Graduate Theses and Dissertations

Iowa State University Capstones, Theses and Dissertations

2009

Network coding in wireless networks

Nishanth Reddy Gaddam
Iowa State University

Network coding in wireless networks

by

Nishanth Gaddam

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Computer Engineering

Program of Study Committee:
Arun Somani, Major Professor
Manimaran Govindarasu
Aditya Ramamoorthy

Iowa State University

Ames, Iowa

2009

TABLE OF CONTENTS

iii

ACKNOWLEDGEMENTS

I take this opportunity to express thanks to those who made a difference in my life for the last couple of years in Ames. First and foremost, I thank Dr. Arun Somani for giving an opportunity to pursue research at Iowa State University as an undergraduate exchange student from India. He gave me complete freedom and support to choose research topic and gave me direction to pursue it till completion. His guidance, patience, motivation throughout my academic life in Ames helped me to grow professionally and personally. During my Teaching Assistant duties he also motivated me to learn some of the expertise he had in Compute Architecture and Digital Design. I would like to thank Dr. Manimaran for his valuable suggestions that also helped me take up research in my earlier days of graduate study. I enjoyed taking the CPRE 545 course. The discussions helped me to improve my problem analysis skills and critically evaluate the problem. I also thank Dr. Aditya Ramamoorthy for introducing research problems in the network coding course, from where I developed a solution for one of my dissertation topics.

I specially thank Anil for being my mentor throughout my college days. He was always there with helping hand, when I definitely needed. I thank Veerendra for his comments, insights and for sharing the vast knowledge he posses. I also thank Prasad for being part of my boat from BITS to ISU. I thank Visu for all his help from the initial days of providing accommodation to attending defense mock sessions. I also thank my labmates David, Prem, Pavan, koray for keeping the lab atmosphere friendly and lively. Also, I would like to thank Pavan Vennapusa, Satya, Mahesh, Santhosh, Prasad and many others, who have provided me many memories to remember in the future.

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Network coding improves throughput in wireless networks. When applied to battery driven devices, like wireless sensor nodes, it extends the network lifetime. Network coding reduces the energy consumption by minimizing the number of transmissions required to communicate a given amount of information across the network. However, aggressive application of network coding adversely affects the network lifetime. We illustrate this trade off in this paper, and show that the existing throughput based network coding approaches cannot be applied to energy-constrained networks. Specifically, we address the following routing problem. Given a set of traffic demands the goal is to route the demands across the network with the objective of minimizing the total energy consumption while providing guarantees on the lifetime of individual nodes.

This work studies both multi-path and single-path variations of the above routing problem. We present analytical formulations to solve the problems optimally. Evaluation of the multi-path problem indicates that the proposed solution is 35% more energy efficient than no-network-coding solution while still meeting required lifetime constraints.

However, network coding is a costly technique to apply. This technique involves extra overhead in terms of control message transmissions, and may result into unbounded delays. These factors offset the performance enhancements that are otherwise achievable through network coding. In this work, we characterize a network to determine regions (nodes), where application of coding can be advantageous. This serves two purposes. First, if a network is well suited to effectively use coding then performance enhancement would dominate instead of latency and additional overhead issues. Second, coding-aware routing protocols can be designed, which use topology information to route the packets effectively in the network.

In the report, we construct a neighborhood map at a node and derive a graph, called a transformed graph, out of the map to analyze coding opportunities at that node. We theoretically show that all possible coding opportunities can be derived from this transformed graph. We also develop an algorithm to determine all possible coding scenarios at that node. This approach is used to determine coding capabilities of all nodes in the network. We experimentally evaluated our scheme on different topologies. We also demonstrate that the node degree and the edges between the neighbors of the node are the important factors in determining coding gains at a node. Finally, a light-weight heuristic solution is proposed to identify the coding degree of node in the network.

CHAPTER 1.   Introduction

Recent technological advancements in wireless communications are fundamentally changing the manner by which devices communicate with one another. Modern wireless devices build networks on their own and aid each other in passing information to any device in the network. In these networks throughput, latency and network lifetime have become the crucial performance parameters.

The network coding as a technique improves throughput in wireless networks. The throughput enhancement achievement is many folds higher than normal routing in wireless networks.

The basic idea of network coding is illustrated in Fig. 1.1 where nodes A, B and C share the common wireless medium (4). Consider a scenario where nodes A and C have information to exchange. Due to the channel constraints only one node can transmit at any given time. One possible way of accomplishing this information exchange is as follows. Node A sends its packet ($p1$) to relay node B. Node B forwards this packet to node C. Similarly, node C sends its packet ($p2$) to node B which in turn forwards it to node A as shown in the figure. This involves a total of four transmissions.
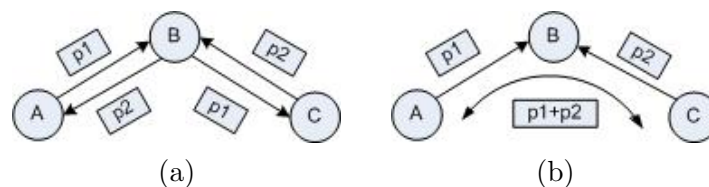
Figure 1.1   (a) No-Coding (b) Network-Coding.

Now consider the scenario where network coding is applied to reduce the number of transmissions. Nodes A and C transmit packets to central node B sequentially (two transmissions).

Node B, instead of directly forwarding each packet to its destination, XOR's the two packets and broadcasts the result as a single packet in the shared medium as shown in the figure. Both nodes A and C know their own packet ($p1$ and $p2$, respectively) that originated from them. They can therefore retrieve the unknown packet by XORing the known packet with broadcast packet. For example, node A on receiving $p1 \oplus p2$ performs the operation $p1 \oplus (p1 \oplus p2)$ to obtain $p2$. Similarly, node C retrieves packet $p1$. This entire process takes exactly three transmissions as opposed to four transmissions as discussed above.

The above particular implementation of network coding concept where the final packet is retrieved using XOR-ed broadcast packet (in the above example, it is $(p1 \oplus p2)$) and the locally originating packet ($p1$ and $p2$) is often termed as opportunistic coding (4). A different flavor of network coding called opportunistic listening exploits the shared characteristic of the broadcast wireless medium to achieve even better savings. The following example demonstrates the basic idea behind opportunistic listening.

Fig. 1.2 shows four border nodes surrounding a single node. In this case all nodes broadcast their own packets say $p1$, $p2$, $p3$, $p4$, respectively. Assume node A has the opportunity to listen to the transmissions from node B and D. Similarly, every border node can listen to the two other nodes i.e., its neighbor nodes on either side. Now, assume that nodes A and C each have a message to exchange with each other (messages, $p1$ and $p3$). Similarly, nodes B and D each have a message to exchange with each other (messages $p2$ and $p4$). After each node transmits its message to the central node, the central node has a chance to XOR all the four packets ($p1 \oplus p2 \oplus p3 \oplus p4$) and send the result in a single transmission. Each destination node can retrieve the transmitted packet using the packets it listened from its two neighboring nodes. For example, node C has packets $p2$ and $p4$ by the direct listening of nodes B and D transmissions. In addition it has its own locally generated packet $p3$. Now, it can retrieve the target packet ($p1$) by performing, $p2 \oplus p3 \oplus p4 \oplus (p1 \oplus p2 \oplus p3 \oplus p4)$. The entire process reduces the number of transmissions from 8 to 5. This is due to the opportunistic listening of nodes in the network.
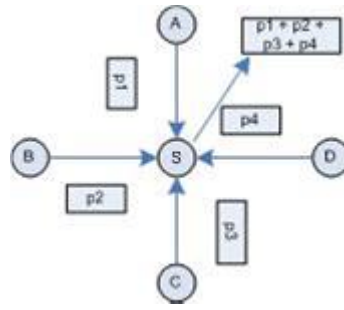
Figure 1.2   Opportunistic listening in wireless networks

1.1   Energy minimization through network coding

In the present work, we apply network coding as a energy minimization technique at the routing layer and analyze the related trade offs. Applying network coding to energy limited networks requires addressing a new set of challenges and brings out the trade offs between total energy savings and network lifetime. Routing traffic to encourage network coding with the aim of minimizing the total energy consumption develops hot spots in the network. These hot spots result in poor network lifetime.

To the best of our knowledge, our work is the first to apply network coding as an energy minimization technique and study the trade off between total energy minimization and network lifetime. We analyze this trade off by addressing two specific and related routing problems namely, *Multi-Path Routing Problem* and *Single Path Routing Problem* in a static manner.

1.2   Determining coding regions in network

The practical network coding face challenges in terms of overhead incurred in routing the control messages and unbounded latency in packet transmissions (23). In this report we present an approach to determine exact coding capability at each node in the network. This allows identification of the regions in the network that are more suitable to deploy network coding. This information then can be used in the design of the routing scheme to optimize the network performance.

In most of practical applications wireless nodes are deployed randomly in the network. A

nodes location in the network, its geographic proximity to other nodes, its neighborhood, and idle listening capability affects possibilities for network coding in comparison to other nodes. In this report, we present an algorithm to identify nodes that have higher chances of being able to deploy network coding. Based on the results obtained and analysis presented in the report, we demonstrate that coding gains at a node has very high correlation with the factors such as the degree of the node and the number of edges between its neighbors. We also explain the impact of each factor on the coding gains. These results play a greater role in designing the lightweight approaches in identifying the coding opportunities at a node in network and more effective routing algorithms.

CHAPTER 2.   Review of Literature

The recent work on network coding primarily focuses on improving the network throughput by aggressively applying coding technique described above (4) - (15). The work presented in (4) applies network coding to maximize network throughput at the MAC layer. Their protocol uses the known ETX metric to decide the possibility of coding the packets at the relay node or central node. The authors have shown that the network coding is an excellent technique to maximize throughput. The work in (5) presented routing protocols to aggressively exploit coding opportunities in the network. The basic idea is to route the flows in the network to a region where network coding can be performed. This improves the throughput, however it greatly affects the network lifetime. Concentrating large amounts of traffic to a small region in the network burdens the involved nodes. This routing approach can eventually lead to a network breakdown. Therefore, such an approach cannot be applied to energy constrained networks.

In (6), the authors presented a coding aware routing protocol for multiple unicast sessions in a wireless mesh network. The primary objective here is to apply network coding techniques to maximize the throughput while respecting the interference constraints due to channel capacities. They identified the trade off between routing flows for coding advantages and for avoiding interference. Although the proposed solutions are effective for energy-unaware networks, they cannot be directly applied to energy-constrained networks where network lifetime is of paramount importance.

In (8), MAC scheduling algorithms are presented to achieve throughput gains as expected from network coding. In most of the earlier work on network coding the underlying basic assumption is that there is no upper bound on the number of packets that can be coded

during a single transmission. This assumption in practice implies that network coding can result in infinite gains. But the real advantage gained through network coding depends on how effectively and efficiently the coding scheme can identify opportunities under practical settings.

