# ENERGY-EFFICIENT EVEN-PARITY NETWORK CODING FOR ADHOC NETWORKS

A Thesis by

Jyothsna Yarlagadda

B.Tech, Gudlavalleru College of Engineering, JNTU, 2008

Submitted to the Department of Electrical Engineering and Computer Science
and the faculty of the Graduate School of
Wichita State University
in partial fulfillment of
the requirements for the degree of
Master of Science

December 2012

ENERGY-EFFICIENT EVEN-PARITY NETWORK CODING FOR ADHOC NETWORKS

The following faculty members have examined the final copy of this thesis for form and content, and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science with a major in Electrical Engineering.

________________________________________

Ravi Pendse, Committee Chair

________________________________________

John M.Watkins, Committee Member

________________________________________

Bob Minaie, Committee Member

DEDICATION

To my husband Siva Rama Krishna

# ACKNOWLEDGEMENTS

First, I would like to express my deepest gratitude to my research adviser Dr. Ravi Pendse for his guidance, patience, encouragement, and support throughout this study.

I would also like to extend my gratitude to my thesis committee members: Dr. John Watkins and Dr. Bob Minaie.

My sincere thanks to Amarnath Jasti, Vijay Ragothaman, Tish Best, and Nagaraja Thanthry from ANRC whose stimulating suggestions helped me in writing this thesis and broadened my view and knowledge.

Last but not least, a sincere thank you goes to my family, friends, and to God for all of their unconditional love, prayers, and constant support.

## ABSTRACT

The major difference between wired and wireless networks is their broadcast nature, specifically how transmitted data by one node may reach other nodes and vice-versa. This broadcast nature is a curse for wired networks but a blessing for wireless networks. Network coding is a technique where instead of just forwarding packets arrived at relay nodes, the node will collect several packets and then combine them together using an algebraic algorithm for transmissions. Network coding reduces the energy consumption by decreasing the number of transmissions required to communicate a given amount of information across the network. The aim of this thesis is to enhance network-coding strategy in order to improve energy gain, which in turn increases throughput in ad hoc networks. To that end, this thesis also proposes an even parity scheme that reduces processing time and power of nodes by improving coding opportunities. In addition, an even parity scheme allows for the coding of large numbers of packets at a time instead of coding just two packets using normal XOR operation.

# TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

# CHAPTER 1

# INTRODUCTION

Wireless, in its various forms, is an increasingly important communication medium. It provides the means for network-wide internet connectivity, mobility, disseminated sensing, etc. Wireless networks in most cases offer only one hop wireless connectivity through access points, which then have to connect to the internet through wired links. Therefore, to provide widespread coverage, entire buildings have to be wired, which is quite costly and difficult to maintain.

Wireless ad hoc networks have been proposed to provide a way to avoid those issues, where nodes communicate with each other without involving any central access points. However, next generation applications are extremely challenging. These challenges include video over internet, large file sharing, etc. These applications require high throughput and performance, which cannot be achieved by current wireless ad hoc networks.

## 1.1 Challenges in Wireless Networks

Emerging applications of wireless networks put a huge demand on the limited resources currently available. Network access for these applications is available over a wireless link as opposed to a wired link. Consider an ad hoc wireless network in which communication happens through multi-hop nodes. These nodes are responsible for carrying data from source to destination. Therefore, these wireless nodes experience lower bandwidth and higher error rates. The bandwidth is a major concern in wireless networks; therefore, appropriate measures have to be taken to make more efficient use of bandwidth. Second, the battery life of wireless nodes is of great concern in rescue areas where flooding of redundant information consume huge amount of power, which directly affects the battery life of wireless nodes. Third, the wireless link may be viewed as unreliable in terms of accessibility. Users may not have access to the network because of their mobility away from the coverage area. In addition, interference is another factor that plays a major role in wireless networks.

Finally, mobile devices will likely have inadequate processing and power capabilities compared to desktop PC's. For most of the application, there is a tradeoff between power consumption and the amount of data that can be collected. For example, Wi-Fi requires more stream more continuous data than ZigBee with high power consumption. On other hand, ZigBee can run for years on batteries by limiting amount of data they collect and transmit at a required frequency. Therefore, power consumption and amount of data reception depends on applications.

One of the crucial factors with wireless communication is network and application performance. For wireless networks and lower bandwidth connections; the parameters include response time, latency and reasonable throughput in the presence of high error rate. In the presence mobility, the network performance parameters include fast connection establishment and packet delivery. These are the critical performance norms for faster information retrieval and interactive communication applications.

### 1.2 Network-Coding and its Importance

Wireless ad hoc networks are often deployed as a means of data communication because of their flexible structures. Usually these are utilized in battlefields where the infrastructure availability is less. Meanwhile, wireless devices in wireless ad hoc networks have become more frequently battery powered. Therefore, energy efficiency has become a crucial factor in wireless ad hoc networks. Broadcasting is one of the frequent events in wireless ad hoc networks for the distribution of data in many applications, so there has been an increasing focus on designing an energy-efficient broadcasting algorithm for wireless nodes in ad hoc networks. Network-coding is a great breakthrough in the area of information theory, and has become increasingly popular in regard to enhancing throughput in wireless ad hoc networks. Network-coding is a technique that combines multiple data packets at intermediate node in a way that eventually recovery data at their respective destinations while leading to fewer transmissions in a medium. This can be well illustrated by the example below.

As shown in Figure 1.1, customer D wants to buy product P1 from dealer A and product P2 from dealer B. Agent C hosts advertisements for these two standard dealers. Customer D needs to pay shipping charges for both the products if he purchases those products separately from their respective dealers. Instead, agent C will cut down the shipping charges for one product if the shipment times correspond and are directed toward the same address, therefore reducing shipping charges for items purchased.

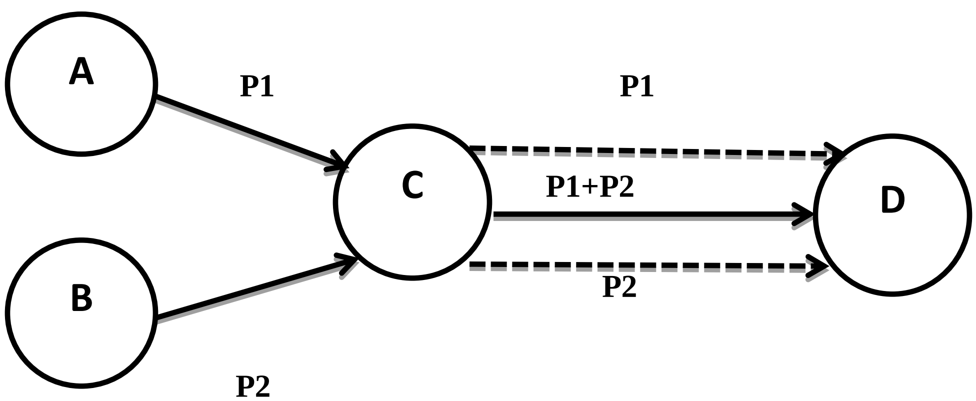



Figure 1.1. Network Coding Example

In the same way, network-coding encodes packet at an intermediate node instead of simply forwarding packets to their destinations. This process reduces number of transmissions and increases the throughput.

