

Multipoint Relay flooding

Network coding improvements

WANNING ZHU



**KTH Information and
Communication Technology**

Master of Science Thesis
Stockholm, Sweden 2009

TRITA-ICT-EX-2009:5

Multipoint Relay flooding: Network coding improvements

Wanning Zhu

2009-04-21

Supervisor: Anders Hansson, FOI

Abstract

Network coding is a new research area with potential to reduce network resources. With network coding, intermediate nodes forward packets that are linear combinations of previously received packets.

The type of networks we consider are vehicle mounted mobile tactical radio networks. Tactical communication may be required in areas where pre-deployed base stations are unavailable. Mobile ad hoc networks satisfy this requirement. Due to low antenna heights, network resources are scarce in mobile networks without base stations. Broadcast traffic, which disseminates information network-wide, is very important in tactical mobile networks. Multipoint relay flooding is a well-known technique for efficient distribution of broadcast traffic. It is therefore interesting to evaluate how much network coding can reduce the number of transmissions in a mobile ad hoc network. In this thesis we show that it is possible to further reduce the number of transmissions for Multipoints Relay flooding by using network coding. However, this improvement is largely theoretical - there has been no effort to evaluate this approach in practice.

Contents

1	Introduction	1
1.1	Background	1
1.2	Problem Definition	1
1.3	Outline	1
2	Mobile ad hoc networks	3
2.1	Introduction to MANETs	3
2.2	Ad hoc routing protocols	3
2.3	Optimized Link State Routing Protocol	4
2.4	Different types of traffic	5
2.5	Medium Access Control	5
3	Broadcast traffic	6
3.1	Multipoint Relays	6
3.2	Multipoint relay flooding	7
4	Network Coding	8
4.1	Linear Network Coding	8
4.2	Encoding	10
4.3	Decoding	10
4.4	Advantages and disadvantages	11
5	Local network coding improvement for MPR flooding	12
5.1	Main idea	12
5.2	Transmission reduction	12
5.3	Example	13
6	Simulations	15
6.1	The Simulator	15
6.2	Assumptions for the simulations	15
6.3	Simulation setup	15
7	Results	17
7.1	Graphic results	17
7.2	Compare the simulation results	17
7.3	Comments	23

8	Conclusions and future work	26
8.1	Conclusions	26
8.2	Future work	26
	References	26

List of Figures

2.1	A simple MANET with 3 nodes, where the node B acts as a router . . .	4
3.1	Traditional Flooding (a) and MPR Flooding (b)	7
4.1	Linear network coding at a relay node	8
4.2	Traditional method (a) and versus network coding (b)	9
4.3	Example of decoding	10
5.1	Transmissions in MPR flooding when the MPR nodes use network coding	14
7.1	traffic reduction, network coding be used in a rectangle network with 10 nodes	18
7.2	traffic reduction, network coding be used in a square network with 10 nodes	18
7.3	traffic reduction, network coding be used in a rectangle network with 20 nodes	19
7.4	traffic reduction, network coding be used in a square network with 20 nodes	19
7.5	traffic reduction, network coding be used in a rectangle network with 40 nodes	20
7.6	traffic reduction, network coding be used in a square network with 40 nodes	20
7.7	traffic reduction, network coding be used in a rectangle network with 80 nodes	21
7.8	traffic reduction, network coding be used in a square network with 80 nodes	21
7.9	traffic reduction, network coding be used in a rectangle network with 160 nodes	22
7.10	traffic reduction, network coding be used in a square network with 160 nodes	22
7.11	'Bottleneck' distributed nodes in network with one MPR node	23
7.12	Example network which gives best result on 'G'	24
7.13	Special network	25

Chapter 1

Introduction

1.1 Background

The type of networks we consider are vehicle mounted tactical radio networks. Tactical communication may be required in areas where pre-deployed base stations are unavailable. Mobile ad hoc networks (MANETs) satisfy this requirement. MANETs is an important technology for military communications. MANETs enable a group of nodes to form a communication network without requiring infrastructure components, such as base stations and fixed power sources[1].

Broadcasting is a common operation in a network. In a tactical military MANET, due to host mobility, such operations are expected to be executed more frequently (such as group voice calls, status information exchanging, orders). Therefore, efficient support for group broadcast is critical for these networks. The use of Multipoint Relay (MPR) flooding significantly reduces the number of retransmissions of broadcast messages. This thesis will examine if broadcasts can be further improved via network coding.

1.2 Problem Definition

Network coding is a new research area that may have interesting applications in real networks. Network coding is based upon intermediate nodes combining packets before forwarding. Thus network coding enables better resource utilization and can achieve the max-flow (a theoretical upper bound of network resource utilization [14]). In this project, we will use this advantage of network coding to study if it is possible to improve MPR flooding by using network coding. Specifically we determine the resulting reductions in the number of transmissions in the network.

1.3 Outline

This thesis starts with an overview of MANET technology in Chapter 2, broadcast traffic in Chapter 3, and network coding in Chapter 4. Chapter 5 presents the main idea and models that will be explored in the rest of the thesis. Chapter 6 describes the simulations and the assumptions that were made to implement network coding for

MPR flooding. The simulation results and special cases are presented in Chapter 7. Finally in Chapter 8, conclusions are drawn and some future work is presented.

Chapter 2

Mobile ad hoc networks

Most wireless infrastructure based networks utilize a single hop radio connection between a node and the wired network. In such networks, performance analysis can be done in terms of this single radio link. However, in this report we are concerned with multihop wireless networks, specifically with mobile ad hoc networks (MANETs). Thus we need to examine the effects that occur because of these multiple hops. In addition, we look to see if there are advantages of multiple hops that we can exploit.

2.1 Introduction to MANETs

A MANET is a decentralized network that utilizes self organization of multihop communication between potentially moving nodes (which do not necessarily move in a coordinated fashion). Hence the set of nodes participating in such a network and the network's topology may change over the course of time and space.

A MANET is composed of a group of nodes. All of the nodes in the network can transmit and receive data and also relay data. Thus all nodes are both hosts and routers. It is the latter property which enables multihop routes through the MANET. In addition, one or more nodes need to be listening for communications while at least one node attempts to transmit. For the remainder of this report we will assume that all nodes have a single kind of radio and that all can communicate directly if and only if they are within communication range. Thus we have explicitly assumed that nodes that are not transmitting are listening [2].

Figure 2.1 shows a simple MANET of three nodes A , B , C , where the middle node B acts as a router allowing communication between nodes A and C . Node B not only has to forward traffic between A and C , but node B must also deal with the problems that occur because nodes A and C can not hear each other.

2.2 Ad hoc routing protocols

The technique of finding, maintaining, and utilizing multihop paths is called routing. An *ad hoc* network of mobile nodes requires a routing protocol that can deal with the changes in topology that node mobility may cause. The network should be self-organizing and the routing decisions should be made in a decentralized fashion. By adopting this self-organization and decentralized model, the network can adapt to both

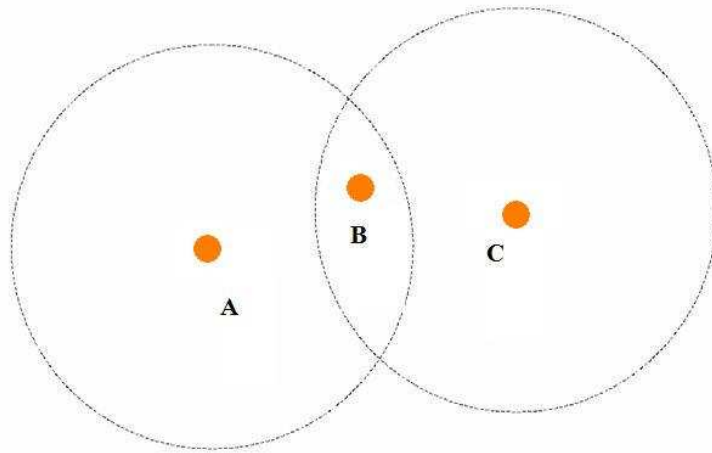


Figure 2.1: A simple MANET with 3 nodes, where the node B acts as a router

the arrival of new nodes and the departure (including departure due to failure) of nodes. To solve the problems that can occur in *ad hoc* routing, a large number of routing protocols have been proposed. Protocols for *ad hoc* networks are often divided into the two groups reactive and pro-active.