Researchers in (16) - (18) have provided bounds on the throughput gains that can be achieved through network coding. (16) derived a constant upper bound for the throughput gains in wireless network. (17) studied and derived bounds for the gain in terms of number of transmissions when network coding is applied to minimize energy spent in sensor networks. (18) studied how realistic physical layer and medium access controlled by random access mechanisms affect the performance of practical network coding. It provides bounds on the number of packets that can be encoded by a coding node in each transmission.

CHAPTER 3.   Energy-aware Network Coding

3.1   System Model Description

Network topology is assumed fixed and all the nodes are monitored by a base station which computes paths based on requested connections. Base station is assumed to be a high-energy node with continuous supply of power.

The energy required to receive a packet is assumed negligible in our case because applying network coding does not reduce the number of packets received by a node. In relative terms, this energy is almost equal to a non-coding case. Also the magnitude of the energy consumed for receiving a packet is very less compared to the transmitted energy.

Our model does not assume homogeneity with respect to energy or functionality of a node in the network. They are allowed to start with different battery powers but once deployed in the network we assume that they do not have access to the external battery devices.

In this report we propose multi-path and single-path routing solutions depending on whether flows can be split or not. For the multi-path routing problem packets routed through different paths for the same flow might encounter variable delays and reach destination at different time. We assume destination node can take care of such unordered packets.

Network coding is discussed through a simple XOR operation on the transmitted packets. This operation requires very low energy. So we assume energy spent for XOR is negligible. A header is added to a packet to distinguish XORed packets from normal ones. The header also carries the sequence number of the packets that are XORed. This helps receiver node select relevant packet from its buffer and XOR with the packet received to retrieve the unknown packet. To keep our model simple we consider the case where only two packets can be XORed. Extra gains arising from XORing more than two packets are not considered in the present

paper and is left for future work.

3.1.1 Motivational Example

Network coding applied to energy constrained networks reduces the total energy consumed for a given amount of information across the network. Existing coding aware routing protocols discover regions in the network where there is a high probability for mixing the packet and diverts all traffic toward them. As a result, a few key nodes in the network handle high amount of traffic while others are left idle. Since residual energy depletes very fast due to network coding, key nodes die quickly compared to other nodes in the network. This leads to significant imbalance in residual energy of the nodes across the network which may result in reduced network lifetime even though a majority of the nodes are still alive and left with high residual energies.

We demonstrate the importance of designing lifetime aware routing protocols through a simple illustration. Consider the network in Fig. 3.1. Let there be two flows $f1$ and $f2$ scheduled from node $S1$ to node $D1$ and from node $S2$ to node $D2$, respectively. Each flow has a traffic value of 100 units i.e. $t(f1) = 100$ and $t(f2) = 100$. Assume all the nodes have initial energy equal to 200 units. Let E_R denote the residual energy and TE denote the total energy spent by all the nodes in forwarding the given traffic. We assume a single unit of energy is consumed to forward one unit of traffic through a single link.

Case I : No-Coding (Shortest Path Routing)

When network coding techniques are not applied, flows $f1$ and $f2$ choose shortest paths $P1$ and $P4$, respectively. Below we calculate the approximate total energy spent to meet the demands of two flows and residual energies in each node.

Transmissions required to forward unit traffic: 8.

$TE = 8 * 100 + 8 * 100 = 1600$ units of energy.

$E_R = 200 - 100 = 100$ units. This is same for all the nodes on paths P1 and P4 except $D1$ and $D2$.

For rest of the nodes E_R is 200 units. In this case we see that all the nodes along paths $P1$ and $P4$ drain to half of their initial energy, while nodes on paths $P2$ and $P3$ have residual energy equal to their initial value.
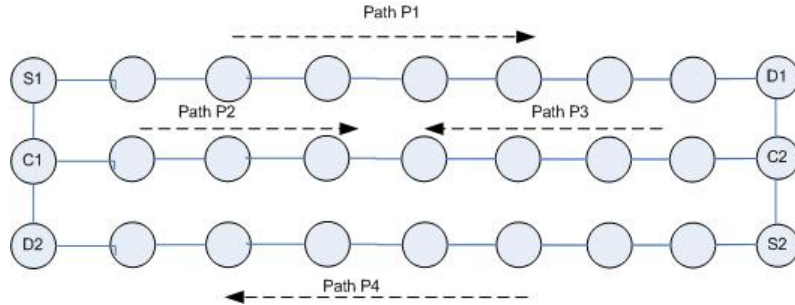
Figure 3.1   An Example to illustrate the importance of lifetime aware routing

Case II : Coding Aware Routing

Coding aware routing protocols encourage flows $f1$ and $f2$ choose paths $P2$ and $P3$, respectively. This is because these paths have higher path overlap. Network coding is possible when two opposite flows choose paths with same set of nodes.

Transmissions required for flow $f1$: 10.

Transmissions required for flow $f2$: 10.

Nodes where coding is possible: 7

Transient energy per node (Tr): 1 (explained later).

$TE = 10 * 100 + 10 * 100 - 7 * 100 + 6$

In the above computation we deduct 700 units of energy because when coding is used one unit of energy is sufficient to send two packets which are XORed together. Additional 6 units of energy is added to account for the energy spent during initial start-up process explained in next section.

ie., $TE = 1306$ units.

$E_R = 200 - 100 - 1(Tr) = 99$ units is same for all the nodes on paths $P2$ and $P3$ except corner nodes $C1$ and $C2$. These nodes spend single unit of energy to transmit one unit of

traffic as coding is not possible at those spots. The total traffic routed in the central route is 200 units. Corner nodes eventually die as they drain their entire energy to meet the demands of flows $f1$ and $f2$.

For rest of the nodes (excluding $S1$ and $S2$) E_R is equal to their initial energies. The total energy spent in coding aware routing is lesser than no-coding case. But the two corner nodes are dead as they drained their entire energy. As a result network is disconnected and at this point it is dead.

Case III : Lifetime Aware Routing

Lifetime aware routing protocols encourage flows to split traffic between different paths to maximize lifetime and to minimize transmission energy. Here we assume $f1$ diverts half of its traffic from path $P1$ to path $P2$. Similarly, flow $f2$ sends half of its traffic on the path $P3$ while the other half remains on $P4$. This way traffic is balanced on all the paths and retain coding advantages.

$TE = (TE_{no-coding} + TE_{coding})\,/2$

ie., $TE = 1453$ units

$E_R = 200 - 50 - 1(Tr)$(transient energy) $= 149$ for all the nodes on paths $P2$ and $P3$ except corner nodes. While nodes on the paths $P1$ and $P4$ drain 50 units of energy to route 50 units of traffic.

The corner nodes in the network have dissipated 100 units of energy and still have 100 units for further communication.

Lifetime aware routing reduces total energy consumption by coding part of the traffic. Though the total energy consumed in this case is slightly higher than the case where simply coding is applied, it guarantees longer network lifetime by transmitting the traffic through all available paths.

	Total Energy Spent	Residual Energy Distribution	Network Lifetime
No-Coding	High	Non-Uniform	Medium
Coding Aware	Low	Non-Uniform	Low
Lifetime Aware	Medium	Uniform	High

Table 3.1   General trends in three routing strategies

3.2   Energy minimization using network coding

3.2.1   Multi-Path Routing

Our goal in this section is to impose lifetime constraints on the network and minimize total energy consumed by all the nodes in the network. The problem can be stated as follows:

Given a traffic matrix and a set of routes for each source-destination pair, our goal is to determine the amount of traffic on each path with the objective of minimizing the total energy consumed subject to lifetime constraints while making use of the network coding.

We consider multi-path routing in our model. The above problem of minimizing total energy consumption subject to lifetime constraints can be modeled as a Linear Programming (LP) problem. We use the following notations as given below.

- v: set of nodes.

- a,b,c,d...: denote the node number.

- F: input set of demand

- P_i: set of paths for i^{th} demand.

- $s(i)$: source node of ith demand.

- $d(i)$: source node of ith demand.

- P_{ij}: j^{th} path of ith demand

- f_{ij}: amount of traffic routed on path P_{ij} for i^{th} demand

- t_i: amount of traffic to be routed for the ith demand.

- E_b: Energy consumption of node b in the network.

- E'_b: Initial energy of node b.

- $\gamma^{b,c}$: amount of traffic on edge bc.

- $\gamma^{a,b,c}$: amount of coded traffic onto the edges ba and bc transmitted by node b.

- CS_k denotes the clique space.

Objective function:

Minimize:

$$\sum_{\forall b \in v} E_b \tag{3.1}$$

where,

$$E_b = \sum_{i \in F} \sum_{j \in P_i, b \in P_{ij}, b \neq d(i)} f_{ij} - \sum_{\forall ba, bc \in e} \gamma^{a,b,c} \tag{3.2}$$

The objective of the linear program formulated above is to minimize sum of total energy consumed by each sensor node while routing the given traffic. Energy consumed by each node (E_b) includes the term $\sum_{i \in F} \sum_{j \in P_i, b \in P_{ij}} f_{ij}$ in Equation 3.2. This term corresponds to the aggregate of the flow values for all demands where node b is present in the path chosen by flow. Consider illustration shown in Fig. 3.2. The node 4 is in the paths P_{11} and P_{14} chosen by demand 1 and in the path P_{21} chosen by demand 2. It is involved in routing all traffic forwarded on the paths P_{11}, P_{14} and P_{21}. Node 4 can code flows on the paths P_{11} and P_{21} and broadcast on the links $4 \rightarrow 1$ and $4 \rightarrow 6$ simultaneously. This broadcast traffic is a coded traffic and is given by term $\sum_{\forall ba, bc \in e} \gamma^{a,b,c}$ in the equation. Subtracting this term from the first term ensures that broadcast traffic is counted only once.

Flow constraint:

$$\sum_{j \in P_i} f_{ij} = t_i \tag{3.3}$$

The flow constraint given by Equation 3.3 ensures that sum of the flow values for demand i routed through multiple paths is equal to the amount of traffic to be routed for the ith demand. In the illustration shown in Fig. 3.2, for demand 1 ($1 \rightarrow 7$) available paths are P_{11}, P_{12}, P_{13} and P_{14}. The total traffic routed through these four paths should be equal to the traffic demand from $1 \rightarrow 7$ given by t_1. Similarly, for demand 2, traffic routed through paths P_{21}, P_{22} and P_{23} should be equal to t_2.