## 1.3 Objective

The aim of this thesis is to enhance network-coding strategy in order to improve energy gain which in turn improves throughput in wireless ad hoc networks. To that end, this thesis also proposes parity network-coding which allows coding large number of packets instead of coding overhead.

## 1.4 Thesis Organization

Chapter 2 presents network-coding and its importance with technical details, explains the importance of ad hoc routing protocols in network-coding, and emphasizes background and related work. Chapter 3 explains even-parity scheme in detail, which includes the algorithm for encoding and decoding packets in a network. Chapter 4 explains this study's algorithm with an example. We used Matlab for simulating different scenarios. It was determined that an increase in the number of nodes increases coding opportunities, which in turn decreases number of transmissions. Conclusions and discussions on future research are given in chapter 5.

# CHAPTER 2

# LITERATURE SURVEY

## 2.1 Technical Aspects on Network-Coding

Network coding is a technique where instead of just forwarding packets that arrive at relay nodes, the node will collect several packets and then combine them using an algebraic algorithm for transmissions. In network-coding, algebraic algorithms are applied at nodes to gather various transmissions. The received transmissions are decoded at their respective destinations. This means fewer transmissions are required to transmit all the data, but more processing power at intermediate and terminal nodes is required.

In traditional routing networks, packets are cached and forwarded out. Therefore, if the routing node receives two packets for the same destination, it will forward them one after other, queueing the remaining packets. This requires separate transmissions for each packet to deliver, which decreases network efficiency. In network-coding, algorithms are used to combine packets together and send out multiple messages in one transmission, then used again to decode said messages at their respective destination using the same algorithm.

### 2.1.1 Basic Working Principle of Network-Coding

To explain the working scheme of network-coding, consider Figures 2.1 and 2.2. In traditional network-coding, relay acts as a dumb hub that receives and forwards data. To forward only one message takes four transmissions, as illustrated in Figure 2.1. However, by implementing network-coding, the message from S to D and vice-versa takes only three transmissions as shown in Figure 2.2.

In network-coding, the relay node not only stores and forwards the data packets but also combines the messages to form a coded packet to broadcast. Assume the bit stream of packet S_DATA is 1001010110… in its data portion and the bit stream of packet D_DATA is 1011011101… as shown in Figure 2.2. When the relay receives packets S_DATA and D_DATA, it combines them by performing bitwise XOR to produce a coding packet, i.e T_DATA = S_DATA + D_DATA where '+' operator signifies XOR operation. After performing even parity with packets from S and D nodes, the resulting packet contains a bit stream of 0010001011…in its data portion.

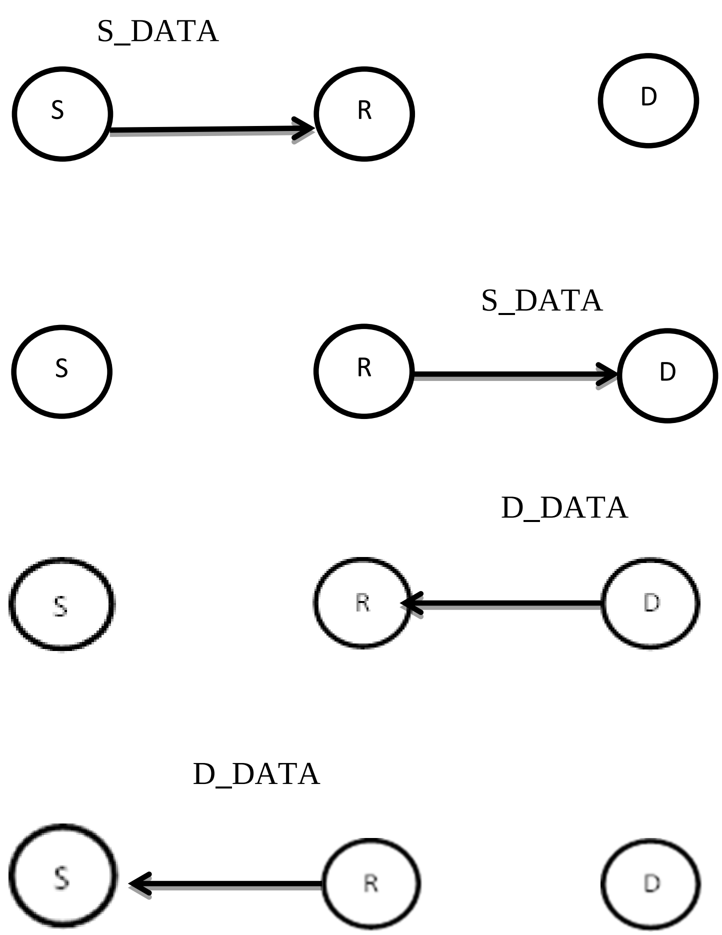



Figure 2.1. Traditional Routing

On receiving a broadcasted message, the T_DATA node S extracts D's message from T_DATA by using Node S perform even-parity with T_DATA and S_DATA as shown in Figure 2.3.

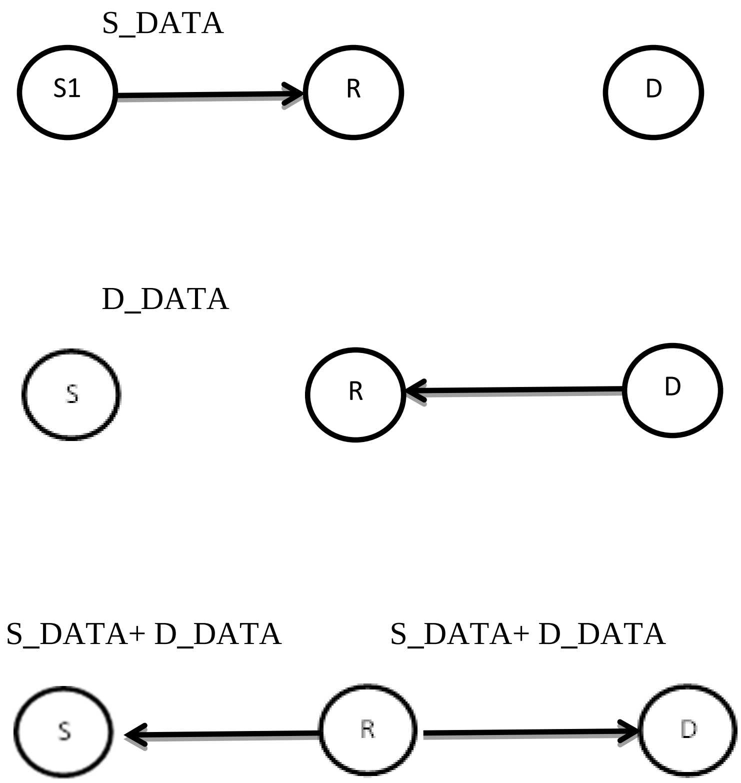



Figure 2.2. Network-Coding

Similarly, node D extracts node S's message from T_DATA by using its own message as shown in Figure 2.4. The remaining transmissions are used to send new messages that save link bandwidth and increase the throughput of the network.

