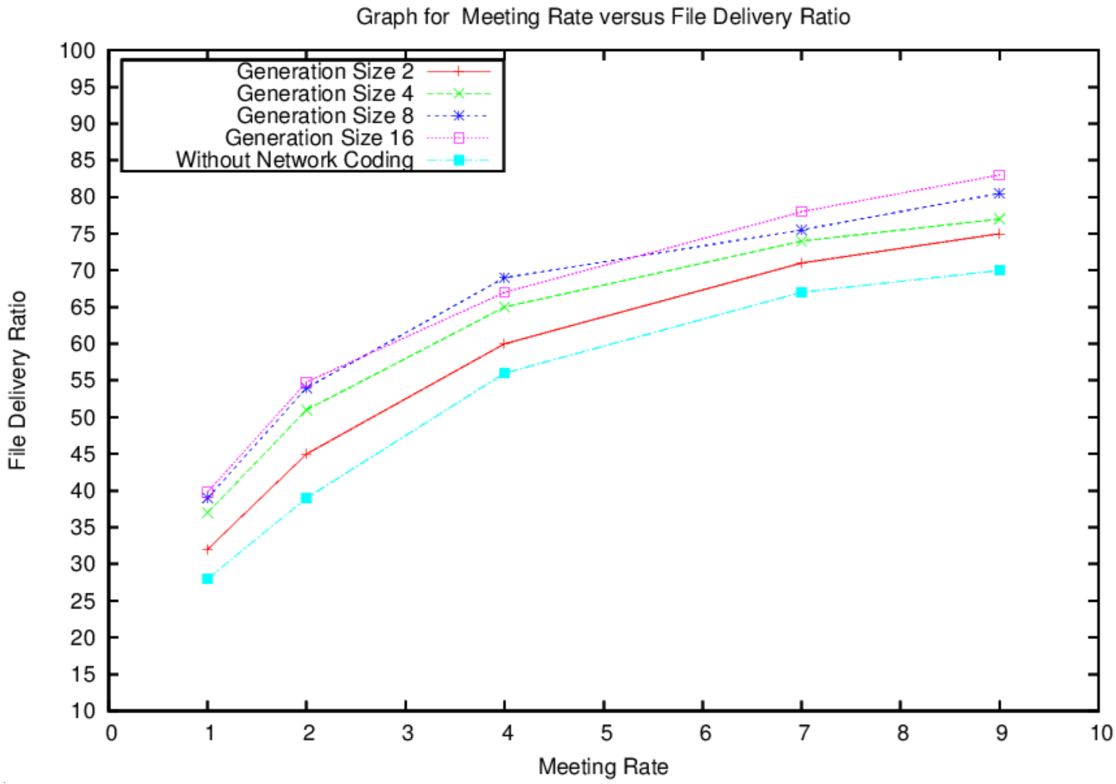


Figure 6.2: Meeting rate versus File Delivery Ratio

As meeting rate increases probability of delivering more number packets increases. As shown in graph as meeting rate is increased, average number of packets delivered is also increased in all five cases. But proposed protocol outperforms protocol using conventional scheme and as generation size increases, file delivery ratio increases.

Graph in Figure 6.3 represents file size versus File Delivery Delay for G by G wise and Without Network Coding. Simulation is carried out by taking various files (like MP3, AVI, TXT, TR, DOC, etc.) of different size. As shown in graph as file size is increased, delay is increased in all five cases. Result shows that if we increase the size of file, delay is higher in generation size 16 as compared to generation size 8. As mention previously, delay increases because node has to wait until entire generation is decoded. For small file size, node has very less contact to download entire file. So delay is increased in file size, it does not increase very much. But after that file size increases delay increases and rate of increment in delay gradually slows down with increase in file size.