A reactive routing protocol only updates a route when it is necessary. An example of a reactive routing protocol is Dynamic source routing (DSR). When a packet should be forwarded and no route is available at the node, then a search-process is started to find a suitable path [11].

In contrast, a pro-active routing protocol continuously tries to update the routes in the network. An example of such a pro-active routing protocol is Optimized Link State Routing Protocol (OLSR), see Section 2.3. Because a pro-active routing protocol always maintains a full set of routes, when a packet should be sent the route is already known and can be used at once.

We assume that the pro-active routing is best for the type of networks that we consider, since some of the services will require continuously updated routes. Broadcasting of status information among the nodes is one such example.

2.3 Optimized Link State Routing Protocol

The Optimized Link State Routing Protocol (OLSR) [8] is an IP routing protocol which is optimized for mobile *ad hoc* networks, but can also be used on other wireless *ad hoc* networks. OLSR makes use of "Hello" messages to find its one hop neighbors. It finds its two hop neighbors through its neighbors' responses. The sender can then select its multipoint relays (MPR) (see section 3.2) based on the first hop node which offers the best routes to the second hop nodes. Each node has also an MPR selector set which enumerates nodes that have selected it as an MPR node. OLSR uses Topology Control (TC) messages along with MPR forwarding to disseminate neighbor information throughout the network.

2.4 Different types of traffic

There are several different ways of addressing and transmitting a message over a network. One way in which messages are differentiated is in how they are addressed, more specifically to how many receivers the message is addressed. Which method is used depends on what the function of the message is, and also on whether or not the sender knows specifically whom they are trying to contact, or only generally knows whom the message is intended for.

- **Broadcast** is used when a single node wishes to transmitting a message to all other nodes which are connected to the network. The goal is that when a node transmits a 'broadcast' packet that all the other nodes in the network will receive that packet.
- **Unicast** is used when packets to be sent from one sender to one receiver.
- **Multicast** is a very different from both unicast and broadcast. A multicast transmission is a type of transmission in which information is sent to a set of receivers. Note that there might not be any receiver that is currently interested in receiving this multicast. Additionally, it should be noted that more than one node can be sending a packet to a given multicast address at any point in time. Thus a multicast destination address can be viewed as a logical destination of zero or more recipients.

2.5 Medium Access Control

Medium Access Control (MAC) is a sublayer of the data link layer specified in the seven-layer ISO model. It provides addressing and channel access control mechanisms that make it possible for several interfaces to communicate within a multipoint network, typically a local area network (LAN) or metropolitan area network (MAN). In the case of a shared channel, the MAC layer emulates a full-duplex logical communication channel in a multipoint network. This channel may provide unicast, multicast, or broadcast communication service [9].

When more than one radio must share the same channel resources (be it time, frequency, code, ...), we need a medium access protocol to manage the transmissions in order to avoid collisions and to efficiently utilize the available bandwidth. In this thesis project, we will assume that our systems use time division multiple access (TDMA) communication.

TDMA is a collision-free MAC protocol where the channel sharing is done in the time domain. This means that the time is divided into time slots and each node is assigned one or more time slots when it is allowed to use the channel [10].

Chapter 3

Broadcast traffic

Broadcast traffic is often used to disseminate information to all nodes. Another important use of broadcast is to find unicast routes in ad hoc networks. Because all nodes will be required to receive or to retransmit, hence all nodes will have resources consumed, thus broadcast efficiency is very important. Due to the potentially dynamic nature of ad hoc networks, Multipoint Relay algorithms (see section 3.2) are much more robust and effective, i.e., fewer messages are required as they have less maintenance overhead.

3.1 Multipoint Relays

In MANETs, packets can be forwarded on the same interface that it arrived on. Instead of pure flooding where all nodes retransmit all packets, with Multipoint Relays (MPR) packets are forwarded only by the node's MPRs in order to reduce the number of transmissions that are needed to successfully deliver the packets[4]. A MPR set is a subset of a node's one-hop neighbors, such that together this subset are able to reach all the two-hop neighbors [6],[8]. In order to calculate the MPRset, the node must have link state information about all one-hop and two-hop neighbors.

Let $N_1(u)$ denote the set of one-hop neighbors of u , and $N_2(u)$ denote the set of 2nd-hop neighbors of u .

1. Start with an empty MPR set $MPR(u)$.
2. Select those one-hop neighbor nodes in $N_1(u)$ as multipoint relays which are the only neighbor of some node in $N_2(u)$, then add these one-hop neighbor nodes to the multipoint relay set $MPR(u)$.
3. While there still exist some nodes in $N_2(u)$ which are not covered by the multipoint relay set $MPR(u)$:
 - For each node in $N_1(u)$ not in $MPR(u)$ compute the number of the nodes that it covers among the uncovered nodes in the set $N_2(u)$.
 - Add that node of $N_1(u)$ in $MPR(u)$ for which this number is maximum.

In this work, we add the following modification to step 3: If there are more than one first hop neighbors covering the same number of second neighbors, add the one with

more neighbors (regardless of whether they are already be covered or not). The reason for this modification is that it increases the probability that several nodes choose the same node as a MPR.

3.2 Multipoint relay flooding

Multipoint relay flooding is a broadcast mechanism used in the ad hoc routing protocol OLSR. The principle is that each node has computed a multipoint relay set, and only these selected neighbors, will retransmit a packet broadcasted by the node. Obviously, the smaller this set is, the more efficient the mechanism will be (i.e., the greater the optimization).

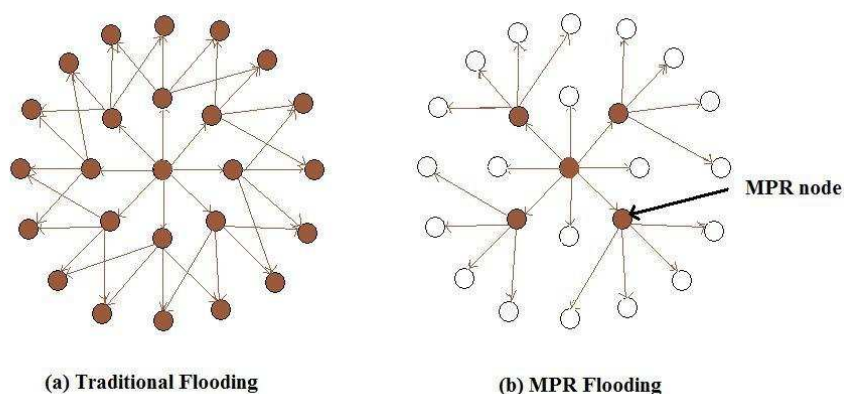


Figure 3.1: Traditional Flooding (a) and MPR Flooding (b)

Figure 3.1 shows both a traditional flooding algorithm and the MPR flooding algorithm. Here we see that there is a reduction in the number of transmissions by using MPR flooding [7].

1. Using the traditional flooding:
 - A source node u broadcasts message M .
 - Each node v that receives the message forwards M unless it has been previously forwarded.

2. Using MPRs for flooding leads to scoped flooding. In this case:
 - A source node u broadcasts its message M .
 - Each node v that receives M re-broadcasts it only if:
 - (a) v is a multipoint relay of the previous hop of the message;
 - (b) the message was not previously recieved by v .