Broadcast traffic constraint:

$$\gamma^{a,b,c} \leq \sum_{i \in F} \sum_{j \in P_i, ab \in P_{ij}, bc \in P_{ij}} f_{ij} \tag{3.4}$$

$$\gamma^{a,b,c} \leq \sum_{i \in F} \sum_{j \in P_i, cb \in P_{ij}, ba \in P_{ij}} f_{ij} \tag{3.5}$$

Broadcast traffic refers to the coded traffic in the present formulation. $\gamma^{a,b,c}$ is the broadcast traffic sent by node b on the links $b \rightarrow c$ and $b \rightarrow a$ after coding. Fig. 3.3 illustrates the broadcast traffic constraint. Node 4 receives unicast (uncoded) traffic from nodes 1, 6 and 5. Paths P_{11}, P_{21} share links $1 \leftrightarrow 4$ and $4 \leftrightarrow 6$. So, coding is possible at node 4 on the flows routed through paths P_{11} and P_{21}. The broadcast traffic on links $4 \rightarrow 1$ and $4 \rightarrow 6$ should be less than the unicast traffic sent by nodes 1 and 6 on the links $1 \rightarrow 4$ and $6 \rightarrow 4$, respectively.

The terms $\sum_{i \in F} \sum_{j \in P_i, ab \in P_{ij}, bc \in P_{ij}} f_{ij}$ in Equation 3.4 and $\sum_{i \in F} \sum_{j \in P_i, cb \in P_{ij}, ba \in P_{ij}} f_{ij}$ in Equation 3.5 correspond to the total unicast traffic forwarded on the links $a \rightarrow b$, $b \rightarrow c$ and $c \rightarrow b$, $b \rightarrow a$, respectively. The constraints in these equations ensure that node b can at most

code a minimum of the unicast traffic.
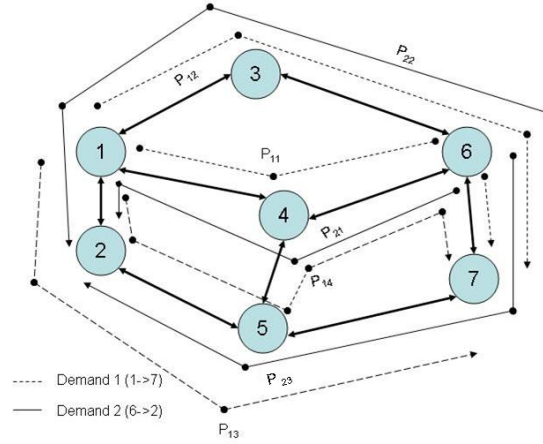
Figure 3.2   Illustrates LP formulation in the network with flow demands
from nodes 1 to 7 and 6 to 2
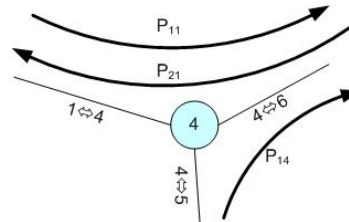
Figure 3.3   A part of the network for illustrating constraints in LP formu-
lation

Unicast traffic constraint:

$$\gamma^{b,c} = \sum_{i \in F, s(i)=b} \sum_{j \in P_i, bc \in P_{ij}} f_{ij} + \sum_{\forall a \in v} (\sum_{i \in F} \sum_{j \in P_i, ab \in P_{ij}, bc \in P_{ij}} f_{ij} - \gamma^{a,b,c}) \tag{3.6}$$

Unicast traffic is the uncoded traffic forwarded on a link. Traffic routed through a link can
be unicast or broadcast. The unicast traffic on the link $b \rightarrow c$ is given by $\gamma^{b,c}$. In Fig. 3.3,
link $4 \rightarrow 6$ is shared by paths P_{11} and P_{14}. Node 4 cannot code traffic on path P_{14} ($5 \rightarrow 4$,

$4 \rightarrow 6$) because not a single flow choose path containing links $6 \rightarrow 4$ and $4 \rightarrow 5$. Therefore the traffic it receives from node 5 must be forwarded to node 6 on the link ($4 \rightarrow 6$) using unicast transmissions. The unicast traffic on this link also includes uncoded traffic at node 4 on the path P_{11}.

In Equation 3.6 $\sum_{\forall a \in v}(\sum_{i \in F} \sum_{j \in P_i, ab \in P_{ij}, bc \in P_{ij}} f_{ij})$ is the total traffic received by node b from its neighbor nodes and routed through the link $b \rightarrow c$. The broadcast traffic $\sum_{\forall a \in v} \gamma^{a,b,c}$ is deducted to calculate unicast traffic on the link $b \rightarrow c$. The unicast traffic $\gamma^{b,c}$ also includes flows originated at node b and forwarded on the link $b \rightarrow c$. It is given by $\sum_{i \in F, s(i)=b} \sum_{j \in P_i, bc \in P_{ij}} f_{ij}$ in the equation.

Capacity constraint:

$$\sum_{ab, xy, yz \in CS_k} \frac{\gamma^{a,b} + \gamma^{x,y,z}}{C} \leq 1, \forall CS_k \tag{3.7}$$

The capacity constraint ensures that at the MAC layer, all the transmissions sharing the same medium are not scheduled simultaneously. In the present example, transmissions at node 4 interrupt all the transmissions at nodes 1, 5 and 6. Node 4 shares communication medium with these nodes. All unicast and broadcast transmissions at node 4 should be carefully scheduled such that they do not effect other nodes' transmissions.

In Equation 3.7 the possible transmissions which share clique with node b is given by $\sum_{ab, xy, yz \in CS_k} \gamma^{a,b} + \gamma^{x,y,z}$. This value should be less than capacity C of the link.

Lifetime constraint:

$$E'_b - E_b \geq \eta E'_b, \forall b \in v \tag{3.8}$$

The formulation without Equation 3.8 encourages network coding opportunities in the network for the given traffic load and aims at minimizing the total energy consumption. Consequently, it

aggressively routes demands with the aim of maximizing energy savings. The lifetime constraint in Equation 3.8 ensures a minimum required network lifetime for each individual node and hence saves the network from dying. The minimum energy stored at each node is η times the initial energy after routing the given traffic.

This constraint in a way tries to achieve uniformity in residual energy distribution across the nodes in the network. A minimum residual energy requirement ensures that no single node in the network is completely run out of its battery power. In this way network can stay alive for longer duration satisfying the traffic demands.

3.2.2   Single Path Routing

A flow that cannot be split needs to be routed in single path. Single path routing for our model is NP-complete. Due to space constraint we could not provide relevant proof here. We provide Integer Linear Programming (ILP) formulation for solving single path routing problem in this section.

We use the following additional symbols in the formulation.

- P_i: denotes the single path obtained for routing demand i from its source $s(i)$ to destination, $d(i)$.

- $y_{b,c,i}$: binary variable, which indicates the existence of traffic for demand i on the edge (b, c). $y_{b,c,i}$ is 1 only when edge bc belongs to the path for demand i. For any other edge this term is 0.

- $x_{a,b,c,i}$: binary variable, which denotes whether path P_i consists of consecutive edges (a, b) and (b, c). It takes the value of 1 when consecutive edges (a, b) and (b, c) are used, and 0 otherwise.

Minimize:

$$\sum_{\forall b \in v} E_b \tag{3.9}$$

where,

$$E_b = \sum_{i \in F} \sum_{\forall (b,c) \in E} (y_{b,c,i} * t_i) - \sum_{\forall ab, bc \in E} \gamma^{a,b,c} \tag{3.10}$$

Subject to:

$$\forall i, \sum_{(a,b) \in E} y_{a,b,i} - \sum_{(b,c) \in E} y_{b,c,i} = \begin{cases} \text{-}1 & \text{if } b = S(i) \\ 0 & \text{if } s(i) \neq b \neq d(i) \\ 1 & \text{if } b = d(i) \end{cases} \tag{3.11}$$

$$\gamma_{a,b,c} \leq \sum_{i \in F} (x_{a,b,c,i} * t_i) \tag{3.12}$$

$$\gamma_{a,b,c} \leq \sum_{i \in F} (x_{c,b,a,i} * t_i) \tag{3.13}$$

$$\gamma^{b,c} = \sum_{i \in F, s(i) = b} t_i + \sum_{\forall a \in v} (\sum_{i \in F} (x_{a,b,c,i} * t_i) - \gamma^{a,b,c}) \tag{3.14}$$

$$\sum_{ab, xy, yz \in CS_k} \frac{\gamma^{a,b} + \gamma^{x,y,z}}{C} \leq 1, \forall CS_k \tag{3.15}$$

$$E'_b - E_b \geq \eta E'_b, \forall b \in v \tag{3.16}$$

The above formulation accepts a set of demands and a network topology (set of edges and nodes). It minimizes the total energy consumption while obtaining a single path for each demand and respecting the channel capacity constraints. The objective function in Equation 3.9 minimizes the total energy consumption and is similar to the objective function presented in the previous section (see Eq. 3.1). The energy consumption of each node in the network is

calculated in Equation 3.10 and is similar to the Equation 3.2. However, Equation 3.10 uses the individual binary variables to estimate the traffic on each edge.

Equation 3.11 ensures that a single valid path is selected for each demand i. This is achieved by balancing the number of edges selected for each demand about any intermediate node. As shown in the equations the cases for source and destination nodes are handled separately.

Equations 3.12 and 3.13 estimate the amount of coded traffic on the consecutive edges ab and bc. In Equation 3.12, the term on the right hand side refers to the total traffic that is routed in the direction, a, b, c and the corresponding term in Equation 3.13 denotes the total traffic routed in the direction, c, b, a. Both traffic components use the two edges ab (or ba) and bc (or cb) and the coded traffic is the minimum of these two components. Here, the term $(x_{a,b,c,i} * t_i)$ is the amount of unicast traffic for demand i on the consecutive edges ab and bc.

Equation 3.14 calculates the total unicast traffic, $\gamma_{b,c}$ forwarded on the edge bc. It is equal to sum of total traffic generated at node b forwarded on the link bc and the uncoded traffic transmitted on link bc. The traffic that is not coded is calculated by deducting broadcast traffic $\sum_{\forall a \in v} \gamma^{a,b,c}$ from the total unicast traffic $\sum_{\forall a \in v} (\sum_{i \in F} (x_{a,b,c,i} * t_i)$ received at node b from all the links incident onto it.

Finally, Equations 3.15 and 3.16 denote the capacity and lifetime constraints of the problem and are similar to Equations 3.7 and 3.8 in the previous formulation.

3.3 Performance Evaluation

3.3.1 Experimental Setup

We evaluated the proposed schemes to compare their relative performance by varying the traffic and channel capacity conditions. The formulated LP problems are evaluated using the ILOG CPLEX 10.100 software. The presented scheme was evaluated on a grid topology of size $10 * 10$ with unit normalized distance between adjacent nodes. Our choice of grid topologies for the performance evaluation studies is based on the fact that they offer better network coding opportunities compared to other general topologies as a result they offer a good platform to evaluate the network coding and its trade offs. For each evaluation run, we generated 20 traffic

demands with randomly chosen source and destination nodes.