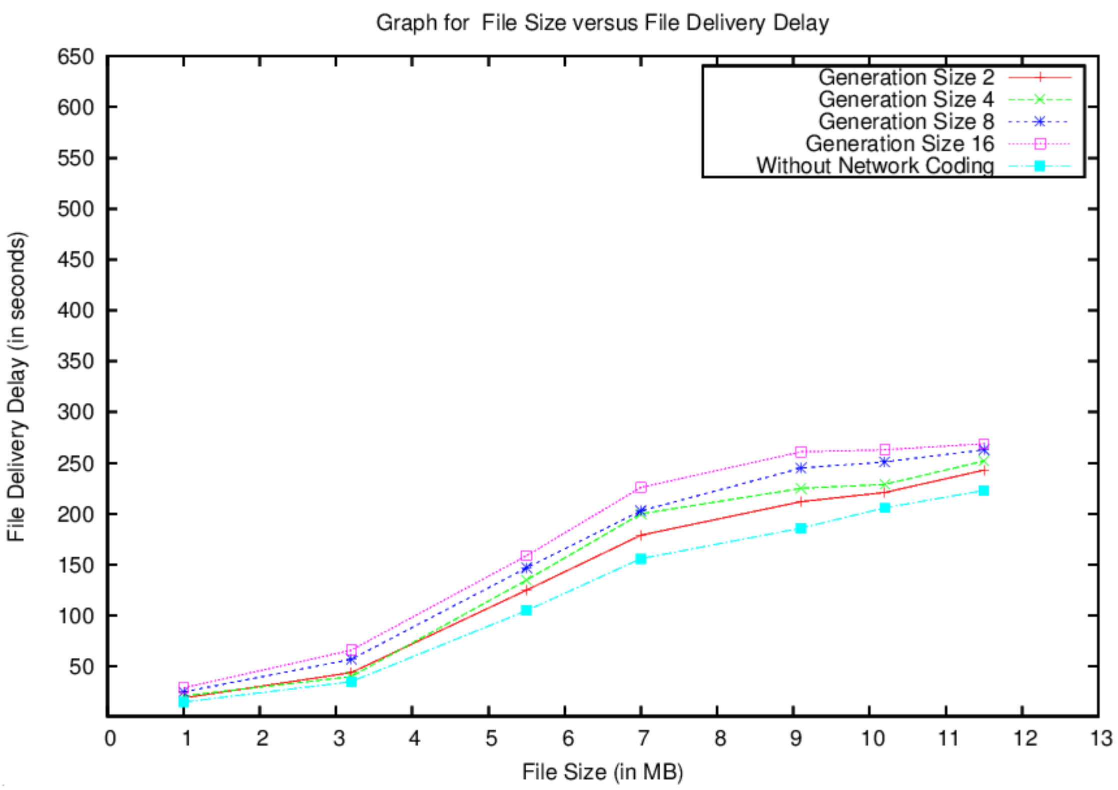


Figure 6.3: File Size versus File Delivery Delay

Graph in Figure 6.4 represents effect of Number of RSUs on performance parameter File Delivery Delay for G by G wise and Without Network Coding.

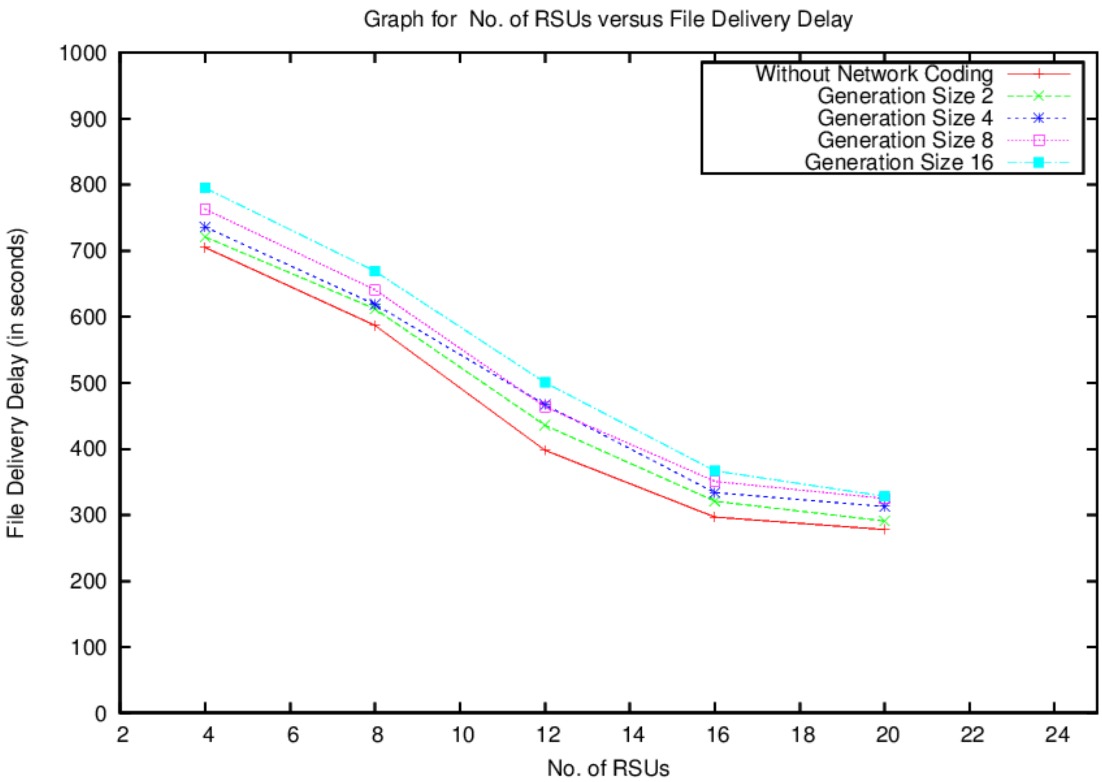


Figure 6.4: Number of RSUs versus File Delivery Delay

Here we vary the number of RSU from 4 to 20. We measure the File Delivery Delay for different number of RSUs. As shown in graph, as number of RSUs increases delivery delay decreases but if we increase number of RSU beyond certain limit (i.e 16), the change in delivery delay is negligible. It is because almost entire area is covered by RSUs (i.e 16).