Chapter 4

Network Coding

The main idea underlying network coding is that instead of simply forwarding data packets, the intermediate nodes form linear combinations of the previously received packets to recombine into new packets that will be forwarded [3]. In this chapter, we will introduce what network coding does and how it operates by studying an example in the following sections.

4.1 Linear Network Coding

In our example network, each node combines a number of incoming data packets and create one or more outgoing packets by using network coding.

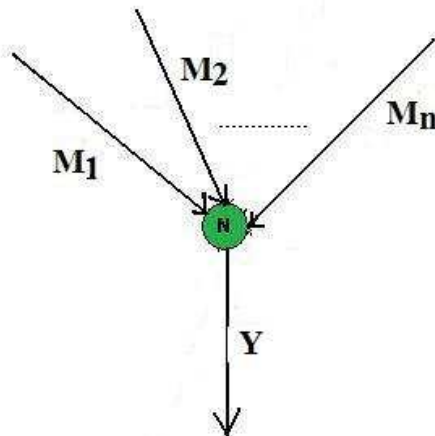


Figure 4.1: Linear network coding at a relay node

This example (shown in Figure 4.1) gives a more formal introduction to linear network coding:

- $M_1, M_2, \dots, M_i, \dots, M_n$, are incoming decoded data packets that have arrived at node N .

- $g^j = g_1^j, g_2^j, \dots, g_i^j, \dots, g_n^j$, are the corresponding local encoding coefficients used by node N .
- Y_j is a linear combination of the recieved data packets that will be transmitted by node N . [4]:

$$Y_j = (g_1^j \quad g_2^j \quad \dots \quad g_i^j \quad \dots \quad g_n^j) \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_i \\ \vdots \\ M_n \end{pmatrix} = \sum_{i=1}^n g_i^j \cdot M_i \quad (4.1)$$

This summation is performed using a unary *xor*.

The following example show us the difference between the traditional method for simply forwarding packets and network coding.

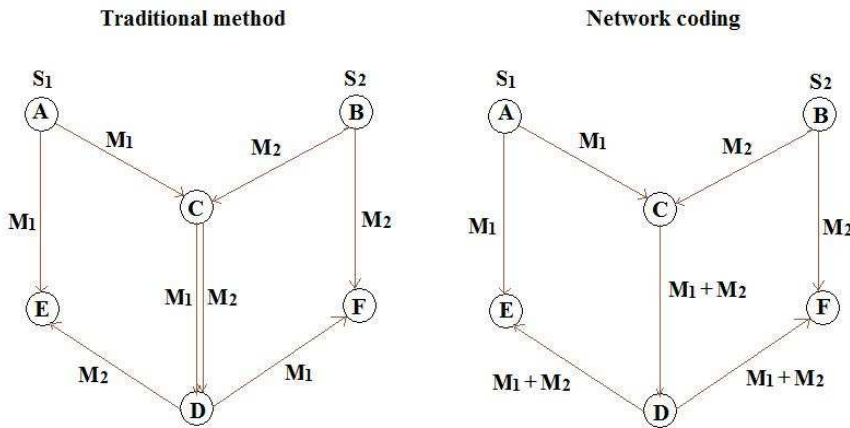


Figure 4.2: Traditional method (a) and versus network coding (b)

- Figure 4.2 (a): Assume that we wish to multicast two packets M_1 and M_2 to both node E and node F from the source source S_1 and S_2 . As the figure shows, between node C and node D , either two channels or two transmissions times are needed (here we assume that the two channels could be implemented by frequency division multiplexing or code division multiplexing: where as the use of two time slots uses time division multiplexing). Additionally we are able to send M_1 from node A to node E and M_2 from node B to node F [5].
- Figure 4.2 (b): If we do the same transmission using network coding. Node C receives and *xors* the two packets M_1 *xor* M_2 , then sends the result to node D . Node 4 forwards the encoded packets to node E and node F . Node E receives the combined result from node D and the original M_1 from node A . Therefore it is able to decoded the packet M_2 . Similarly node F receives the combined result from node D and the original M_2 from node B and can decode packet M_1 . Due

to the synchronization of all the links each of the 7 channels sends only a single packet during each time interval - in order to transmit the message once [5].

- Node C performs a simple *linear coding* of the packets it received [5].

4.2 Encoding

Encoding can also be performed recursively on previously encoded packets. We assume that a node has received and stored a set of encoded data packets $(g^1, Y_1), (g^2, Y_2) \dots (g^n, Y_n)$. Let $G = (g^1, g^2, \dots, g^n)$ be a matrix where row j is the coefficients g^j corresponding to the packet sum Y_j , $Y = (Y_1, Y_2, \dots, Y_j, \dots, Y_n)^{-1}$ be a vector where Y_j is the packet sum at node j . This node will create a new encoded packet (g', Y') by choosing a set of coefficients $h = (h_1, h_2, \dots, h_n)$ and computing a new linear function $Y' = \sum_{i=1}^n h_i \cdot Y_i$. Note that the set of coefficients g' is not simply equal to h , since the coefficients are with respect to the original packets M_1, M_2, \dots, M_n , $g' = \sum_{i=1}^n h_i \cdot g_i^j$ [4].

4.3 Decoding

If a node has received the set (G, Y) , then the node needs to solve the equation system for each j : $Y_j = \sum_{i=1}^n g_i^j \cdot M_i$ in order to retrieve the original packets [4]. The decoding is based on solving a set of linear equations and since linear dependencies may occur Gaussian elimination is used to remove these from the matrix to achieve full rank. The node recovers the source packets $M = (M_1, M_2, \dots, M_n)$, by computing $M = G^{-1} \cdot Y$ [4].

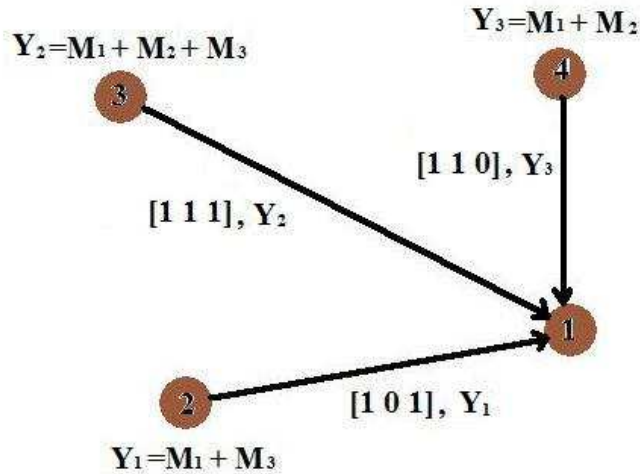


Figure 4.3: Example of decoding

Figure 4.3 shows an example of decoding. Node 1 receives data from its neighbors, node 2, 3, and 4 (these are messages M_1, M_2 , and M_3):

$$G \cdot M = Y \quad (4.2)$$

$$\begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} M_1 \\ M_2 \\ M_3 \end{pmatrix} = \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \end{pmatrix} \quad (4.3)$$

In this example G is the matrix with ones and zeros, M is the set of source messages M_1, M_2 , and M_3 , and Y is the set of outputs Y_1, Y_2 , and Y_3 . IF this system of equations are linear independent (hence the inverse of the matrix exists), and the number of unknown messages and the number of equations are equal, then the system of equations can be solved by $M = G^{-1} \cdot Y$.

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \end{pmatrix} = \begin{pmatrix} M_1 \\ M_2 \\ M_3 \end{pmatrix} \quad (4.4)$$

4.4 Advantages and disadvantages

1. Advantages

Network coding enables better resource utilization by allowing a network node to encode its received data before forwarding it [13].