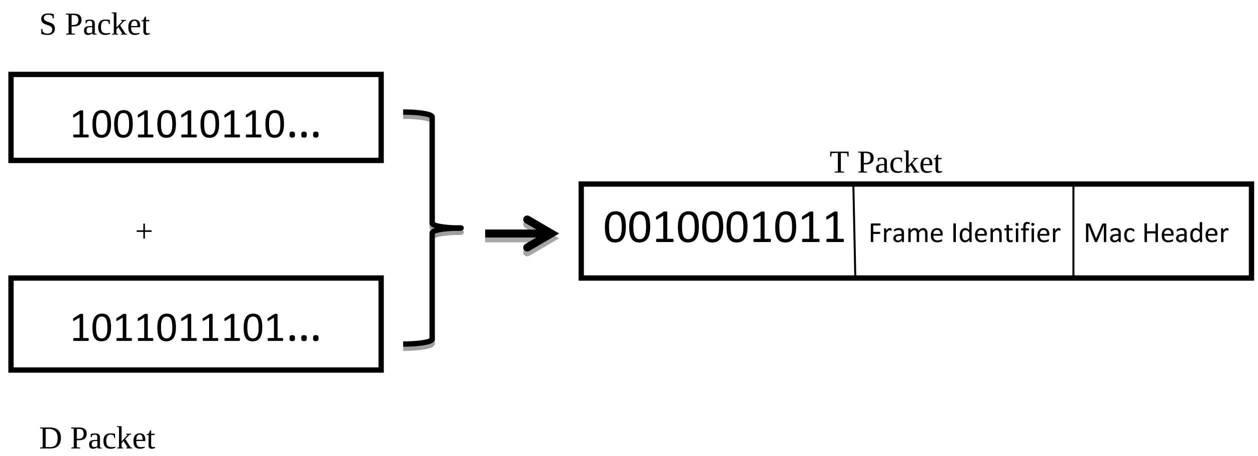



Figure 2.3. Coding Packet at Relay Node

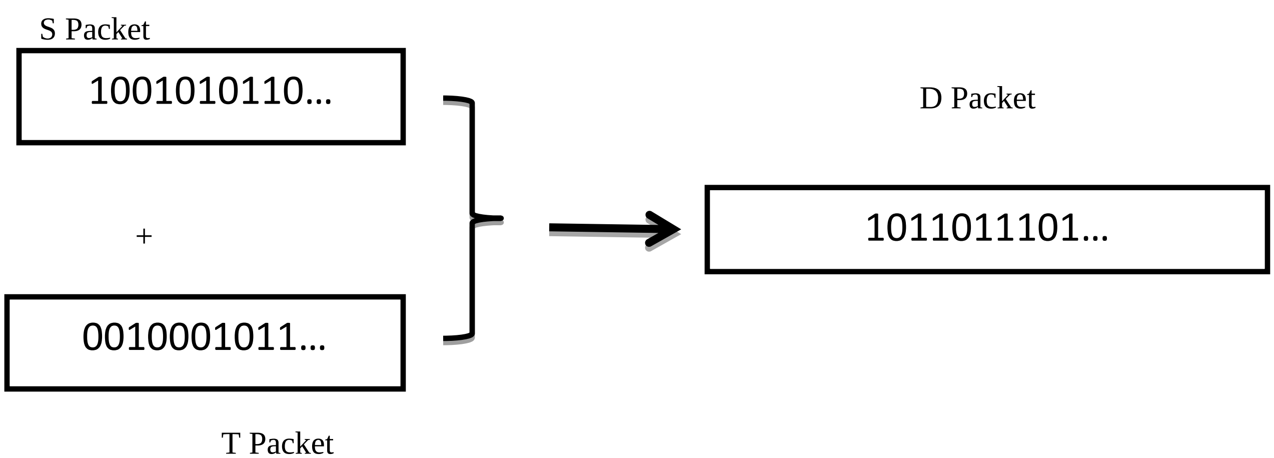



Figure 2.4. Extraction of D Packet from T Packet at Node S

Network-coding is highly dependent on traffic patterns, medium access link scheduling, and their respective topologies. For instance, Figure 2.1 shows a scenario in which Nodes S and D send their message, combine, and broadcast as a coded message. However, in real networks it may not be the same case. Consider a scenario in which Node S sends its packet first, followed by the relay node, then node D.

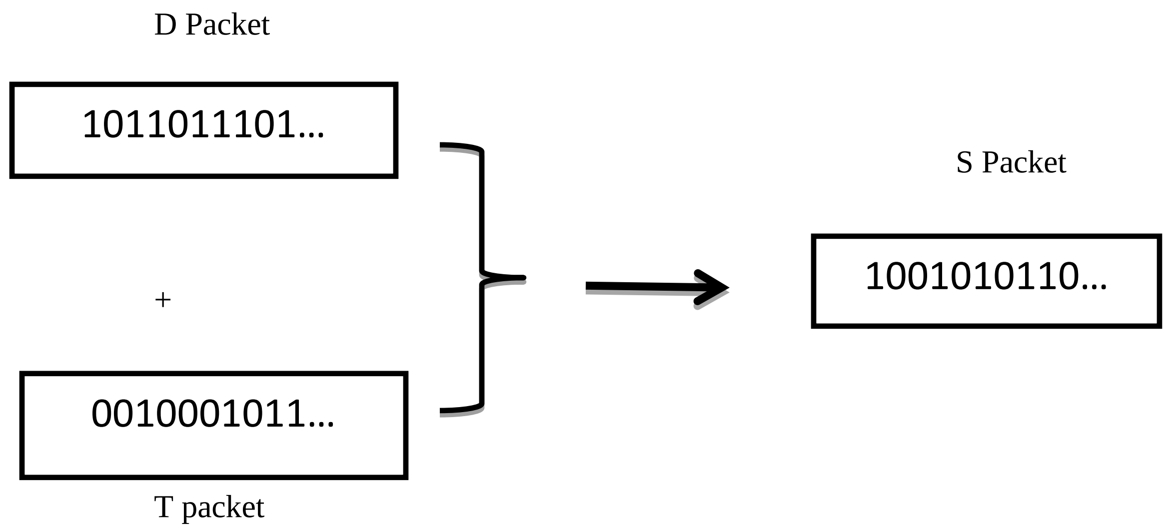



Figure 2.5: Extraction of S Packet from T Packet at Node D

In the above case, network-coding is not possible because the relay has only one message to forward. To use network-coding efficiently, previous studies considered a cooperative network-coding in which traffic flows are made to overlap to create coding chances. In the above scenario, transmission could be suspended until messages from all other nodes combine. This will help increase the network throughput and bandwidth of bottleneck links. The only drawback in this procedure is performance degradation, i.e the packets are halted at a node and it may have a long wait before being delivered to its destination.

### 2.1.2 Benefits of Network-Coding

There are many potential benefits to instigating network-coding in a wireless environment.