We measured the normalized total energy consumption (normalized with respect to the scenario where no coding is performed and each demand is routed via the shortest path) and the average standard deviation to compare the relative performance of the following two schemes: Energy minimization using network coding (EM-LP) and energy minimization with a lifetime constraint (LTC-LP). The first scheme had no lifetime constraint while the second scheme did consider a lifetime constraint.

The following parameters were varied in our evaluation studies:

- Channel capacity factor: ratio of channel capacity to the total demand traffic in the network;

- Traffic load factor: ratio of total demand traffic to the channel capacity; the amount of traffic is kept fixed while varying the channel capacity factor and while the channel capacity is fixed when varying the traffic factor;

- Lifetime constraint (η).

3.3.2   Results

Effect of channel capacity on energy consumption:

Fig. 3.4 compares the relative performance of LTC-LP-10 (LTC-LP with $\eta = 0.10$) and LTC-LP-5 (LTC-LP with $\eta = 0.05$) with the EM-LP. As channel capacity is increased for a given traffic load, more traffic can be sent via the same path while still respecting the MAC constraints. In other words, with the increasing channel capacity factor the opportunity for network coding increases as a result, and the normalized energy consumption for all schemes decreases in the figure. However, as we further increase the channel capacity for LTC-LP we hit a point where we do not show further energy reductions. This is because of the lifetime constraints imposed on the individual nodes, which constrain the amount of traffic routed via each path in the network. This trend is particularly seen in LTC-LP-10 where the lifetime constraints on the individual nodes are higher. The EM-LP curve also stabilizes after exploiting
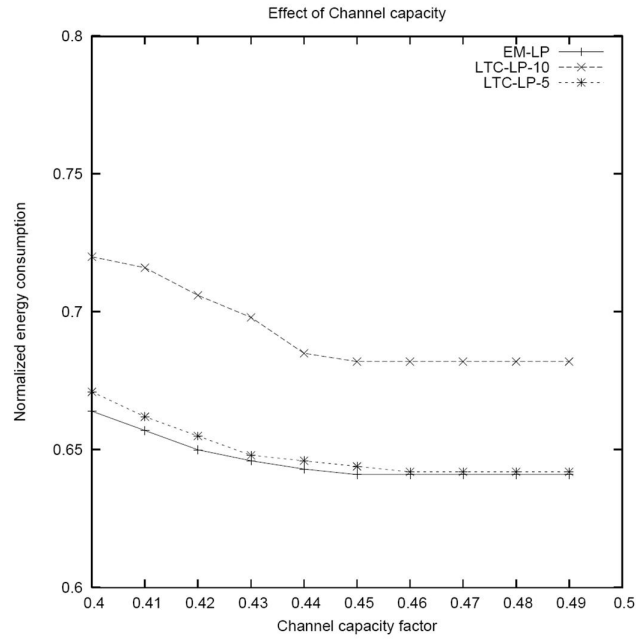
Figure 3.4   Figure shows the effect of channel capacity on energy consumption

all network coding opportunities as shown in the figure. EM-LP shows more than 30% improvement over the non-coding based routing, the LTC-LP-10 shows about 27% improvement over the non-coding based routing.

Effect of traffic factor on energy consumption:

Fig. 3.5 compares the relative performance of LTC-LP (with $\eta = 0.20$) and EM-LP schemes by varying the traffic factor. As the traffic load is increased for a given channel capacity, the network coding opportunity initially increases as more traffic is available to perform better traffic mixing. However, as the traffic is further increased the energy reductions due to network coding are surpassed by the increase in energy consumption from the increased workload. Consequently, both schemes show an increased normalized energy consumption after the traffic factor of 0.6. In this figure, the LTC-LP and EM-LP schemes show an improvement of 28%
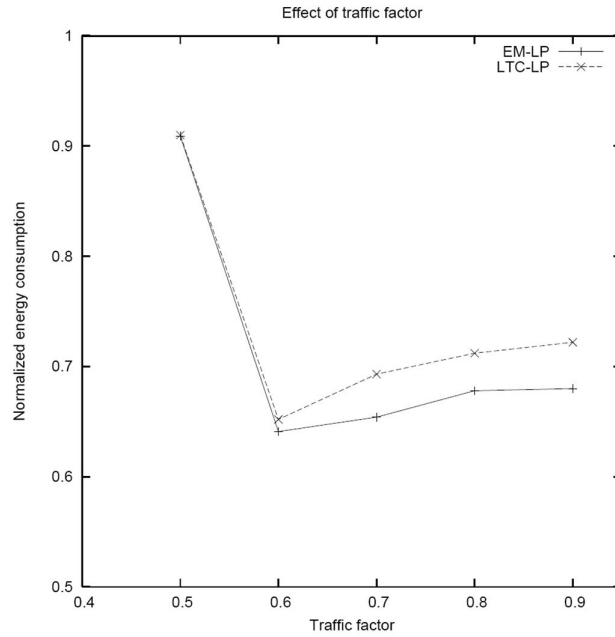
Figure 3.5   Figure shows the effect of traffic factor on energy consumption

and 35% over the non-coding based routing at traffic factor 0.9 .

Effect of lifetime constraint on energy consumption:

Fig. 3.6 compares the relative performance of the EM-LP scheme with LTC-LP at different values of η. As η is increased, the traffic is constrained to be more widely spread. As a result, the amount of network coding decreases. Consequently, the total energy consumption increases. However, throughout both the schemes show significant energy savings over the no-coding based routing scheme.

Effect of traffic factor on residual energy distribution:

Fig. 3.7 depicts the effect of traffic factor on the standard deviation of the remaining energy values of the individual nodes in the network. The standard deviation values are normalized with a suitable and large enough number. In the figure, the EM-LP shows a huge

Figure 3.6   Figure shows the effect of lifetime constraint on energy consumption

standard deviation implying that the residual energies among the individual nodes after handling the given traffic are very unevenly distributed. As the traffic i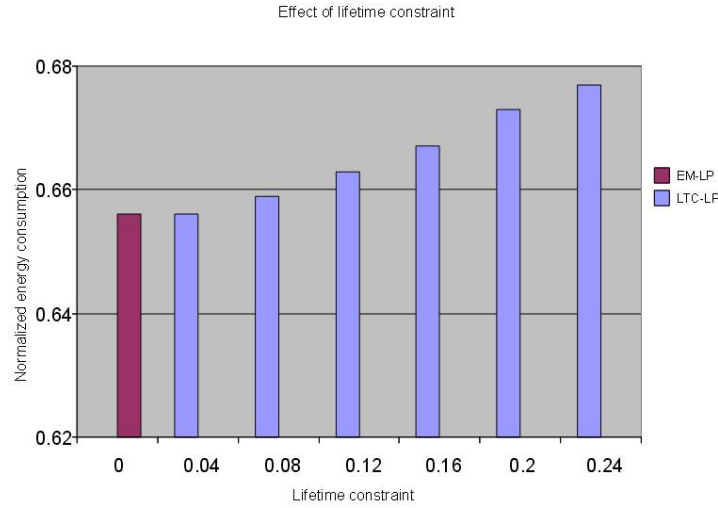ncreases, the traffic is more widely spread and hence the standard deviation decreases for both the schemes as shown in the figure. Throughout, the LTC-LP ($\eta = 0.20$) scheme which tries to spread the traffic across the network to meet the lifetime constraints achieves better residual energy distribution.

Effect of channel capacity factor on residual energy distribution:

Fig. 3.8 compares the residual energy distributions of LTC-LP (with $\eta = 0.20$) and EM-LP schemes by varying the channel capacity for a given traffic. As the channel capacity is increased the opportunity for network coding increases as a result there is more room for the traffic to be routed via a single path and achieve higher coding gains. However, this results in more skewed traffic distribution and higher standard deviation values. However, as we further increase the traffic the LTC-LP scheme stabilizes due to energy constraints and limits the traffic load distribution from further skewing. Throughout, the LTC-LP ($\eta = 0.20$) shows better standard deviation values than the EM-LP scheme.
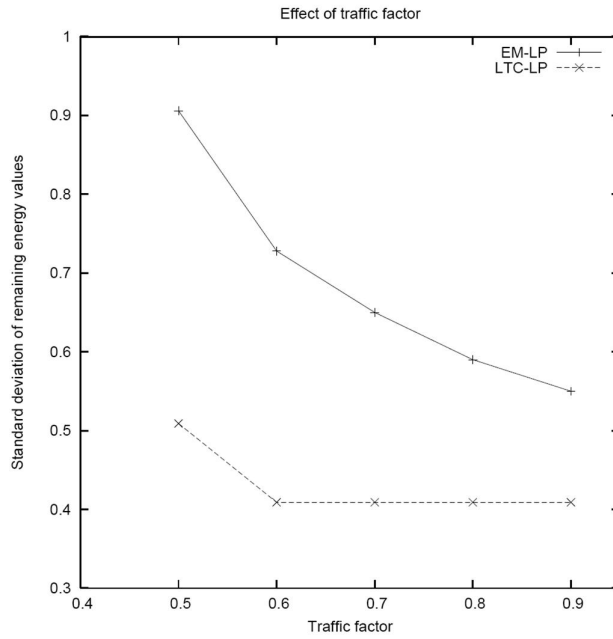
Figure 3.7 Figure shows the effect of traffic factor on residual energy distribution
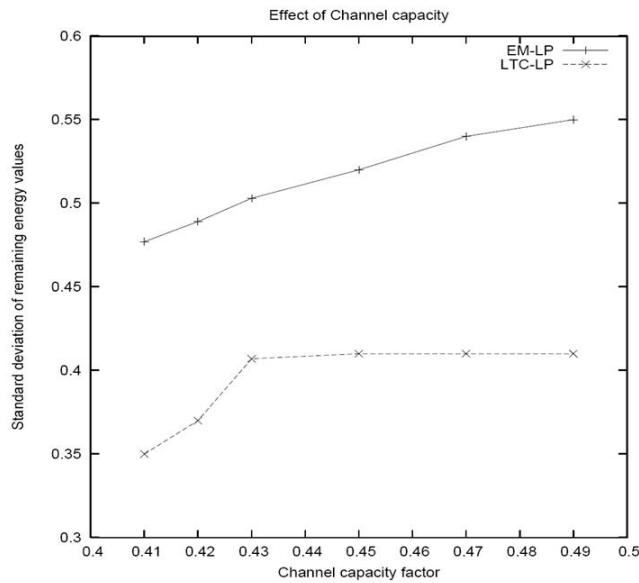
Figure 3.8 Figure shows the effect of lifetime constraint on energy consumption

CHAPTER 4. Neighborhood Graph Transformation to Determine Coding Regions in a Network

In this chapter, we explore a mechanism and develop an algorithm using a graph transformation of a neighborhood map of a node to determine coding opportunities at the node (24). A node and its single-hop neighbors are first converted into a neighborhood map (called neighborhood graph) of that node. This is a unique graph with respect to the node under consideration. The node set of a neighborhood graph consists of the node under consideration and all of its single-hop neighbors. Edges between the selected node and its neighbors are part of the edge set for the graph. In addition, any edges between any two neighbors of the selected node are also part of the edge set. We limit our discussion in this section to this neighborhood graph (G_x) for the selected node x. Fig. 4 is the neighborhood graph of node x. The neighborhood graph is transformed into a transformed graph for determining the coding opportunities.
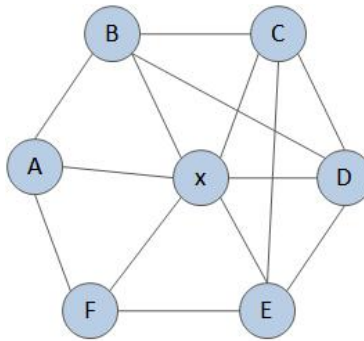
Figure 4.1   Neighborhood Graph $(G_x(V_x, E_x))$

We use the following notations.