2. Disadvantages

- The major problem with network coding is that the loss of one packet could affect many other packets and renders some information useless at the receiver. In network coding, one packet loss in the network results in several packet losses for the receivers. In fact, the combining of packets leads to an increase in shared fate, thus significantly reducing the reliability of packet delivery (as the probability of a successful transmission is dependent upon their being no errors on any of the links).
- When network coding is used, the delay will be increased, as there is an increased need for buffering, and more calculations are required to do than simply forwarding packets.

Chapter 5

Local network coding improvement for MPR flooding

5.1 Main idea

This thesis project focus on determining if we can significantly reduce the number of transmissions in MPR flooding by allowing the MPR nodes to use network coding.

We take advantage of the fact that in MPR flooding each MPR node has information about the links between its neighbors, which means that a MPR node knows exactly which of its neighbors that receive the same packets as it receive itself. By using this stored information, the node can use network coding to further reduce the number of transmissions needed to forward packets to its neighbors.

5.2 Transmission reduction

For MPR flooding without network coding, let Λ_i be the number of packets that node i transmits during a time interval Δ . Of these Λ_i packets, Γ_i packets originates from the node i itself, while the rest are retransmitted. Let Λ_{li} be the number of known packets from node l that node i must retransmit to its neighbors.

$$\Lambda_i = \sum_{l, l \neq i} \Lambda_{li} + \Gamma_i \quad (5.1)$$

If node i is not selected as an MPR node, then $\Lambda_i = \Gamma_i$. For simplicity, in the rest of the report we assume that all nodes transmit one packet each as source nodes. This means that $\Gamma_i = 1$ for all nodes i .

Assume that MPR node i uses network coding to encode packets into an equation system, so that all its neighbors can decode the received equation system. Of the Λ_i packets that are encoded, some of the packets are already known by the neighbors, and some packets are unknown. Let U_i be the maximum number of unknown packets at any of node i 's neighbors. We assume that it is always possible to encode the Λ_i packets into U_i packet sums so that all neighbors can solve the received equation system. With network coding, the number of transmissions from node i is now reduced from Λ to U_i . In order to calculate U_i we need to know the number of known packets at each neighbor.

Therefore, let $X_i(l)$ be the number of known packets at a neighbor l to node i .

$$X_i(l) = \Lambda_{li} + \sum_{m \in N(i), m \neq l} \Lambda_{mi} \cdot A_{ml} \quad (5.2)$$

In this function we have that:

1. $N(i)$ is the number of neighbors of node i .
2. A is connection matrix for the network. $A_{ml} = 1$ when there is a link between node m and l , which means that node l and node m are neighbors.
3. $\Lambda_{mi} \cdot A_{ml}$ is the number of packets sums which both node l and the MPR node i received from node m when node m transmitted a packet to node i .
4. Summing over all neighbors of node l , expression $\sum_{m \in N(i), m \neq l} \Lambda_{mi} \cdot A_{ml}$ give the number of packets which both node l and the MPR node i receive from all the neighbors of node l when they transmit their packets to node i .

Now we can express U_i , the maximum number of unknown packets at node i 's neighbors as:

$$U_i = \Lambda_i - \min_{l \in N(i)} X_i(l) \quad (5.3)$$

Let R be the relative traffic reduction with network coding for the network. Without network coding, MPR flooding generates $\sum_{i=1}^n \Lambda_i$ transmissions in the network. With network coding, each node i will save $\Lambda_i - U_i$ transmissions. Thus we can express R as:

$$R = \frac{\sum_{i=1}^n \Lambda_i - U_i}{\sum_{i=1}^n \Lambda_i} = \frac{\sum_{i=1}^n \min_{l \in N(i)} (X_i(l))}{\sum_{i=1}^n \Lambda_i} \quad (5.4)$$

5.3 Example

Consider the number of transmissions in MPR flooding when the MPR nodes shown in figure 5.1 use network coding.

- Nodes (l, m, n) have selected node i as their MPR node, node i receives packets from all of these nodes and re-broadcast a computed packet to its neighbors if they have not received packets from their own neighbors previously.
- the links with arrows, such as $k \rightarrow i$, means that node i is one of the MPR nodes of node k .
- the link without arrows, means that the two nodes are not MPR nodes to each other, however they are neighbors.
- M_i, M_k, M_l, M_m, M_n are the packets from nodes i, k, l, m, n .

In this example, node l received one packet from node n , and it has one packet from itself before it send packet to node i ; node n received one from node l and it has one packet from itself; node m and node k received only one packet from themselves. In order to that all the MPR nodes to node i could decode all the packets they have received, when the node i receives all the packets from nodes k, l, m, n , it will re-combine and

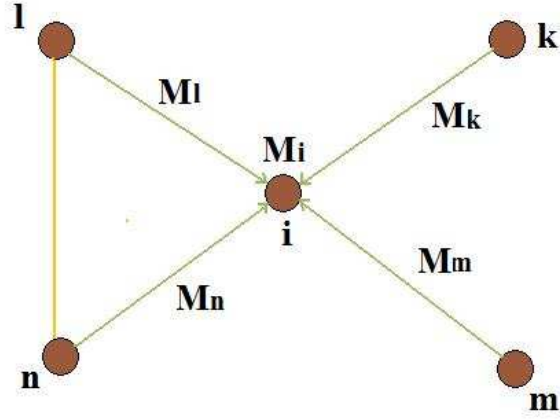


Figure 5.1: Transmissions in MPR flooding when the MPR nodes use network coding

re-broadcast all packets (include the packet from node i) to it's MPR node (here node i 's MPR are nodes k, l, m, n).

By studying the example above, when network coding is used, it is enough for node i to transmit packet sums $(M_i, M_m + M_l, M_k + M_n, M_n)$ so that the other nodes can decode all the packets ($U_i = 4$, see the calculations blow). We see that there are four re-transmissions at node i by using network coding. Without using network coding there would be one more re-transmission.

- $\Lambda_{li} = 1, \Lambda_{ni} = i, \Lambda_{ki} = 1, \Lambda_{mi} = 1$ We assume that i has one own packet to transmitt: $\Gamma_i = 1$.
- $\Lambda_i = \sum_{l \in \{k, l, m, n\}} \Lambda_{li} + \Gamma_i = 5$
- $X_i(l) = 2, X_i(n) = 2, X_i(k) = 1, X_i(m) = 1$
- $U_i = \Lambda_i - \min_{l \in N(i)} X_i(l) = 5 - 1 = 4$
- $R = \frac{\Lambda_i - U_i}{\Lambda_i} = \frac{5-4}{5} = 20\%$

Chapter 6

Simulations

6.1 The Simulator

A Matlab based simulator is used to evaluate the reduction in messages for a number of different network topologies and with a number of different total numbers of nodes. The simulator is divided into functions . The functions for generating a network were developed by Jacob Löfvenberg (a resercher in FOI). The other functions were written by myself, specially: a function for selection of MPR nodes, a function for MPR flooding, and a function for calculating known packets.

6.2 Assumptions for the simulations

Earlier we stated some of our assumptions. In order to simplify the simulation, network coding will be the single factor which we consider. Therefore we have made the following assumptions during our simulations:

1. We create a random network using nodes distributed in a two dimensions. We study the propagation of traffic streams through this network. Each node in the network broadcasts a data packet.
2. In order to simplify the simulation, we assume that all the nodes in the network are perfectly synchronized and that our system nodes utilize distributed TDMA for their communication.
3. We also assume all nodes in the network can decode all the incoming packets from its previous neighbors' transmissions.
4. We assume that there are not bit errors over the link, hence there is no packet loss and there are no error packets transmited.
5. Finally, we assume that MPR nodes always have correct and current knowledge about their neighbors's neighbors based upon 'HELLO' messages.