#### 2.1.2.1 Throughput

Throughput is a major concern in today's wireless environment. Network-coding improves throughput by coding data into a single packet to send out in one transmission instead of multiple transmissions. Coder node matches what each neighbor has with what another neighbor wants and delivers multiple packets to different sources in a single transmission. Network-coding is widely used in peer-to-peer networks. Today people want to hear their favorite songs in public places using common hot spots. Bandwidth usage plays a prominent role in such cases. Let us consider an example where C and D are customers of the same service and $S_C$ and $S_D$ are the songs each customer has on their respective devices. Customer C wants to listen to song $S_D$ and customer D wants to listen to song $S_C$. Therefore, the service point can use network-coding to send the XOR'ed version of the broadcast stream instead of wasting bandwidth in sending each stream singularly. Customers can decode the XOR'ed stream easily and can hear their respective songs at a same time. In this scenario, network-coding doubles the throughput by reducing the number of transmissions.

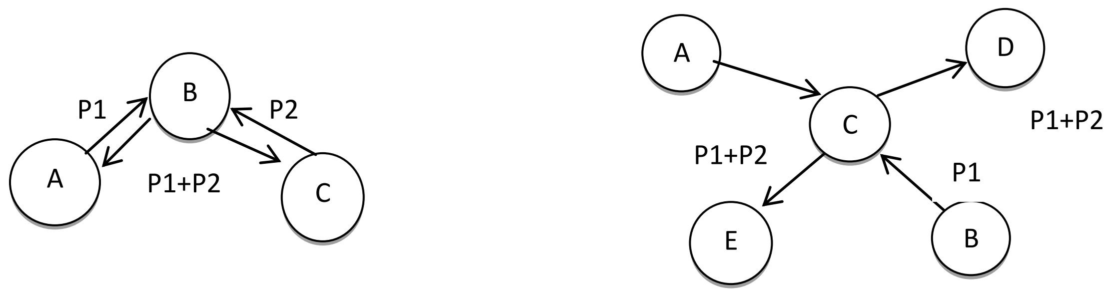
#### 2.1.2.2 Reliability

Wireless links are lossy links due to transmission errors and collisions in shared medium. Reliability means efficient retransmission of packets in case of packet loss. Network-coding is considered a more reliable method of packet delivery. To illustrate this, consider the traditional approach where the source needs to know which packet destination needs retransmission. In an unreliable environment, to search for the required node takes extra bandwidth and retransmitting the packet takes more transmissions. Network-coding generally doesn't care about individual packets and in case of retransmissions it can transmit all lost packets in one transmission.

#### 2.1.2.3 Power Consumption

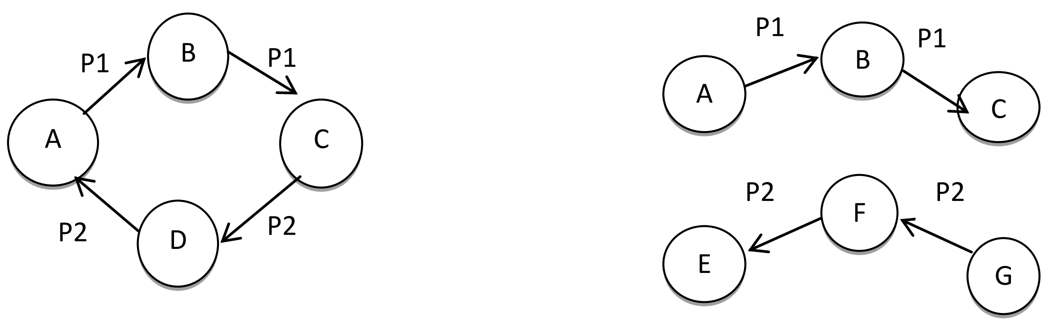
In the traditional method, power is wasted through multiple transmissions. In network-coding, savings in transmission are coupled with the possibility of an increase in processing at each node. However, using even-parity network-coding, the power consumption used by a node for processing will be greatly reduced. Even-parity network-coding takes less CPU power for its computation so it requires less power to process coding packets. Therefore, our approach takes less power than traditional network-coding.

### 2.1.3 Topologies and its Effects




(a) Chain Topology

(b) X Topology




(c) Cannot Form Chain Topology

(d) Cannot Form X Topology

Figure 2.6. Topologies

Network-coding decreases transmission time, which is illustrated by the topologies below. Figure 2.6(a) shows basic network-coding topology where node B is acting as a relay, carrying multiple messages at once without wasting transmission time and bandwidth. In the chain topology, no opportunistic listening is necessary. Another coding topology is X topology shown in Figure 2.6(b). In this topology, node B and E are in the transmission range of A and D. Via opportunistic listening, node D can overhear packets sent by node B, and node A can overhear packets sent by node E. When node C sends a XOR'ed packet to both node E and D, they will decode the packets intended to them with XOR, i.e node D will extract packet P1 sent to it by P2+(P1+P2), while node E extracts P2 by P1+(P1+P2). In both cases, without network-coding, four transmissions are required while with network-coding three transmissions are sufficient. The singular drawback to network-coding is that coding opportunities are highly dependent on topology. Topologies like Figure 2.6(c) and Figure 2.6(d) will not have any coding opportunities.

## 2.2 Ad Hoc Routing Protocols

In wireless ad hoc networks, moving packets from source to destination through mobile nodes requires a routing protocol to store routing updates. Routing protocols are divided into three categories:

- Reactive Protocols
- Proactive Protocols
- Hybrid Protocols

In proactive protocols, every node in the network maintains one or more routing tables that are updated periodically. Each node sends a broadcast message in the network when there is a topology change, which incurs huge overhead cost, but routes are available at all times. In our work, we are using proactive protocol to update routing information of every node in the network. In addition, we form a neighbor table based on this information. The following sections discuss some table-driven ad hoc protocols.

### 2.2.1 Distance-Sequenced Distance Vector (DSDV) Routing Protocol

In distance-sequenced distance vector routing protocol, every mobile node in the network maintains a routing table that contains available destinations, number of hops to reach each destination, and a sequence number assigned to that particular destination node. This sequence number is used to differentiate stale routes to that of new ones and thus avoid forming routing loops. Stations in a network periodically send their routing updates to their nearby nodes. If there is any change in its routing table then nodes send updates to their intermediate nodes. Therefore, the updates are both table-driven and event-driven. The routing table updates sent to nearby nodes is either full dump or incremental update. In full dump, the entire routing table itself is send to its neighbors, whereas in incremental update only metric changed routes are sent. When networks are stable, full dumps are not sent to avoid traffic overloading and updates are relatively infrequent. In fast-changing networks, updates are done frequently, usually as full dumps. Every route sent by transmitter (other than routing table routes) also contains a sequence number. The highest sequence number routes are stored in the routing table. If two routes have the same sequence number then the route with the least number of hop counts is used. The only concern with this routing protocol is that it requires frequent updates which uses more battery power and requires a small amount of bandwidth even in idle time.