Graph in Figure 6.5 represents effect of Number of RSUs on performance parameter File Delivery Ratio for G by G wise and Without Network Coding.

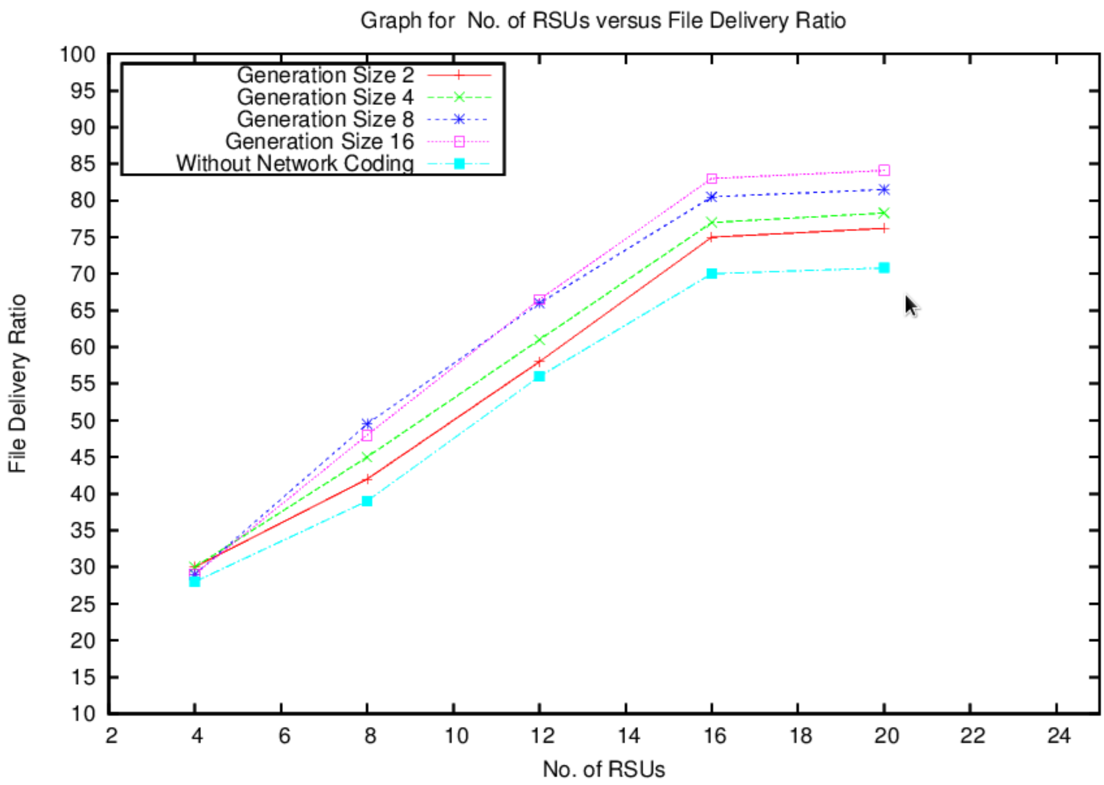


Figure 6.5: Number of RSUs versus File Delivery Ratio

Here we vary the number of RSU from 4 to 20. We measure the File Delivery Ratio for different number of RSUs. As mention previously, as number of RSUs increases delivery ratio increases. Result shows the protocol using network coding gives better performance as compare to protocol without using network coding. It is also derived from graph that it number of RSUs are increased beyond certain limit (i.e. 16), increase in delivery ratio is very little. It is because almost entire area is covered by RSUs (i.e 16). As generation size increases performance in terms of file delivery ratio increases.

Chapter 7

Conclusion

Inherent VANET characteristics make data dissemination quite challenging for different type of applications and scenarios.
Major contributions of our work are as follows:

- Implementation of G-by-G Network coding.
- Protocol design for Infrastructure mode and Ad-hoc mode for content distribution using network coding and implemented the same.
- Comparison of conventional scheme and the scheme using network coding with proposed protocol through simulation.