6.3 Simulation setup

1. Create a random network in two dimensions. We simulate 10 kinds of networks with two forms, 5 rectangle (10*1) networks with 10, 20, 40 ,80 and 160 nodes,

and 5 square (10×10) networks respectively. We chose these kinds of networks, because we estimated that there might be between 10 to 200 vehicles in a network. We simulated 5000 networks for each combination of size and form of network. After creating the networks, we compute the a connectivity matrix. This matrix contain information on the one-hop neighbors for every nodes in the network.

2. Then we use the MPR selection algorithm in section 3.1 to compute a MPR matrix. This matrix is a subset of the one-hop neighbors (i.e. MPR nodes) which are able to reach all the two-hop neighbors, and these MPR nodes forward packets to the two-hop neighbors.
3. Finally, we calculate the total numbers of transmissions for all MPR nodes by using MPR flooding technique in section 3.2. We then estimate the reduced traffic ratio by function in equation 5.4.

Chapter 7

Results

7.1 Graphic results

In figures 7.1 to 7.10, dark blue lines represent the average traffic reduction. Cyan stars represent the traffic reduction ' R ' for the networks where only one of the nodes in the network has been selected as MPR. Magenta squares represent the networks with two MPR nodes. Red cross are for the networks with 3 MPR nodes. Black circles are for networks with four MPR nodes. Green dots present networks with more than four MPR nodes. Black diamonds represent networks without MPR nodes.

7.2 Compare the simulation results

Generally, the largest average traffic reduction ' R ' in these simulations, is between 25% and 30% when a network has only one MPR node, and the average number of neighbors is approximately 70% of the total number of nodes. The worst average traffic reduction is 0% when a network is full connected.

In small networks with 10 nodes (shown in Figure 7.1 and Figure 7.2), we see that the average traffic reduction R is scattered, but we can see the tendency is that average value of R decreases from around 25% (for rectangle networks) and 20 % (for square networks) when the average number of neighbors increase. This is especially true when the networks consist of full connected nodes, where the R is zero. In next section we will discuss this behavior. In Figure 7.7, Figure 7.9, and Figure 7.10, the blue curves are still connected even when the groups of cyan stars are totally separate from the others, because the blue lines are the average values of traffic reduction R , and there are both higher R and lower R in the same interval. However, in these cases there many be zero or few situations near the average.

However, when we increase the number of nodes to 20 nodes (as shown in Figure 7.3 and Figure 7.4), R decreases to around 10% for rectangle networks and 12% for square networks until the average number of neighbors increases to around 12 nodes, after that, there is only one MPR node in each network, and the nodes in networks are almost fully connected, hence R increases and then decreases.

Figure 7.5 and Figure 7.6 show the results for networks with 40 nodes. It can be seen clearly that the group of networks with only one MPR node begin to separate from the other networks. These networks always have a higher average traffic reduction with a higher average number of neighbors. As the average number of neighbors decreases,

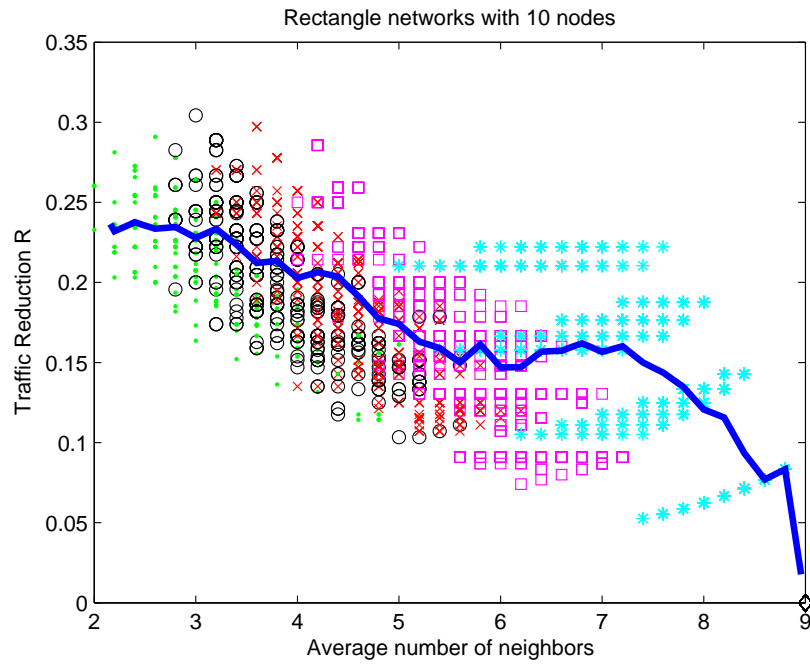


Figure 7.1: traffic reduction, network coding be used in a rectangle network with 10 nodes

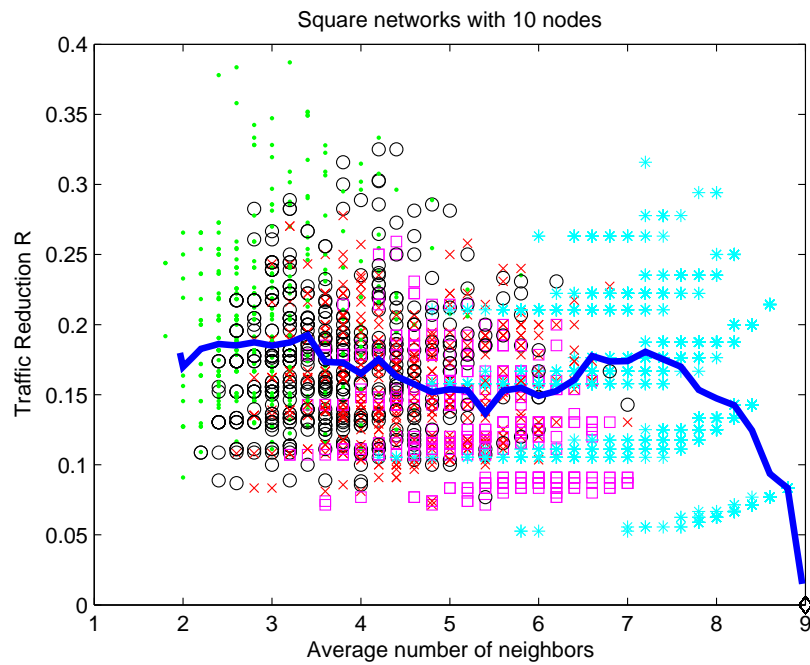


Figure 7.2: traffic reduction, network coding be used in a square network with 10 nodes

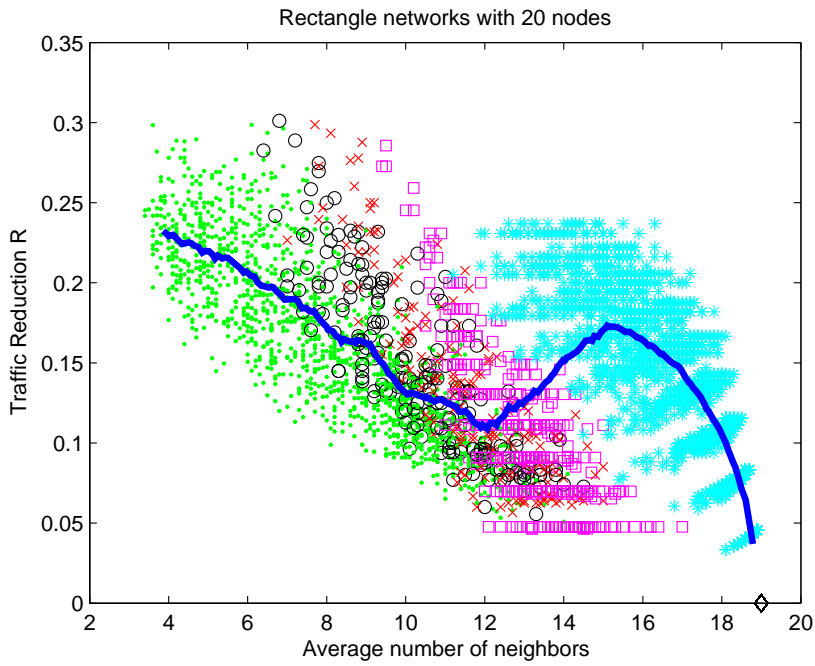


Figure 7.3: traffic reduction, network coding be used in a rectangle network with 20 nodes