### 2.2.2 Optimized Link State Routing Protocol

The Optimized Link State Routing Protocol (OLSR) is an optimization of link state routing protocols in which the size of control packets as well as the number of control packets transmitted is reduced. OLSR reduces control traffic overhead by using Multipoint Relays (MPR), which is an important factor in OLSR. A MPR is a node’s one hop neighbor who can be used to forward packets in the network. Instead of flooding the packets into the network, OLSR uses MPRs to forward traffic within the network. This helps limit network overhead, thus being a more efficient protocol than link state routing protocol [9].

All MPRs declare the link state information for their MPR selectors, thus providing the shortest path to each destination. Consider two nodes, A and B, that don't have recognized links with each other. Initially, node A broadcasts an empty HELLO message. Upon receiving this message, node B finds that its address is not in it. Then node B registers in the routing table that the link A is asymmetric. B broadcasts this message saying that node A is its asymmetric neighbor. By listening to this message, node A sees its own IP address in it and registers node B link as symmetric. Node A then broadcasts a message declaring B as its symmetric neighbor and B register A as symmetric neighbor upon listening to A's message [9].

By using HELLO messages, a node gathers the information of its one-hop and two-hop neightbors. MRPset is calculated using this information, and, similar to the MRPset, an MRPselectors set is maintained at every node. MRPselector set is a set of neighbors that have chosen the node as MRP. Upon receiving a packet, a node checks its MRPselector set to see whether the packet has chosen this node as MRP. If so, then the packet is forwarded. If not, packet is discarded. To maintain routing tables, OLSR imposes huge network overhead. Use of MPRs reduces this control overhead within the network, but in small networks, the gain is minimal.

In our work, we will use DSDV routing protocol for simplicity. Using ad hoc routing protocol, we will gather neighbor information for all nodes and formulate a neighbor table that is helpful for encoding packets in a network.

### 2.3 Related Work

The broadcast nature of wireless networks is generally treated as a curse in regard to most wireless networks. Zhang et al. [2] proposed a physical layer network coding (PNC) scheme, which uses interference as blessing to improve performance in a multi-hop network. In PNC scheme, the relay does not perform coding arithmetic on digital bit streams. Rather, PNC makes use of the additive nature of the concurrently arriving electromagnetic waves by directly mapping the combined signals received concurrently to a signal to be relayed. In a normal three-node linear network (shown in Figure 2.1), the traditional transmission scheme requires four transmissions for exchange of two frames. Network coding requires three transmissions, whereas PNC requires only two transmissions for exchange of two frames. However, PNC scheme assumes symbol-level and carrier-phase synchronization are attained before coding which might not be possible in real-world situations.

Due to the fact that PNC is restricted to small networks, Signal-to-Noise ratio (SNR) increases with increase in nodes. In addition, encoding and decoding procedures are difficult when compared to the proposed network coding of Ahlswede et al. [4]. In traditional store-forward networks, packets are forwarded hop-by-hop along the intermediate nodes from source to destination.  An intermediate node in traditional routing just forwards the packets received to the node on its path. On other hand, network-coding scheme allows an intermediate node to combine all sources before sending the coded data to destinations. Using appropriate network encoding schemes at intermediate nodes, network capacity is boosted drastically. The work of Aslswede et al. [2] reveals that in network coding in general, it is not optimal to regard the information to be multicast as a ‘fluid’ that is simply replicated or forwarded. Rather, coding schemes are employed at nodes to save bandwidth of a network. Network coding scheme can also be applied to different wireless networks, which we will discuss in below paragraphs.

COPE [1] is one of the most practical network-coding schemes in wireless networks. The process of sending two frames using traditional routing takes four transmissions, whereas using network coding it takes three transmissions. The saved transmission is used to send new data which increases wireless throughput. COPE exploits the shared nature of wireless medium in which a node broadcasts its packets in the small neighborhood around its path. Each neighbor node stores the packet overheard for short time [1]. It also tells its neighbor about overheard packets along with its packets. When a node transmits a packet, it uses information gathered from its neighbors, then XORs multiple packets as a single packet [1]. By this scheme, coding opportunities are increased compared to normal network coding. Katti et al. [1] employed a bit-map format in packets reception report to report packets, which have been overheard by the node. Even though this report has two benefits (compactness and effectiveness), their scheme of handling overheard packets has some limitations. Their approach designs code around the norm of never delaying packets, i.e. when the wireless channel is available, the node takes the packet at its output queue, checks to see if other packets in the queue may be encoded with this packet, XOR those packets and transmits out. If no coding opportunity is available then the node does not wait for the arrival of a matching code-able packet [1]. Furthermore, since more packets are overheard, more memory will be required at each node. In addition, high power consumption will be required to transmit packets with high overhead. We proposed a scheme with similar opportunistic listening but we used even-parity network coding which reduces processing power of CPU than COPE. Therefore, we can trade-off processing power with power required to encode or decode all overheard packets.

To save the energy wasted in broadcast storm by simple blind flooding in a network, Shuai et al. [4] proposed a Connected Dominating Set (CDS) - based broadcasting. In this scheme, only the nodes in CDS forward the data to the entire network, which decreases transmissions radically. On other hand, CDS provides a common path for information flow and improves opportunities of intersection of information flow in a network. Dominating nodes (a node that connects to the maximum amount of nodes around it) only forward the received broadcast packets while other nodes just receive them, avoiding redundant transmissions. [4] CDS solves the coding opportunity loss problem by creating more coding opportunities since more information flows are intersected at nodes in CDS. This scheme drastically increases energy gain of a network when compared to normal flooding. However, this scheme is highly dependent on the CDS path. In a given topology if the nodes are unable to form a CDS path then coding opportunity will not occur. In addition, finding a minimum connected dominating set in a network is sometimes a problem. Our proposed method is independent on the established route paths. The only limitation to the proposed scheme is that chain topologies will not work well with respect to creating coding opportunities.

Agha et al. [5] proposed another scheme -- Luby Transform (LT) code -- one of the fountain codes, with a source independent backbone in order to reduce broadcast storms. The proposed backbone concept is very similar to a multipoint relay. The routed path selection is similar to CDS as discussed in the above paragraph. For each message, there exists a selected backbone set of nodes called forwarders that need to retransmit the message to all nodes. This type of backbone set is called a connected dominating set i.e. set K is the dominating set for P if each node from P is either in K or has a neighbor in K. In MPR-CDS, first, nodes with the smallest ID amongst their neighbors are selected. Then the node selects its multipoint relay set (a subset of one-hop neighbors which covers all two-hop neighbors) and attaches the list to its hello message to neighbors. [5] Upon receiving the hello message, the nodes decide to belong to the CDS if it has either smallest ID in its neighborhood or if it is the multipoint relay for the neighbor of the node with the smallest ID. So, when a forwarder has the opportunity to send data, it checks its buffers to check if it has any native or encoded packets, then it randomly mixes the packets and broadcasts them to its neighbors. This scheme uses a simple distribution to choose the degree of encoded packet such as [5] $P(d) = 1/2^d$ ; $0<d\leq k$ where d is the degree of encoded packet and K is the total number of native packets available at the forwarder. The node retrieves the native packets from the received packets by reducing the degree. Suppose that node z has packets s1 and s2 in the buffer, and it receives an encoded packet E=s2+s3+s4 of degree 3. By XORing S2 with E we get E' =E+s2 = s3+s4 which has a degree of 2. If the reduced degree is greater than 1 then the E will be stored in the buffer. Even the singletons (i.e native packets) are stored for further decoding processes. In this scheme, the packet encoding is dependent on the degree of encoding rather than coding opportunities. Therefore, many coding opportunities are missed if the wrong degree of encoding is selected. In our even-parity scheme we are not choosing any routed path or degree of encoding which can improve coding opportunities more than LT code. LT code scheme is highly useful to prevent broadcast storms in ad hoc networks.