Summary of major findings of our work are as under:

- Simulation results show that Generation by Generation Random Linear Network Coding (G-by-G RLNC) increases the delivery ratio from 10% to 15% as compared to conventional scheme without using network coding.
- Improvement of the protocol over conventional scheme and scheme using network coding has been observed for delivery ratio as number of RSUs increases.

7.1 Future Work

The protocol using network coding outperforms conventional scheme in terms of delivery ratio. Consideration of priority of data in protocol, like safety messages must be delivered first then the entertainment data. We intend to introduce purging scheme to utilize finite buffer efficiently.

References

[1] M. Saxena, "Network coding: Theory and reality," tech. rep.

[2] M. Khatri and S. Malhotra, "An insight overview of issues and challenges in vehicular adhoc network," tech. rep., Journal of Global Research in Computer Science, Volume 2, No. 12, pp. 47-50, 2011.

[3] S. S. K. Shabbir Ahmed, "Vanetcode: Network coding to enhance cooperative downloading in vehicular ad-hoc networks," tech. rep., IWCMC'06.

[4] H. W. Fei Ye, Sumit Roy, "Efficient data dissemination in vehicular ad hoc networks," tech. rep., IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 30, NO. 4.

[5] N. Kusumine and S. Ishihara, "Abiding regional data distribution using relay and random network coding on vanets," tech. rep., 26th IEEE International Conference on Advanced Information Networking and Applications, pp. 105-112, 2012.

[6] P. S. Tamer Nadeem and L. Iftode, "A comparative study of data dissemination models for vanets," tech. rep.

[7] J.-C. C. C. T. C. P. M. Francisco J. Martinez1, Chai Keong Toh, "A survey and comparative study of simulators for vehicular ad hoc networks (vanets)," tech. rep., Wireless Communications And Mobile Computing, 2009.

[8] Y.-S. C. YUN-WEI LIN and S.-L. LEE, "Routing protocols in vehicular ad hoc networks: A survey and future perspectives," tech. rep., JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 26, 913-932, 2010.

[9] B. R. Q. Mun J. Lok and J. M. H. Elmirghani, "Characterisation of data dissemination in vehicular ad hoc networks in a city environment," tech. rep.

[10] S. Kuklinski and G. Wolny, "Caravan: A context-aware architecture for vanet," tech. rep., Mobile Ad-Hoc Networks: Applications, Prof. Xin Wang (Ed.), ISBN: 978-953-307-416-0, InTech, pp. 105-112, 2011.

[11] `VANET_INFO.,http://www.vanet.info/projects`.

[12] T. L. Elmar Schoch, Frank Kargl and M. Weber, “Communication patterns in vanets,” tech. rep.

[13] J. Guo and N. Balon, “Vehicular ad hoc networks and dedicated short-range communication,” tech. rep., 2006.

[14] R. Karim, “Vanet: Superior system for content distribution in vehicular network applications,” tech. rep.

[15] D. Sutariya and D. S. N. Pradhan, “Data dissemination techniques in vehicular ad hoc network,” tech. rep., International Journal of Computer Applications (0975 8887), Volume 8 No.10, pp. 35-39, 2010.

[16] B. K. C. Pratibha Tomar and G. S. Tomar, “State of the art of data dissemination in vanets,” tech. rep., International Journal of Computer Theory and Engineering, Vol.2, No.6, pp. 957-962, 2010.

[17] S. Rajwani, “Content dissemination mechanisms in vanets: A comparative study,” tech. rep.

[18] J. Y. G. P. M. G. Uichin Lee, JoonSang Park, “Codetorrent: Content distribution using network coding in vanet,” tech. rep., MobiShare’06, 2006.

[19] A. P. Jitendra Bhatia and Z. Narmawala, “Review on variants of network coding in wireless ad-hoc networks,” tech. rep., INTERNATIONAL CONFERENCE ON CURRENT TRENDS IN TECHNOLOGY, NUiCONE 2011, 2011.

[20] C. Fragouli and E. Soljanin, “Network coding applications,” tech. rep., Foundations and Trends in Networking Vol. No.2, 2007.

[21] J.-Y. L. B. Christina Fragouli, “Network coding: An instant primer,” tech. rep., ACM SIGCOMM Computer Communication Review, Volume 36, January 2006.

[22] F. H. Mohsen Sardari and F. Fekri, “Infocast: A new paradigm for collaborative content distribution from roadside units to vehicular networks,” tech. rep., IEEE Secon, 2009.

[23] W. SONG and M. LI, “Application for the random network coding and models for random linear network coding,” tech. rep., International Conference on Computer Science and Network Technology, pp. 2533-2537, 2011.

[24] `REDTITAN_Technology_Ltd.,http://www.pclviewer.com/rs2/galois.html`.