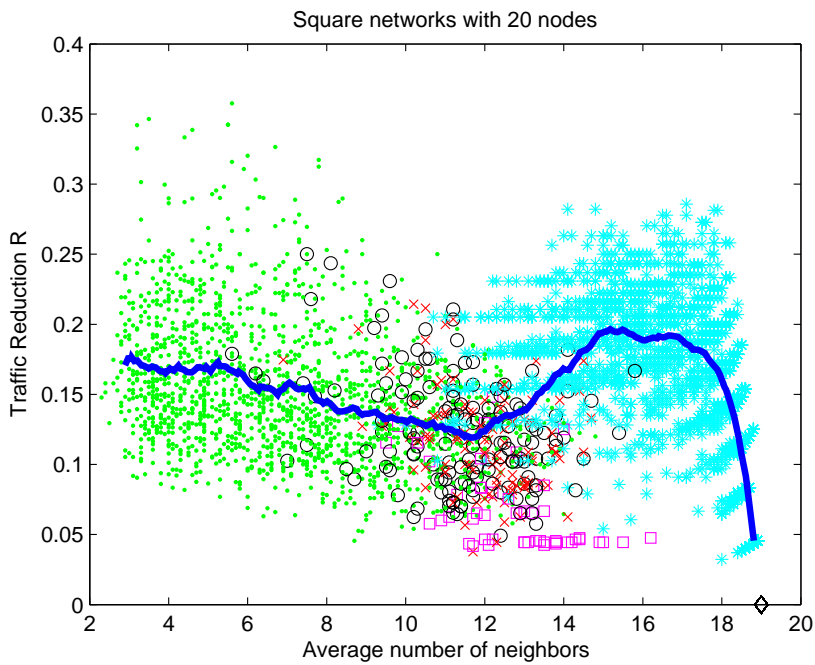


Figure 7.4: traffic reduction, network coding be used in a square network with 20 nodes

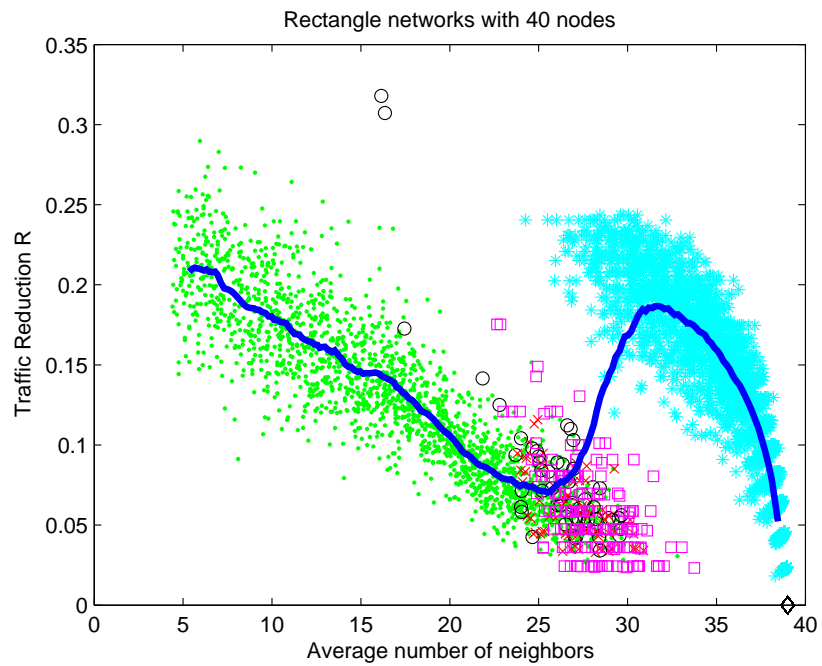


Figure 7.5: traffic reduction, network coding be used in a rectangle network with 40 nodes

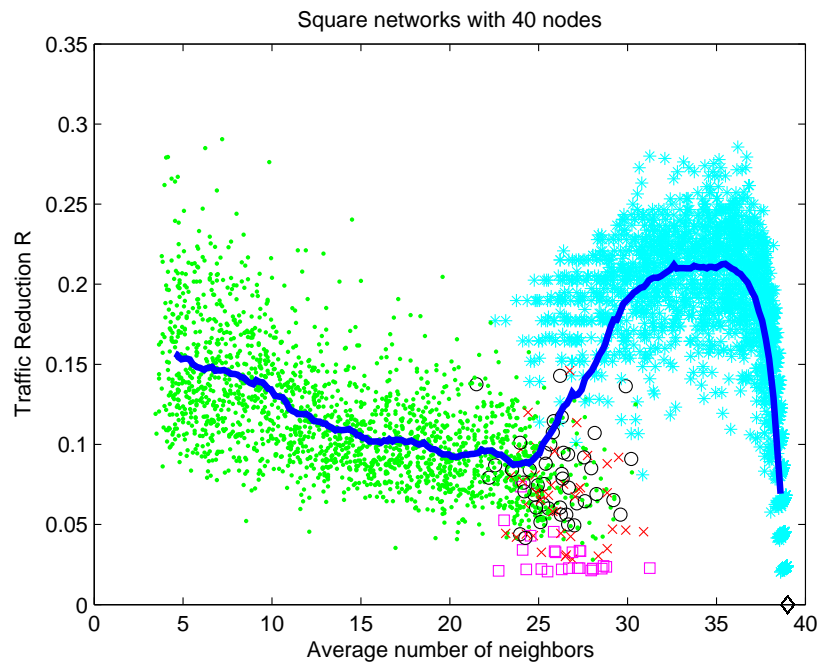


Figure 7.6: traffic reduction, network coding be used in a square network with 40 nodes

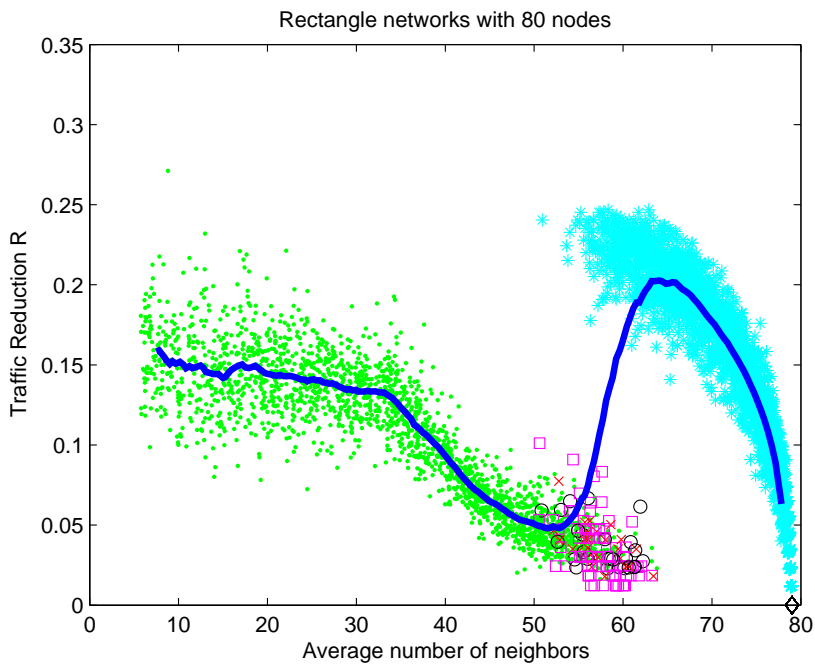


Figure 7.7: traffic reduction, network coding be used in a rectangle network with 80 nodes