Notations:

- $N = (U, L)$ represents the network with vertices U and edges L.

- x is the central node at which coding opportunity is being analyzed.

- $G_x = (V_x, E_x)$ denotes undirected neighborhood graph with vertices V_x and edges E_x for node x.

- $X(V_x', E_x')$ represents the transformed bidirectional graph of $G_x(V_x, E_x)$.

- u, v, w, and $x \in V_x$ denote node in the graph G_x. These nodes in the transformed graph X are denoted by u', v', w', and z', respectively.

- $(u, v) \in E_x$ implies a unidirectional edge between node $u \in V_x$ and node $v \in V_x$. The $(u', v') \in E_x'$ denotes edge from node u' to node v' in X.

- $N(u)$ denotes single hop neighbors of node $u \in V_x$.

- $n(N(u))$ denotes the total number of neighbors of u.

- $P_{u \to x \to v}$ denotes a path through node x between $u, v \in N(x)$.

Figure 4.2   Transformed Graph $(X(V_x', E_x'))$

Graph transformation: Let $G_x(V_x, E_x)$ be the neighborhood graph defined at node x. A transformed graph $X(V_x', E_x')$ is obtained as follows. The node set V_x' is same as V_x. Node $v \in V_x$ in graph X denotes the node corresponding to node $v \in V_x$ in G_x. The edge set of X is

defined as $E'_x = \{(u', v'), \forall u', v' \in V'_x\} - E_x$. The undirected edge $(u, v) \in E_x$ can be split into bidirectional e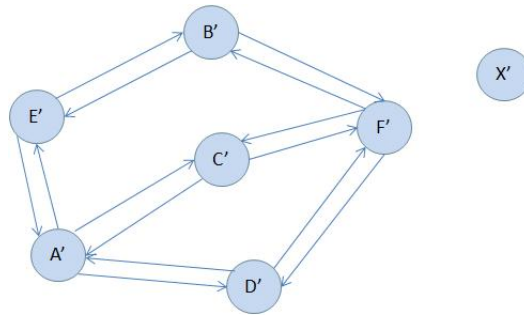dges $u \rightarrow v$ and $v \rightarrow u$. While determining edge set of E'_x, each undirected edge in E_x is split into two edges. Thus edges that do not exist in G_x is present in X, and vice versa.

Definition 1 *Coding number gives the number of packets that can be coded at central node and transmitted over the network in a single transmission.*

Definition 2 *Coding groups give the total number of coding combinations possible at a node.*

Lemma 1 *An edge $(u', v') \in E'_x$ in transformed graph represents the shortest path of hop-count two from node u to node v via central node x in the original graph G_x .*

In graph G, central node x is directly connected to all its neighbors. There exists a path $P_{u \rightarrow x \rightarrow v}$ that contains edges (u, x) and (x, v). Let $(u', v') \in E'_x$ in X. According to the transformed graph construction an edge in X represents no-edge in G. This implies that there exists no direct edge between node u and node v in graph G. If there was such an edge then it contradicts the statement that edge (u', v') is present in transformed graph X. Since each node in G is connected to central node, path $P_{u \rightarrow x \rightarrow v}$ exists between node u and node v through central node x.

Lemma 2 *If the shortest path between $u' \rightarrow v'$ is of length two and is given by $P'_{u' \rightarrow v' \rightarrow w'}$ then it implies that traffic from $u \rightarrow v$ can be coded with traffic from $v \rightarrow w$ at central node x.*

An edge $(u', v') \in E'_x$ in X implies that there is no edge from u to v in G. Similarly, an edge (v', w') means that there is no edge from v to $w \in E'_x$ in X. According to *Lemma 1* these edges represent paths $P_{u \rightarrow x \rightarrow v}$ and $P_{v \rightarrow x \rightarrow w}$, correspondingly in graph G_x. Fig. 4.3 shows these paths in G_x and corresponding edges in X.

Since no edge (u', w') in X implies that an edge (u, w) exists in G_x, node w overhears traffic from the node u in G_x. Similarly, node v can buffer packets transmitted to node w. Thus packets coded at central node x can be retrieved at both nodes v and w.
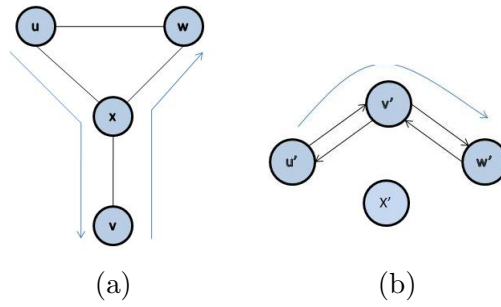
Figure 4.3   Figure to illustrate Lemma 2

Hence, a path length of *two* in X corresponds to routes in G_x that can be coded successfully at central node.

Lemma 3 *Any two edges (u', v') and (w', z') in X, which do not share a common vertex, and the vertices are not connected imply that traffic through path $P_{u \to x \to v}$ can be coded with traffic on path $P_{w \to x \to z}$ at node x.*
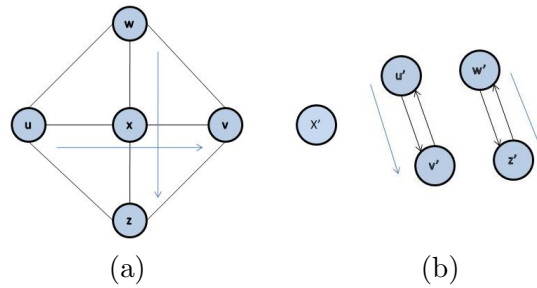
Figure 4.4   Figure to illustrate Lemma 3

Consider (u', v') and (w', z') edges in X shown in Fig. 4.4b. These *four* vertices are unique. According to *Lemma 1* both these edges represent paths $P_{u \to x \to v}$ and $P_{w \to x \to z}$, correspondingly with node x as intermediate shown in Fig. 4.4a. Suppose that central node x combines traffic on these two paths and broadcast. In order to retrieve desired packets from the coded packet node v and node z need to overhear transmissions from nodes w and u, respectively i.e. (v, w)

$\in E_x$ and $(z, u) \in E_x$ in G. Since edges (u', v') and (w', z') do not share any vertex, $(v', w') \notin E'_x$ and $(z', u') \notin E'_x$ in X.

This proves that any two edges without any common vertices represent paths that can be coded in G at central node.
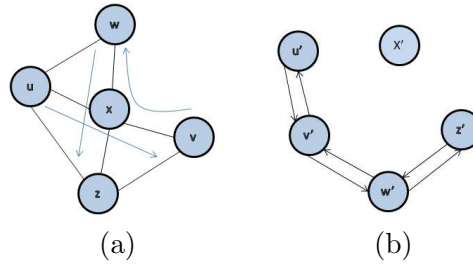
Figure 4.5   Figure to illustrate Lemma 4

Corollary 1 *More coding opportunities can be derived from paths satisfying Lemma 2 and Lemma 3 by coding with additional paths that pass through node x and follow below conditions:*

- *Receiving nodes in newly added paths should be in the broadcast range of transmitting nodes of coding paths.*

- *Receiving nodes in coding paths should be in the broadcast range of transmitting nodes in new paths.*

Lemma 4 *A path of length greater than two in transformed graph indicates that packets through paths corresponding to each edge in transformed graph cannot be coded and transmitted together in single transmission at central node.*

We assume that contradiction of the theorem is true i.e. it is possible to code traffic through paths at central node, which form a path of length greater than *two* in transformed graph. Consider a path of length *three* in X shown in Fig. 4.5b. Let (u', v'), (v', w') and $(w', z') \in E'_x$. These edges correspond to paths $P_{u \to x \to v}$, $P_{v \to x \to w}$, $P_{w \to x \to z}$, respectively in G shown in Fig. 4.5a.

According to *Lemma 2* any *two* consecutive edges correspond to paths that can be coded at central node x in G. Since we assumed that *three* packets can be coded, each receiving node should have *two* packets from the other two nodes in addition to the packet coded. In our example, to retrieve desired packet node v should be in the broadcast range of node w i.e $(v, w) \in E$ (*Corollary 1*). This implies that $(v', w') \notin E_x$. But this contradicts our assumption that $(v', w') \in E_x$. Hence edges in a path of length *three* in X do not correspond to paths that can be coded simultaneously at node x in G_x.
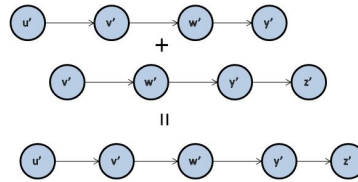
Figure 4.6   Transformed graph with path length four

Let us assume edges in a path of length k in X constitute a group of k paths in G_x, where packets through these paths cannot be coded at node x. Now, consider path of length $k + 1$ in X. Since k packets cannot represent set of k paths that can be coded in G_x, addition of one more path to this set will not improve the coding ratio. This can be seen through an illustration shown in Fig. 4.6. In this example, path of length *four* in X is result of *two* paths of length *three*. Since we have already proved that path of length *three* cannot represent coding flows in G_x, edges in path of length *four* do not correspond to coding paths.

By induction, it can be said that packets through paths with length greater than *two* in X cannot represent coding flows in G_x.

Corollary 2 *The upper bound on the number of packets that can be coded at a central node is $n[N(x)]$, where x is the central node.*

Theorem 1 *The coding groups for coding number $\leq n[N(u)]$ can be derived from the transformed graph.*

The coding opportunity at a node is dependent on the node degree of neighbors (*Corollary 2*). In the transformed graph it is equivalent to the maximum of ($total\ size - node\ degree(u')$) $\forall\ u' \in V$.