# CHAPTER 3

# WORKING AND SYSTEM MODEL

This chapter presents a comprehensive description of the traditional network coding which is the foundation of the entire scheme. Furthermore, this chapter explains the working procedure and system models of even-parity network coding.

## 3.1 Even Parity Scheme Description

Network coding is a scheme to increase the transmission capacity of a network. In traditional store-and-forward networks, intermediate networks simply forward the packets received by its adjacent nodes and packets are forwarded to their destinations along intermediate nodes. On other hand, network coding schemes allow intermediate nodes to combine packets and forward the combined packet to the destination node.

In even-parity network coding as with traditional coding, the relay node not only stores and forwards the data packets but also combines all possible messages received at the relay and forms a coded packet. The relay then broadcasts the coded packet to all nodes within its range. Ad hoc nodes running ad hoc routing protocols maintain topology tables that contain all neighbor information in it. This table helps nodes have up-date information of all nodes, which allows nodes to have routes available on request. The main advantage of this scheme is that even-parity can be performed for more than two different streams of data at a time which is not possible using XOR operation. In addition, we use simple logic for encoding algorithm that increases the possibility of encoding a large number of packets into a single coded packet.

For the decoding process, we are adding an extra identifier field to identify the packets encoded in a coded packet. This field is variable and depends on a number of node scenarios. Let us consider a 16 node scenario in which 4 packets are encoded into a single coded packet. The number of bits required as identifier in each packet is 4 i.e. 2^4 is 16 and the relay node needs to encode four packets into one coded packet, which means 4 packets identifier need to be placed in relay node identifier field. Therefore, it requires four 4-bit fields. This is very small overhead and can be traded off with energy gain. In addition, by using this algorithm, the number of transmissions required is drastically less when compared to un-coded transmissions.

### 3.2 Algorithm for Even-Parity Network Coding

Similar to COPE [1], our scheme also uses opportunistic listening along with even-parity coding scheme. We use opportunistic listening nodes in a network so that we can overhear the packets and coding opportunities and thus make improvements. The packet processing algorithm for even-parity coding is shown in algorithm a. [4] Node p checks whether all its neighbors N (p) received a native packet k based on its neighborhood information (line 1). If yes, relay node tries to see whether it can encode the packet with remaining packets in the queue as per the encoding procedure (line 2). If yes, relay node encode these packets and broadcast the coded packet in one transmission (line 4). If not, the relay will buffer the packet for $T_{threshold}$ time (line 6-7) and then process it. If buffer time exceeds the $T_{threshold}$ time then relay, send out that packet immediately (line 9-10). Finally, if the received packet k is not a native packet then it is decoded (line 14). If a node received packet is, packet not matched with the MAC-address rule then it is stored for decoding purpose (line 15-17).

---

**Algorithm a: Processpkt(k)**

---

1: **if** all neighbors N(p) received Nativepkt(k) then

2: T=Encodepkt();

```
        if T!=1 then
            sendcodedpkt();
        else
            while time(k)<T_threshold do
                Buffer(k,T_threshold);
            end while
            if time(k)≥T_threshold then
                sendNativepkt(k);
            end if
        end if
else
    q=decodepkt(k)
if receivedpkt=sendnativepkt(k)
    store the packet
end if
```

Our main aim is to encode as many packets as possible. In algorithm b, our encoding procedure is explained. The relay node, whenever it has an opportunity to transmit, first picks a packet in the queue and checks with the remaining packets q, satisfies the MAC address matching rule, then the node encodes the passable packets (line 4-7).

**Algorithm b: Encodepkt()**

1: First packet k at the output queue

2: T=1

3: **for** all remaining packets q in the queue **do**

4: **if** q statisfies the *Address Matching* rule then

5: k=Evenparity(k,q)

6: Insert the packet ID's in the identifier field

7: T=T+1

8: **continue**

9: **end if**

10: **end for**

11: return (k, T)

**Address Matching Rule**: With the help of neighborhood information, nodes formulate a send table in which the first packet source address checks whether it is a neighbor of the second packet destination address in queue and vice-versa.

**Algorithm c: Decodepkt(k)**

```
Pick packet k at the head of input queue
T=1
    for all remaining packets r in the input queue do
        if r satisfies identifier matching rule then
            k=Evenparity(k,r)
            T=T+1
        continue
        end if
    end for
return(k,T)
```

**Identifier Matching Rule:** The coded packet identifier should match one of the identifiers of listened packets in the receive table.

### 3.2.1 System Model with Tables

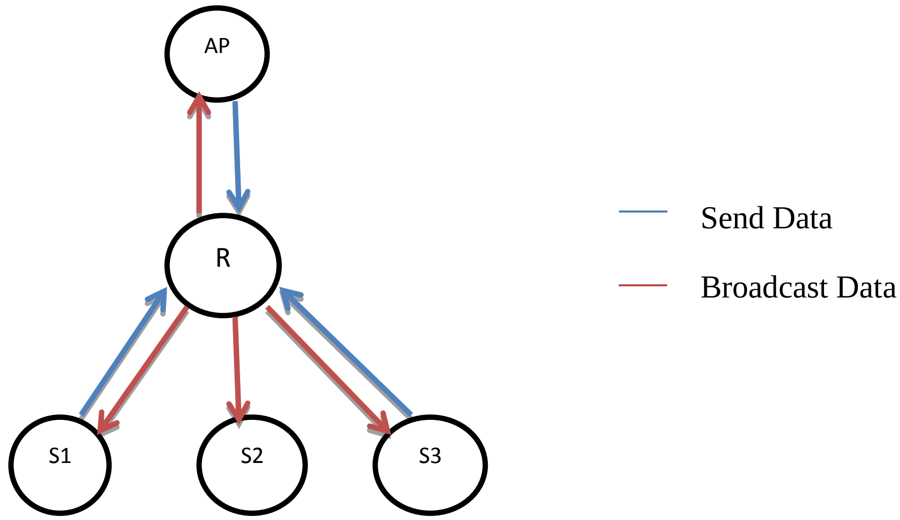



Figure 3.1. System Model

Consider a system model where node S1 and S3 are trying to send data to AP and S2 is trying get data from AP. Below are the tables formulated from neighborhood information exchanged between nodes.