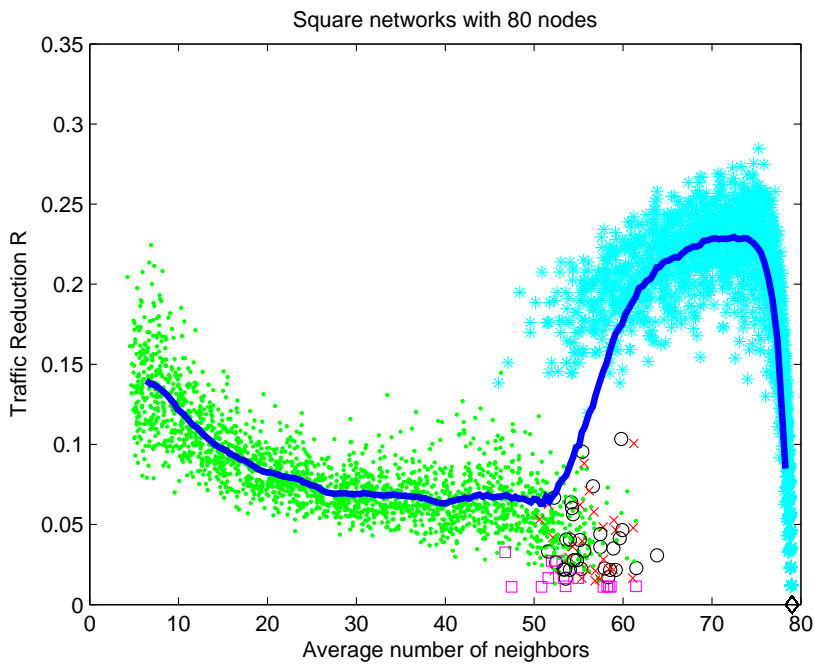


Figure 7.8: traffic reduction, network coding be used in a square network with 80 nodes

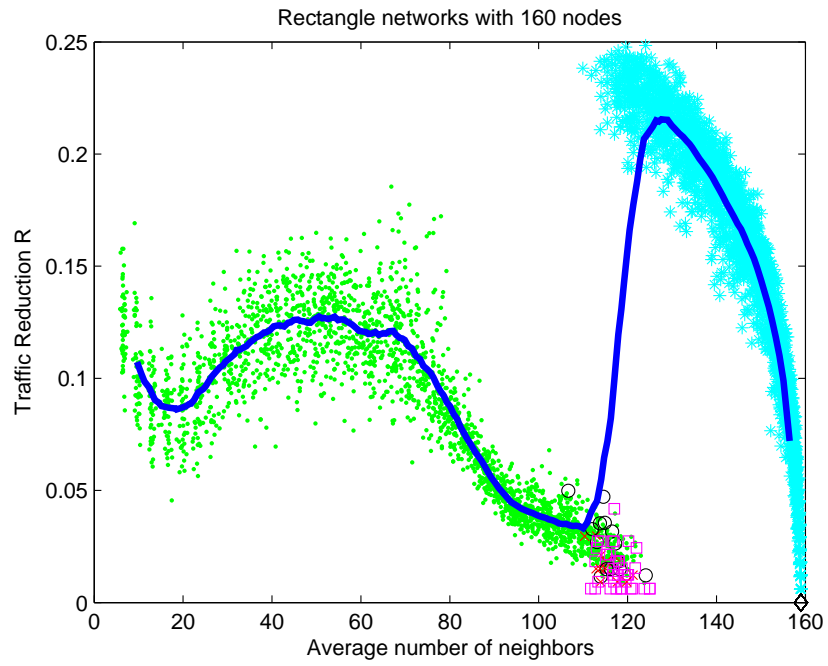


Figure 7.9: traffic reduction, network coding be used in a rectangle network with 160 nodes

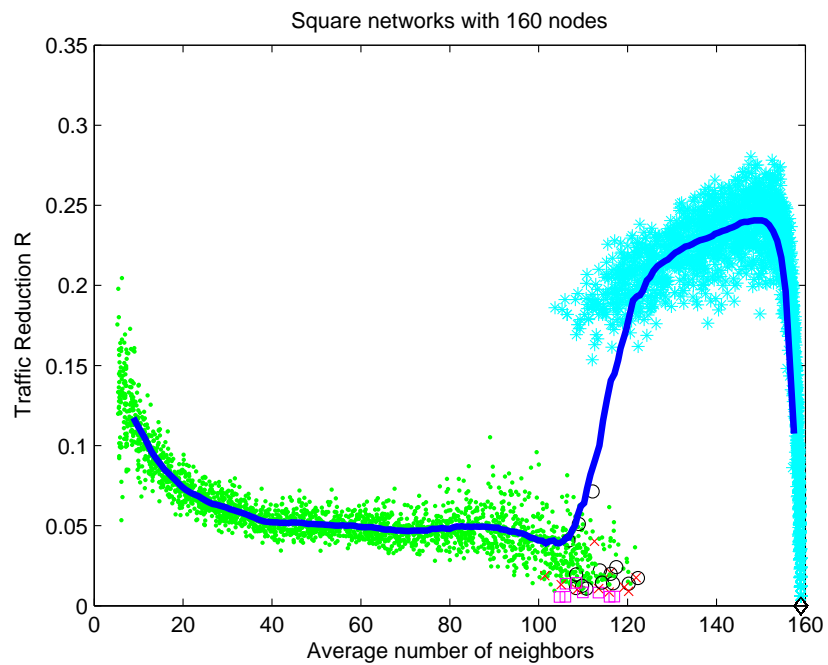


Figure 7.10: traffic reduction, network coding be used in a square network with 160 nodes

the number of MPR nodes increases, and the estimated value of the average R are better than in most of our simulations. Note that in Figure 7.9, there is a 'hump' when the average number of neighbors is between 20 and 80. See section 7.3.3.

7.3 Comments

1. For a fully connected network, the traffic reduction will be zero, because all nodes can reach each other directly, hence there are no MPR nodes. All packets will only be transmitted once. In the figures, if there are n nodes in a network and the average number of neighbors is $n - 1$, there is always a black diamond on the horizontal axis.
2. For a network with only one MPR node, the nodes in this network are almost fully connected. The group of networks with one MPR node is totally separated from the others, especially for networks with 160 nodes (both rectangle and square forms), see Figure 7.9 and Figure 7.10.
 - If there is only one MPR node which connects two group of nodes (see figure 7.11), then because the MPR node chooses the minimal number of known packet sums as the number of encoding equations (as described in section 5.2). If there are the same number of nodes in each group, this leads to the best result for this type of network. The number of minimal number of known packet sums is equal.

$$R = \frac{\frac{n-1}{2}}{n + (n-1)} \approx \lim_{n \rightarrow \infty} \frac{n-1}{4n-2} = 0.25 \quad (7.1)$$

In Figure 7.11, the traffic reduction will be $\frac{4}{9+8} = 0.2352$.

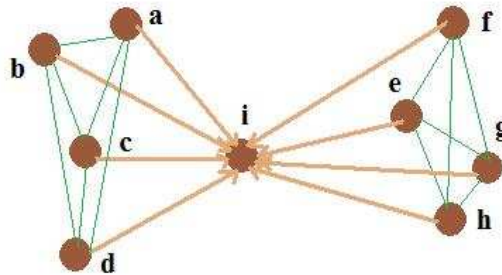


Figure 7.11: 'Bottleneck' distributed nodes in network with one MPR node

When there is only one node on one side, and many nodes on the other side, then the value on R will be very low, because the minimum number of known packet sum is one and the total number of transmissions is high. We can see this in the figures for networks with 40, 80, and 160 nodes.