From *Lemma 2* and *Lemma 3* coding groups for coding number *two* can be determined from transformed graph. The same relationship cannot be extended to explore more coding opportunities (*Lemma 4*).

The central node checks whether certain conditions are met for receiving node to retrieve desired packet successfully. While choosing coding number greater than *two*, following conditions need to be strictly followed with respect to transformed graph (*Corollary 1*). Suppose the edge (u', v') corresponds to path $P_{u \to x \to v}$ in G_x.

- Vertex u (as tail) should not form edge with a node which is head of the edge already chosen.

- Similarly, v (as head) is not connected to any vertex that is tail of the edge already chosen.

We prove the claim that coding groups for coding number $i \leq$ node degree can be derived from transformed graph X. Let k denote number of packets that can be coded in a single transmission and $E(x')$ contains edges in X corresponding to the paths in G_x that can participate in coding at central node x.

In *Lemma 2* and *Lemma 3 two* coding number scenarios are discussed. The transformed graph can determine all such possibilities. Let us assume that X can determine coding group of size k. $E(x')$ contains k edges corresponding to the paths that form a coding group of size k at node x. We have to prove that this claim is true for coding number $k + 1$.

The set E'_x- $E(x')$ gives list of edges that need to be explored to determine whether any edge in this set corresponds to path that can be coded along with k paths. Let us assume that there exists a path $P_{u \to x \to v}$ in G_x at central node x. The corresponding edge is (u', v') $\notin E(x')$. For path $P_{u \to x \to v}$ to be coded along with k, conditions discussed above need to be satisfied. This implies that u' does not form edge with any node, which already shares edge

$\in E'_x$ with other vertex. Similarly, v' is not connected to any vertex that is part of the edge in E'_x. If the edge (u', v') satisfies these conditions, path $P_{u \to x \to v}$ in G_x can be coded along with the coding group of size k. Thus, paths corresponding to coding number $k + 1$ can be determined from the transformed graph, if it exists.

Thus, by induction it can be said that paths corresponding to any coding number $\leq n[N(u)]$ can be derived from the transformed graph.

CHAPTER 5.   Algorithm to Determine Coding Opportunity in a Network

5.1   Algorithm

In the previous chapter we discussed how graph transformation of a node neighborhood map can be used to determine coding opportunities at a node. In this section we discuss algorithm to rank a node in the network based on the coding ratio discussed earlier.

Below are the notations used in describing the algorithm:

- $P(i)$ contains paths that are selected for coding, where i denotes the coding number $\geq$ 2.

- H contains edges in E_x' that can be selected for coding.

The algorithm provides methodology to determine coding opportunity at a node. This is used to compute rank of a node, which implies coding capability of a node. In most of the cases high ranked nodes can code packets more frequently and efficiently than less ranked nodes.

The algorithm in Fig. 5.1 explains the procedure to determine coding capability of any node in a network. Steps 1 to 7 explain the procedure to determine scenarios for coding number 2. Initially one edge e is chosen and all other edges are scanned to identify if they can or cannot be used for coding. The set to be considered is K. In each iteration, one edge from E_x' is chosen, and all the edges that share vertices are removed from K, as explained in step 3 and step 4. Also edges that incident on receiving capable nodes are removed in step 4. At the end of each iteration, edges corresponding to paths that can be coded at central node x is appended to the set $P(2)$.

Similar procedure need to be followed for coding number greater than 2, explained in steps 8 - 12. The possible combinations for coding number i require set $P(i-1)$ as input because

i packets coding scenarios constitute $i-1$ flows and an additional flow. After choosing an element a in step 8, the set K contains edges in E_x', which are not present in a. Corresponding to each edge in a, remove edges from K as explained in steps 3 and 4. After step 12 an element $a \in P(i)$ contains i edges corresponding to paths in original graph that can be coded at the central node. The number of entries in $P(i) \ \forall \ i > 2$ is used to rank a node. These ranks are assigned relatively in the network.

End of Round	Set H	Selected element	Set K	Coding groups
0	$\{(u',v'),\ (v',u'),\ (v',w'),\ (w',v'),\ (w',z'),\ (z',w')\}$		$\{(u',v'),\ (v',u'),\ (v',w'),\ (w',v'),\ (w',z'),\ (z',w')\}$	$\{\}$
1	$\{(v',u'),\ (v',w'),\ (w',v'),\ (w',z'),\ (z',w')\}$	(u',v')	$\{(v',u'),\ (v',w'),\ (w',z'),\ (z',w')\}$	$\{(u',v'),(v',u')\},\{(u',v'),(v',w')\},$ $\{(u',v'),(w',z')\},$ $\{(u',v'),(z',w')\}$
2	$\{(v',w'),\ (w',v'),(w',z'),\ (z',w')\}$	(v',u')	$\{(w',v'),\ (w',z')\}$	$\{(v',u'),(w',v')\},\{(v',u'),(w',z')\}$
3	$\{(w',v'),(w',z'),\ (z',w')\}$	(v',w')	$\{(w',v'),\ (w',z')\}$	$\{(v',w'),(w',v')\},\{(v',w'),(w',z')\}$
4	$\{(w',z'),\ (z',w')\}$	(w',v')	$\{(z',w')\}$	$\{(w',v'),(z',w')\}$
5	(z',w')	(w',z')	$\{(z',w')\}$	$\{(w',z'),(z',w')\}$
6	$-$	$\{(u',v'),\ (v',u')\}$		

Table 5.1   Illustration of Algorithm to determine coding opportunity at a node.

This algorithm is explained through a simple illustration in Table. 5.1. Consider a network in Fig. 4.5a. In this example node x is of degree $four$, so the maximum coding ratio possible is $four$. The transformed graph of node x is shown in Fig. 4.5b.

In Table. 5.1 initial value of variables is shown in round 0. The round 1 to round 5 calculates scenarios where two packets can be coded together. The paths stored in $P(2)$ are

used as input for the round in 6, where scenarios with coding ratio 3 is computed. In this example, we see that there are no three packet coding scenarios.

Even if a node has high coding ratio packets mixing may not occur in the same proportion. The reason might be due to less amount of traffic routed through the node or absence of complement flows, which can be coded and transmitted together. The topology of a network changes due to node mobility. All these factors affect the coding pattern at a node. In the present work we do not consider dynamic behavior of the network.

Algorithm

For coding number $i = 2$

STEP 1*:* $H = E'_x$.

STEP 2*:* Choose edge e, where $e = (u, v) \in H$. $K = H\text{-}\{e\}$.

STEP 3*:* Remove edges from K that originate from u.

STEP 4*:* Remove edges from K that incident on v.

STEP 5*:* $P(2) = \{e, e_i\}, \forall e_i \in K$.

STEP 6*:* $H = H\text{-}\{e\}$.

STEP 7*:* Repeat STEPS 2 - 6 until H is empty. The set $P(2)$ contains all *two* packet coding scenarios at central node.

For coding number $i > 2$

STEP 8*:* Select an element $a \in P(i-1)$. The set $P(i-1)$ contains all the combinations for coding number $i - 1$.

STEP 9*:* $K = E'_x$ - $\{e_i\} \forall e_i \in a$.

STEP 10*:* Remove edge (u', v') from K,

- either (u', x') $\in E'_x$ and $(w', x') \in a$.

- or $(y', v') \in E'_x$ and $(y', g') \in a$.

STEP 11*:* $P(i) = \{a, e_i\}, \forall e_i \in K$.

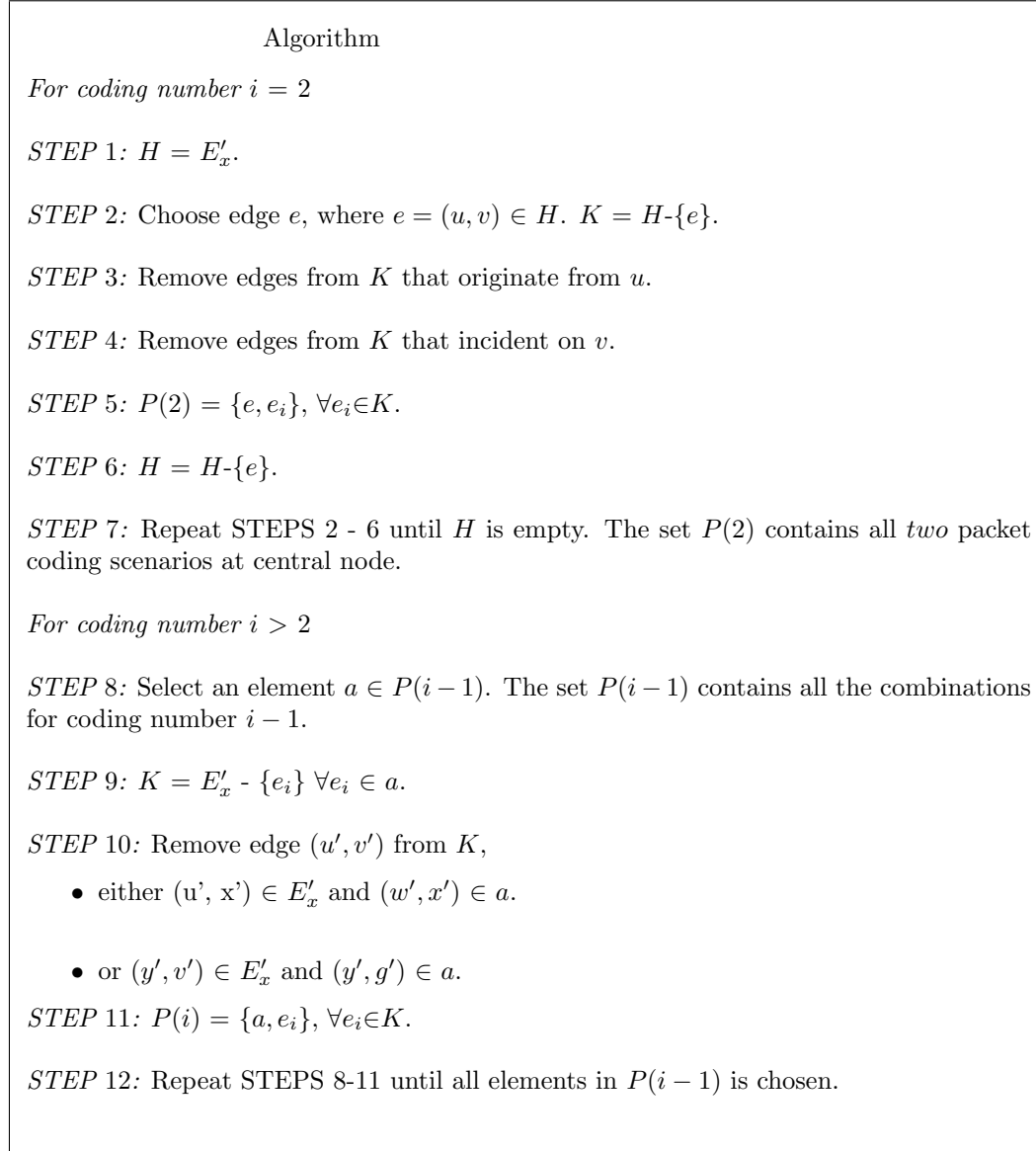*STEP* 12*:* Repeat STEPS 8-11 until all elements in $P(i-1)$ is chosen.