TABLE 1

S1'S NEIGHBOR TABLE, SEND TABLE, AND RECEIVE TABLE

<table>
<tr><td colspan="24">Neighbor Table</td></tr>
<tr><td colspan="12">S1 Neighbors</td><td colspan="12">Neighbor's Neighbor</td></tr>
<tr><td colspan="12">S2</td><td colspan="3">S1</td><td colspan="3">S3</td><td colspan="3">R</td><td colspan="3"></td></tr>
<tr><td colspan="12">S3</td><td colspan="3">S1</td><td colspan="3">S2</td><td colspan="3">R</td><td colspan="3"></td></tr>
<tr><td colspan="12">R</td><td colspan="3">S1</td><td colspan="3">S2</td><td colspan="3">S3</td><td colspan="3">AP</td></tr>
<tr><td colspan="12"></td><td colspan="12"></td></tr>
<tr><td colspan="12">Send Table</td><td colspan="12"></td></tr>
<tr><td colspan="4">Packet Identifier</td><td colspan="4">Time Stamp</td><td colspan="4">Source IP (SRC IP)</td><td colspan="4">Transmitter MAC</td><td colspan="4">Destination IP</td><td colspan="4">Next-hop MAC (NXT MAC)</td></tr>
<tr><td colspan="4">0011</td><td colspan="4">0μs</td><td colspan="4">S1.IP</td><td colspan="4">S1.MAC</td><td colspan="4">AP.IP</td><td colspan="4">R.MAC</td></tr>
<tr><td colspan="12"></td><td colspan="12"></td></tr>
<tr><td colspan="12">Receive Table</td><td colspan="12"></td></tr>
<tr><td colspan="12">Packet Identifier</td><td colspan="12">Payload</td></tr>
<tr><td colspan="12">0101</td><td colspan="12">S3.payload</td></tr>
</table>

TABLE 2

S2'S NEIGHBOR TABLE, SEND TABLE, AND RECEIVE TABLE

| **Neighbor Table** | | | | |
|---|---|---|---|---|
| S2 Neighbors | Neighbor's Neighbor | | | |
| S1 | S2 | S3 | R | |
| S3 | S1 | S2 | R | |
| R | S1 | S2 | S3 | AP |
| | | | | |
| **Receive Table** | | | | |
| Packet Identifier | Payload | | | |
| 0011 | S1.payload | | | |
| 0101 | S3.payload | | | |

S2 initially sending nothing so send table will be empty for now.

TABLE 3

S3'S NEIGHBOR TABLE, SEND TABLE, AND RECEIVE TABLE

| **Neighbor Table** | | | | | |
|---|---|---|---|---|---|
| S3 Neighbors | | | Neighbor's Neighbor | | |
| S1 | | | S2 | S3 | R |
| S2 | | | S1 | S3 | R |
| R | | | S1 | S2 | S3 / AP |
| | | | | | |
| **Send Table** | | | | | |
| Packet Identifier | Time Stamp | Source IP (SRC IP) | Transmitter MAC | Destination IP | Next-hop MAC (NXT MAC) |
| 0101 | 2μs | S3.IP | S3.MAC | AP.IP | R.MAC |
| | | | | | |
| **Receive Table** | | | | | |
| Packet Identifier | | | Payload | | |
| 0011 | | | S1.payload | | |

TABLE 4

R'S NEIGHBOR TABLE, SEND TABLE, AND RECEIVE TABLE

| **Neighbor Table** | | | | | |
|---|---|---|---|---|---|
| R Neighbors | | | Neighbor's Neighbor | | |
| S1 | | | S2 | S3 | R |
| S2 | | | S1 | S3 | R |
| S3 | | | S1 | S2 | R |
| | | | | | |
| **Send Table** | | | | | |
| Packet Identifier | Time Stamp | Source IP (SRC IP) | Transmitter MAC | Destination IP | Next-hop MAC (NXT MAC) |
| 0011 | 1μs | S1.IP | S1.MAC | AP.IP | R.MAC |
| 0101 | 3μs | S3.IP | S3.MAC | AP.IP | R.MAC |
| 0001 | 5μs | AP.IP | AP.MAC | S2.IP | R.MAC |

TABLE 5

AP'S NEIGHBOR TABLE, SEND TABLE, AND RECEIVE TABLE

| **Neighbor Table** | | | | | | |
|---|---|---|---|---|---|---|
| AP Neighbors | | | Neighbor's Neighbor | | | |
| R | | | S1 | S2 | S3 | R |
| | | | | | | |
| **Send Table** | | | | | | |
| Packet Identifier | Time Stamp | Source IP (SRC IP) | Transmitter MAC | Destination IP | Next-hop MAC (NXT MAC) | |
| 0000 | 4μs | AP.MAC | AP.MAC | S2.IP | R.MAC | |

Initially**,** the access-point will not hear any packets from its neighbors, so the receive table is empty.

# CHAPTER 4

# PERFORMANCE EVALUATION AND RESULTS

This chapter presents a performance evaluation of network coding with even-parity scheme. Furthermore, this chapter presents an example scenario, which explains the working procedure of even-parity network coding.

## 4.1 Working Procedure of Algorithm with a Scenario

In this section, consider a real scenario where network coding is more important than traditional routing. Nodes A, B, C are FTP clients uploading or downloading files from an FTP server. Let’s say A=10001010…, B=10110101…., C=11010111…, FTP server=11001011…

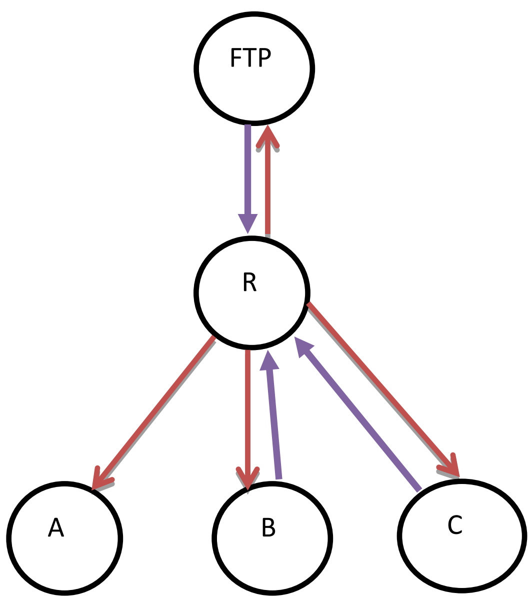



Figure 4.1: Network coding scenario

(Red line = broad cast and blue lines = unicast)

In Figure 4.1, client B and C are trying to upload some data to the FTP server and client A wants to download data from the server at same time.

TABLE 6

SEND TABLE AT RELAY (TTHRESHOLD =10 µS)

| Packet Identifier | Time Stamp | Source IP (SRC IP) | Transmitter MAC | Destination IP | Next-hop MAC(NXT MAC) |
|---|---|---|---|---|---|
| 0010 | 2µs | B.IP | B.MAC | FTP.IP | R.MAC |
| 0100 | 4 µs | FTP.IP | FTP.MAC | A.IP | R.MAC |
| 0011 | 6 µs | C.IP | C.MAC | FTP.IP | R.MAC |

After $T_{threshold}$ time, according to the algorithm, the first packet of the send table will compare its source and destination IP address with its preceding packet destination and source IP; both packets should be neighbors to match the condition. If any packet does not match this condition it will be sent out. After completing a comparison of all packets, the node will perform even-parity with all matching packets one-by-one. The resultant packet will be broadcasted out.