- The example above shows two groups connected by one MPR node. If we move the two groups until they merge into one, there are still unconnected nodes. Figure 7.12 shows an extreme example of a network with one MPR node. Except for the MPR node i , there is only one unconnected node for

each node. The minimum number of known packet sums for the neighbors of i is $n - 2$, and the total transmission packet sums for i is $n + (n - 1)$.

$$R = \frac{n - 2}{2n - 1} \approx \lim_{n \rightarrow \infty} \frac{1 - \frac{1}{n}}{2 - \frac{1}{n}} = 0.5 \quad (7.2)$$

This kind of network will give the best result on R , however the probability of having such a network randomly is very low.

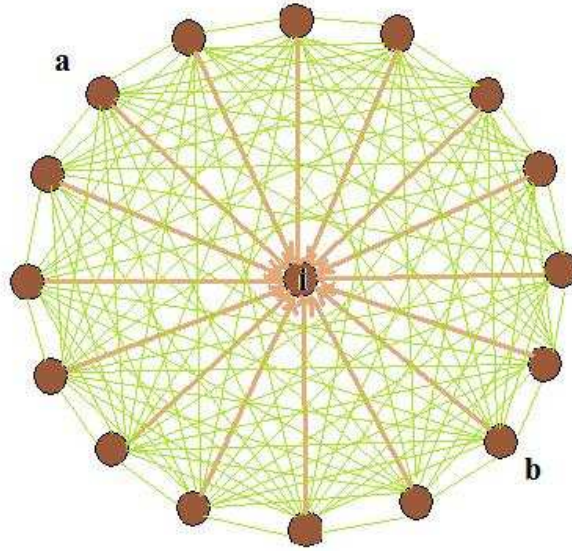


Figure 7.12: Example network which gives best result on 'G'

3. In our simulations, networks with two MPR nodes give poor traffic reduction R , because the minimum number of known packet sums for neighbors of each MPR node are low. This is especially true in a network with many nodes, because the total transmission packet sum is high. Results can be seen clearly in Figure 7.7, Figure 7.8, Figure 7.9, and Figure 7.10. This type of network always happened when the average number of neighbors is between 60% and 70% of total number of nodes. For a small network, the total packet sum is low, hence R is variable, but it could be higher.

As the average number of neighbors decreases, the number of MPR nodes increases, and traffic reduction R increases. However, in Figure 7.9, there is a 'hump' when the average number of neighbors is between 20 and 80. This phenomenon depends on how the MPR nodes are distributed in the networks.

4. For a network with all nodes lined up like a chain, see Figure 7.13. The total number of MPR nodes is $n - 2$, the total transmission packet sum is $n + (n - 1)(n - 2)$, and the total minimum number of known packet sums is $2(1 + 2 + \dots + (\frac{n}{2} - 1))$. Using equation 5.4, the value of traffic reduction is:

For even n :

$$R = \frac{2(0 + 1 + \dots + (\frac{n}{2} - 1))}{n + (n - 1)(n - 2) + 1} = \frac{(\frac{n}{2} - 1)\frac{n}{2}}{n^2 - 2n + 2} \rightarrow 0.25, \quad n \rightarrow \infty \quad (7.3)$$

For odd n :

$$R = \frac{2(0 + 1 + \dots + (\frac{n-1}{2} - 1)) + \frac{n-1}{2}}{n + (n-1)(n-2) + 1} = \frac{(\frac{n-1}{2} - 1)\frac{n-1}{2} + \frac{n-1}{2}}{n^2 - 2n + 2} \rightarrow 0.25, \quad n \rightarrow \infty \quad (7.4)$$

The traffic reduction for each node are:

Nodes number:

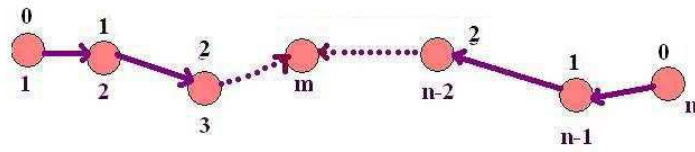


Figure 7.13: Special network

Chapter 8

Conclusions and future work

8.1 Conclusions

We can first conclude that network coding always has a positive effect for MPR flooding in a MANET. The results of our simulations are variable, with the traffic reduction up to 40%, although theoretically it should be up to 50% (see 7.3). The traffic reduction depends heavily on the network topology, how the MPR nodes are distributed, and how many MPR nodes there are in the network.

The nodes in a real tactical MANET move, so that the network topology will change over time. Using the MPR algorithm, the MPR nodes can use the updated information about one hop and two hop neighbors' connections, and it is possible to calculate the potential traffic reduction, thus allowing a decision to be made of whether it is advantageous to use network coding.

In our simulations, the factors we considered are the number of nodes, form of networks, and range which depends on the distance between nodes. However when network coding is used, the factors such as delay, packet losses, and so on, must be considered in the network design.

8.2 Future work

1. It should be interesting to further evaluate how different rectangle proportions effect the results.
2. How does the traffic reduction compare to the additional costs if we include the factors, such as delay, packet losses, channel errors and additional computation for mobile networks.

Bibliography

- [1] Wikipedia, Mobile ad hoc network, http://en.wikipedia.org/wiki/Mobile_ad_hoc_network. last modification on 24 November 2008.
- [2] J. Wu and J. Cao., *Connected k-hop clustering in ad hoc networks*. Digital Object Identifier: 10.1109/ICPP.2005.25
- [3] Wikipedia, Network coding: Theory, http://en.wikipedia.org/wiki/Network_coding#Theory
- [4] S.Crisostomo; J.Barros and C.Bettstetter, *Flooding the Network: Multipoint Relays versus Network Coding* in IEEE International conf. on Circuits and systems for communications.
- [5] Raymond W. Yeung, *Network Coding Theory*, Now Publishers Inc, USA, pp 5-9. June 2006
- [6] L.V.A. Qayyum and A. Laouiti, *Multipoint relaying for flooding broadcast messages in mobile wireless networks* in Proc. Hawaii International Conference on Systems Science, Big Island, HI, USA, Jan. 2002.
- [7] P.Jacquet, A.Laouiti, P.Minet, and L. Viennot, *Performance analysis of OLSR multipoint relay flooding in two ad hoc wireless network models*, INRIA, Tech. Rep. 4260, September, 2001
- [8] T. Clausen and P. Jacquet (Editors), *Optimized Link State Routing Protocol(OLSR)*, IETF, Request for Comments 3626, October 2003.
- [9] Wikipedia, Media Access Control, http://en.wikipedia.org/wiki/Media_Access_Control. Last modification on 21 January 2009.
- [10] Wikipedia, Time division multiple access, http://en.wikipedia.org/wiki/Time_division_multiple_access. Last modification on 4 February 2009
- [11] Wikipedia, Dynamic Source Routing, http://en.wikipedia.org/wiki/Dynamic_Source_Routing. Last modification on 17 March 2009.
- [12] Carlos H. Rentel and Thomas Kunz, *Network Synchronization in Wireless Ad Hoc Networks*, Carleton University, Systems and Computer Engineering, Technical Report SCE-04-08, July 2004
- [13] Taku Noguchi, Takahiro Matsuda, Miki Yamamoto, *Performance Evaluation of New Multicast Architecture with Network Coding*, IEICE Trans. Comm. June, 2003

- [14] R. Ahlswede, N. Cai, S-Y.R. Li, and R.W. Yeung. Network information on. *IEEE Transactions on Information Theory*, vol 46, pp 1204-1216, July 2000.