Figure 5.1   Algorithm for determining coding opportunity at central node

CHAPTER 6.   Simulation Results

6.1   Experimental Setup 1

In this chapter, we experimentally evaluated performance of the proposed algorithm in determining coding regions in a network. For this purpose, we considered network topologies in Fig. 6.1a and Fig. 6.2a. The topology shown in Fig. 6.1a is part of the university network with 24 nodes. To validate our scheme with other networks we considered spin topology with 14 nodes (Fig. 6.2a).

The network in Fig. 6.1 and Fig. 6.2 consists of nodes with capability to code multiple packets in a single transmission. But coding ratio varies among nodes in the network. The nodes with similar coding ratio are grouped, given identical rank and are shown using the same color in the graph.
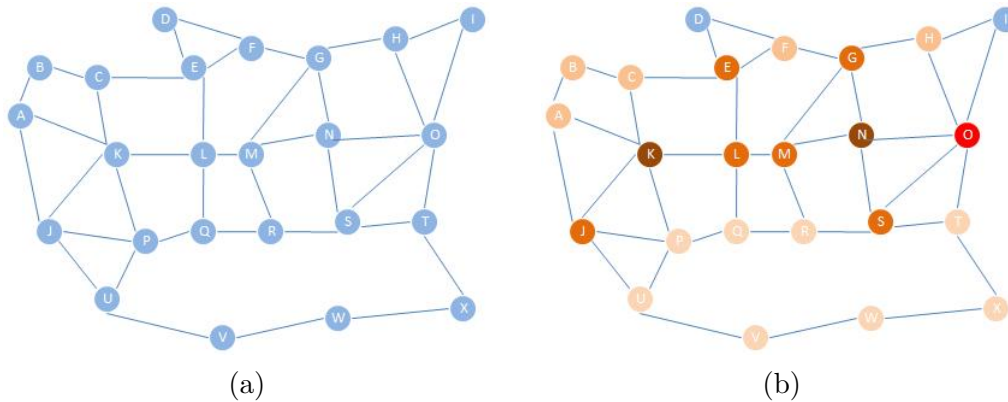

Figure 6.1   a) ISU Network b) Labeled nodes in Network

The results in Fig. 6.3 shows the number of nodes in each group against coding ratio for network in Fig. 6.1a. Higher the rank of a node, higher is the coding opportunity. From the
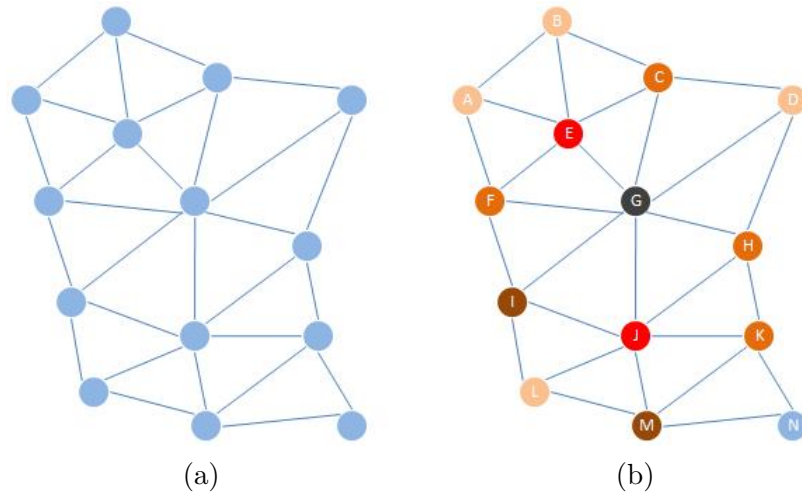
Figure 6.2   a) Spin Network b) Labeled nodes in Network

topology we observe that nodes with same rank have similarities in neighborhood graph and degree. For example, nodes O, K, N have coding ratio above the other nodes in the network. If we observe neighbors of these nodes and the connections between them, it is evident that the node degree is high and majority of the neighbors are connected. The direct connectivity between the neighbors contributes to the coding gains from opportunistic listening, in addition to the opportunistic coding.

Similar observations can be derived from the result in Fig. 6.4. This topology is more suited for coding because the gain from opportunistic listening dominates in this network. For example, consider node E and node J. Most of the neighbors of these nodes are connected; thereby providing opportunity for coding more number of packets in a single transmission. We make the same observation with respect to the situation of node G. The high degree of node G results in high coding ratio compared to any other node in the network.

The nodes D and I in Fig. 6.1b and N in Fig. 6.2b are not ranked by our scheme. The reason is because the neighborhood graphs of these nodes form a fully connected graphs. Thus these nodes are not chosen as intermediary in normal scenario.

6.2   Experimental Setup 2

From the results obtained in earlier section, we observe that node degree and edges between the neighbors are crucial parameters to determine the coding opportunity at a node. If these network statistics are used as a metric in existing routing protocols, we can derive coding gains with lower transmission overheads.
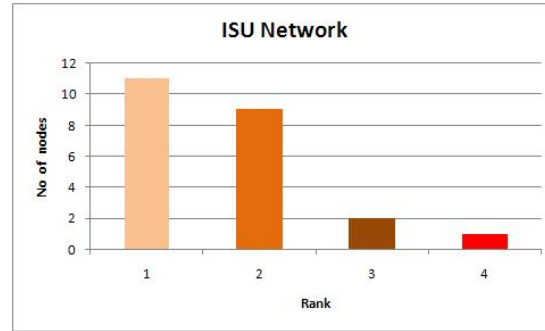

Figure 6.3   Ranks against number of nodes in ISU Network
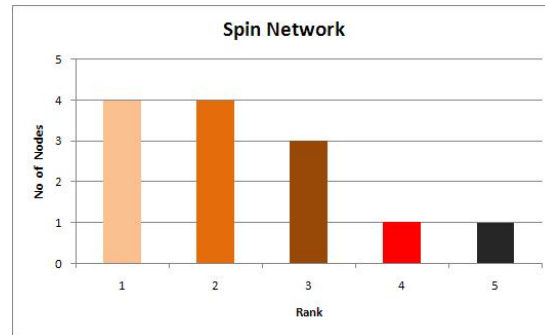
Figure 6.4   Ranks against number of nodes in Spin Network

In order to prove the relationship between coding opportunity and the parameters discussed above, we randomly generated 100 connected network topologies with 20 nodes and 50 links. The degree of each node and the connectivity degree among the pairs of neighbor nodes (edges connecting the neighbor) are computed for each node in these graphs. We compare our simulation based results from the experimental data on coding opportunity with these two parameters.

The result in Fig. 6.5a shows the coding groups possible at a node against the edges between

(a) Coding groups v/s Neighbor edges　　(b) Coding groups v/s Node degree

Figure 6.5　Figure shows variation of coding ratio on network parameters
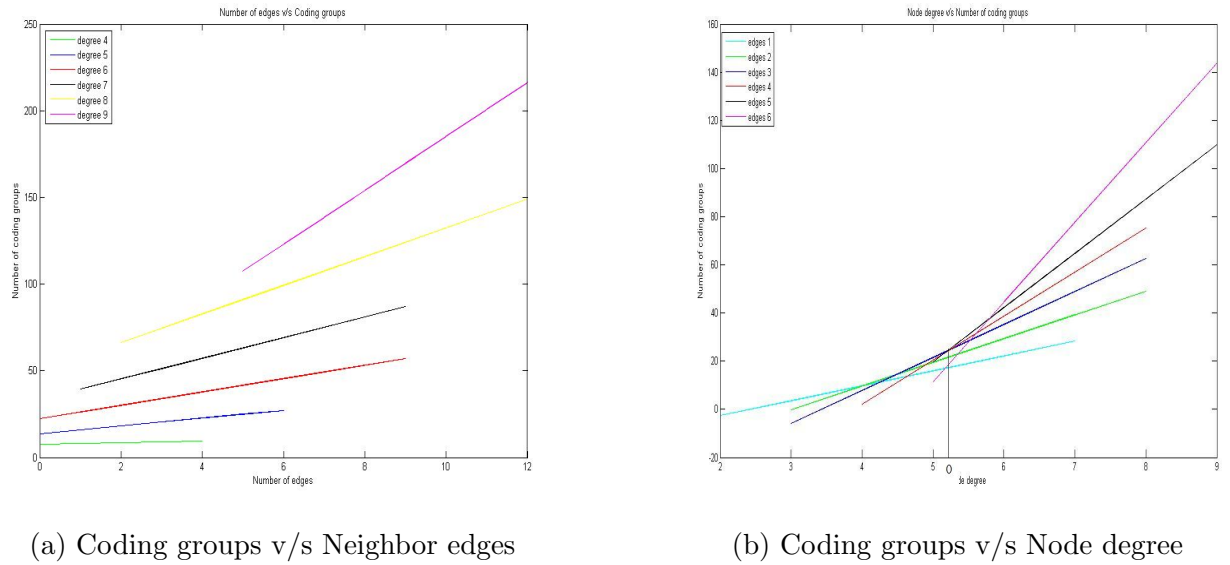
the two neighboring nodes for different node degree. We observe that number of coding groups increase with the number of edges between the neighbors of a node. This behavior is due to opportunistic listening and is common among all the nodes. The slope of the graph increases with node degree. This shows that for the same number of neighbor edges connected, higher the node degree more is the coding possibility. With increase in node degree, the number of possible paths with the node being as the intermediate node increases. Thus the probability for coding also improves.

Next, we consider the dependence of coding groups on node degree as shown in Fig. 6.5b. We observe that in most of the scenarios nodes with high degree and higher number of edges between neighbors have high coding ratio. However, this behavior is not observed until the marked point O in the graph. In this region, node with lower number of edges between the neighboring nodes has a tendency to code more packets. This unusual behavior is explained with an example below.

Consider a graph shown in Fig. 6.6a. In this example, at node 0 two packets coding can take place in four different scenarios, which are $P_{1->3}$, $P_{3->1}$), $(P_{3->2}, P_{2->3})$, $(P_{1->3}, P_{3->2})$ and $(P_{3->1}, P_{2->3})$. Now suppose we add an edge from node 1 to node 3 as shown

in Fig. 6.6b. As a result number of two packets coding scenarios at node 0 decrease to *one*, i.e. $(P_{3->2}, P_{2->3})$. If we add one more edge between nodes 2 and 3 as shown in Fig. 6.6c, then no paths are available that choose node 0 as an intermediate node. From this example and results discussed above it is evident that when node degree is low, lower number of edges between neighboring nodes result in more coding gains.
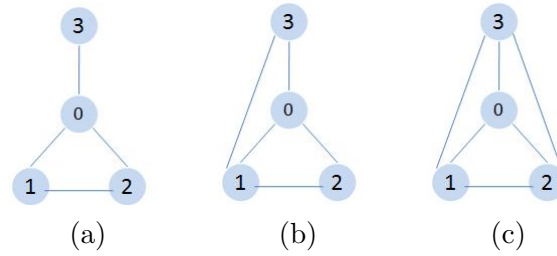

Figure 6.6   Figure shows the unusual behavior when node degree is less

These observations help in deriving a light-weight approach for determining coding groups in a network with a very good accuracy. The distributed protocols can be built using this metric to design efficient coding-aware routing algorithms. This would reduce the additional overhead incurred in improving coding gains in network. In the next section we describe heuristic solution to calculate coding degree of each node in the network based on the neighborhood graph.

6.3   Heuristic solution to determine coding ratio

In the earlier section we showed that coding degree of a node largely depends on node degree and the way neighbors are connected. This input is provided to each node when it is configured in the network. We propose a heuristic solution to determine coding degree of each node in the network based on these factors and show how routing can be done using this information.

We calculate weight for each node. The higher the value higher is the chance for coding. Let d_A gives the degree of node A and n_A is the number of edges between the neighbors. The weight of the node A (W_A) is calculated as below:

$$W_A = d_A(1 + n_A) + \sum_{u \in V_A} d_u/2 \tag{6.1}$$

The first term in Equation 6.1 is dependent on the node degree and number of neighbor edges. While the second term is half of the sum of neighbors node degree. This terms helps in routing the packets to the nodes surrounded by the high coding degree nodes. This heuristic considers the network parameters and also the degree of neighbors.
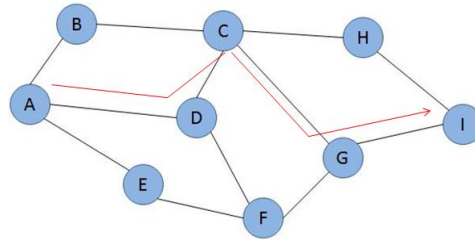
Figure 6.7   Illustration to show the working of Heuristic algorithm

Node	Weight	Coding groups
A	6.5	3
B	5.5	1
C	9	6
D	8	3
E	5	1
F	7	3
G	7.5	3
H	5	1
I	4.5	1

Table 6.1   Compares the values derived from algorithm with the weights
calculated from heuristic solution

The above heuristic is explained through an example in Fig. 6.7. We compare the weights obtained through this method with the values from the algorithm proposed in the earlier section. The comparison is shown in the Table 6.1 against each node. We observe that in most of the cases weights assigned are proportional to the number of coding groups that can be derived from the method proposed earlier.

The weights assigned can be used as a metric to route packets between any two nodes in the network. Since the information required is available locally, routing can be done in a distributed manner. For example, consider node A as source and node I as destination. There are many paths available from A to I. At node A, weight of the node D is higher than any of the neighbor nodes. So node A chooses node D as next-hop. Similarly, node C is chosen over node F to route packets. Since node G has high weight than node H, it is chosen as next-hop by node C to route the packets to destination I. If the process is repeated with coding degree as metric derived from our earlier algorithm, the same path $A \rightarrow D \rightarrow C \rightarrow G \rightarrow I$ would have been chosen (see Table 6.1).

The dynamic behavior of the network such as node mobility, traffic patterns etc., effect the coding behavior. In this paper we proposed the approach to determine coding degree of a node in the network statically. The periodic execution of algorithm with the information about the traffic flows and node mobility can give good results in dynamic environment. We will investigate this in our future work.

Conclusions and Future work

Energy minimization and throughput enhancement (25) - (28) are the key performance enhancements that can be achieved through network coding in wireless networks. In this report we showed that network coding, a technique originally introduced to maximize throughput, can be used as a technique to save energy by minimizing the number of transmissions needed in a wireless network to deliver a packet. We explored the trade off between selecting paths where network coding can be used and network lifetime, which is adversely effected when aggressive use of this technique creates hot spots where nodes die quickly disconnecting the network.

This exploration was done by comparing the transmission energy use by shortest path routing with the results of LP one of which aggressively used the technique (EM-LP), the other guaranteed network lifetime by ensuring all nodes retained a specified fraction of their energy (LTC-LP). Both saved energy when compared to shortest path routing. The saving provided when network lifetime is guaranteed is slightly less and residual energy more evenly distributed, since traffic is distributed amongst more paths.

The additional overhead and unbounded delays incurred in routing the packets stand as obstacles for scaling the throughput and lifetime gains beyond certain limit. In this report we have developed a technique to determine coding regions in a network. This would help network designers to plan in such a way that the performance gains dominate.

In this work, we also have constructed a neighborhood map and its transformed graph at a node to evaluate its coding potential. We theoretically showed that the transformed graph can derive all possible coding opportunities at a node. We also developed an algorithm, which determines coding number of each node in a network. We experimentally evaluated the proposed scheme on different topologies. We showed that nodes with identical topology show

similarities in coding ratio. For example, the topology depicted in Fig. 6.2 is more suitable for coding than the topology depicted in Fig. 6.1 because of the structures of the two networks.

We further evaluated our algorithm on random topologies and derived the relationship between the coding ratio and node degree, and the coding ratio and the number of edges between the neighbors of a node. The dependency relationship varies based on both the node degree and the number of between the neighbors of a node. We showed that when the node degree is low, a lower number of neighbor edges result in more coding gains, which is an unusual scenario.

This research is primarily focused to determine whether applying network coding is beneficial in a network. The results discussed in this report can be used to evaluate the tradeoff between gains achieved and additional overhead incurred. Distributed routing protocols can be designed using the light-weight approach discussed in the report for enhanced throughput and lifetime. In the future, we would like to investigate the effect of dynamic behavior of network on the coding degree.

BIBLIOGRAPHY

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, " Network information flow," IEEE Transactions on Information Theory, Vol. 46, No. 4. (2000), pp. 1204-1216.

[2] S. R. Li, R. W. Yeung, and N. Cai, " Linear network coding," In IEEE Transactions on Information Theory," 2003.

[3] R. Koetter and M. Medard, " An algebraic approach to network coding," IEEE/ACM Transactions on Networking, 2003.

[4] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," In SIGCOMM. Pisa, Italy: ACM, September 2006.

[5] Y. Wu, S.M. Das, and R. Chandra, "Routing with a Markovian Metric to Promote Local Mixing,"' MSR Technical Report, Nov 2006.

[6] S. Sengupta, S. Rayanchu1, and S. Banerjee, "An Analysis of Wireless Network Coding for Unicast Sessions: The Case for Coding-Aware Routing," In proc. of IEEE INFOCOM, 26th IEEE International Conference on Computer Communications, May 2007.

[7] Nishanth Gaddam, Anil Gathala, David Lastine and Arun Somani, "Energy Minimization through Network Coding for Lifetime Constrained Wireless Networks," In 17th International Conference on Computer Communication and Networks (ICCCN), Aug 2008.

[8] P. Chaporkar and A. Proutiere, "Adaptive network coding and scheduling for maximizing throughput in wireless networks,", In proc. 13th Annual ACM conference on mobile computing and networking, 2007.

[9] J. Goseling and J.H. Weber,"Energy Benefit of Network Coding for Multiple Unicast in Wireless Networks," In proc. 29th Symposium on Information Theory in the Benelux, 2008.

[10] T. Cui, L. Chen, and T. Ho, "Energy Efficient Opportunistic Network Coding for Wireless Networks,"' In proc. of IEEE INFOCOM, May 2008.

[11] J. Le, J.C.S. Lui, and D.M. Chiu, "DCAR: Distributed Coding-Aware Routing in Wireless Networks," In proc. of IEEE ICDCS, 2008.

[12] C.-H. Liu and F. Xue, Network coding for two-way relaying: rate regions, sum rate and opportunistic scheduling, In proc. of IEEE ICC, May 2008.

[13] C. Fragouli, J. Widmer, and J.-Y. Le Boudec, "A network coding approach to energy efficient broadcasting: From theory to practice," In proc. of IEEE INFOCOM, Apr. 2006.

[14] P.S. David and A. Kumar, "Network coding for TCP throughput enhancement over a multi-hop wireless network," In proc. of COMSWARE, Jan. 2008.

[15] S. Omiwade, R. Zheng, and C. Hua, "Practical Localized Network Coding in Wireless Mesh Networks,"' In proc. of IEEE SECON, June 2008.

[16] J. Liu and D. Goeckel, and D. Towsleyv, "Bounds on the gain of network coding and broadcasting in wireless networks,"', In proc. of IEEE INFOCOM, May 2007.

[17] A.K. Haddad, and R Riedi, "Bounds on the Benefit of Network Coding: Throughput and Energy Saving in Wireless Networks," In proc. of IEEE INFOCOM, May 2008.

[18] Le Jilin, John Lui and and Dah Ming Chiu "How Many Packets Can We Encode?-An Analysis of Practical Wireless Network Coding," In proc. of IEEE INFOCOM, May 2008.

[19] S. Rayanchu, S. Sen, J. Wu, S. Banerjee, and S. Sengupta, "Loss-aware network coding for unicast wireless sessions: Design, implementation, and performance evaluation," In proc. ACM SIGMETRICS, June 2008.

[20] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing," in Proc. ACM SIGCOMM, Aug. 2007.

[21] C. Fragouli, D. Katabi, S. Katti, A. Markopoulou, M. Medard, and H. Rahul, "'Wireless network coding: Opportunities and challanges,"' In proc. of Military Communications Conference, 2007.

[22] F.R. Dogar, A. Phanishayee, H. Pucha, O. Ruwase, and D.G. Andersen, "Ditto: a system for opportunistic caching in multi-hop wireless networks," In proc. of ACM MobiCom, Sept. 2008.

[23] A. Mahmino, J.Lacan and C. Fraboul,"Guaranteed Packet Delays with Network Coding," In proc. of IEEE SECON, June 2008.

[24] Nishanth Gaddam and Arun Somani, "Determining coding structures in a wireless network," submitted to ACM MobiCom 2009.

[25] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," In proc. of IEEE INFOCOM, June 2002.

[26] Y. Yang, B. Krishnamachari, and V.K. Prasanna, "Energy-latency trade offs for data gathering in wireless sensor networks," In proc. of IEEE INFOCOM, March 2004.

[27] J. H. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad-Hoc Networks," In proc. of IEEE INFOCOM, March 2000.

[28] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," In ACM MobiCom, September 2003.