Consider our scenario above; the first packet source IP is neighbors with second packet and vice-versa. The first packet will compare the same condition with a third packet. This time B.IP is not neighbor of FTP.IP and vice-versa, so it will send out the third packet to its next hop address, i.e. R (relay), and then it will send it to destination node FTP. Since comparison has been completed, it will perform even-parity of the packets of data and combine packet identifiers in an id field.

B and FTP packet XOR:

B=10110101

FTP =11001011

-----------------------

XOR=01111110

TABLE 7

CODED PACKET

| Packet Identifier | | Source MAC | Destination MAC | data |
|---|---|---|---|---|
| 0010 | 0100 | R.MAC | FFFF | 01111110 |

After 20µs, in order to receive the table algorithm, the first packet identifier will be compared with the second packet identifier in the table. If it matches, it will perform even-parity with those two packets and store the packet. This process will continue until all packet identifiers finish comparison with each other.

TABLE 8

NODE A RECEIVE TABLE

| Packet Identifier | Payload(data) |
|---|---|
| 0010 | 10110101 |
| 0011 | 11010111 |
| 0010 0100 | 01111110 |

At node A, the first packet identifier is matched with relay broadcasted value, so it will perform even-parity with those two data packets. i.e

B= 10110101

Broadcasted data=01111110

--------------------------

11001011 →Data downloaded for A from FTP server

TABLE 9

NODE B RECEIVER TABLE

| Packet Identifier | Payload(data) |
|---|---|
| 00000011 | 11010111 |
| 00000010 00000100 | 01111110 |

At node B, the first packet identifier is not matched with its preceding packets in table, so the node just stores the broadcasted packet.

TABLE 10

NODE C RECEIVER TABLE

| Packet Identifier | Payload(data) |
|---|---|
| 00000010 | 10110101 |
| 00000010 00000100 | 01111110 |

At node c, the first packet identifier is matched with its preceding packet in the table, so the node performs even-parity with the two matched packets.

B=10110101

Broadcast value=01111110

------------------

11001011

Store the resultant packet from above operation (even-parity).

TABLE 11

FTP SERVER RECEIVER TABLE

| Packet Identifier | Payload(data) |
|---|---|
| 00000011 | 11010111 |
| 00000010 00000100 | 01111110 |

In this case, the receive table needs to have its own packet; only then can the information for this node be extracted.

## 4.2 Simulated Values and Their Respective Graphs

We simulated the scenarios in MATLAB to compare relative energy gain by varying the number of nodes. Our choice of scenarios for the performance evaluation studies is based on the fact that they will offer better network coding opportunities when compared to general topologies. Energy gain [4] is defined as a total number of transmissions without coding to total number of transmissions with coding. This metric is used to prove our algorithm works better than traditional network coding. We considered different scenarios and collected a number of transmissions with and without coding. Here is the table of values.

TABLE 12

MATLAB SIMULATED VALUES FOR PLOTS

| **Number of Nodes** | **Number of transmissions without coding** | **Number of transmissions with COPE** | **Number of transmissions proposed coding** | **Energy gain with proposed coding** |
|---|---|---|---|---|
| 5 | 6 | 5 | 5 | 1.2 |
| 10 | 16 | 15 | 12 | 1.33 |
| 15 | 26 | 21 | 19 | 1.37 |
| 20 | 32 | 28 | 22 | 1.45 |

In table 12, we can see that, as the number of nodes increases, the coding opportunities increase as per our algorithm. For smaller node scenarios, the difference in transmissions with and without coding are much less, whereas the difference is noticeable for a 20 node scenario. As the number of nodes increases, the energy gain increases with it. These values may differ by changing the transmission range of nodes and in worst case number of transmissions with coding is equal to number of transmissions without coding. So we are not affecting performance of the network by using energy-efficient even-parity network coding. From the above table we can see as number of nodes increases the difference between number of transmissions with COPE and proposed coding is increasing. A graph has been plotted using the above values. Figure 4.2 represents the energy gain of nodes in each scenario.

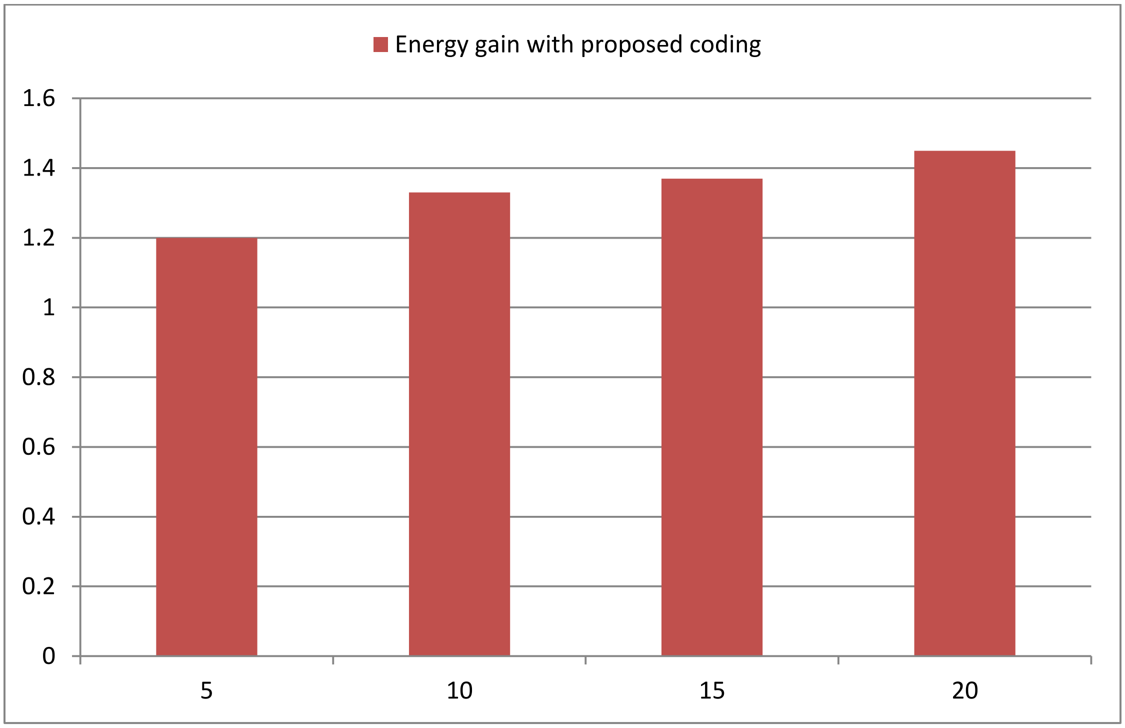



Figure 4.2. Comparison of Number of Nodes Scenario to Energy Gain in Each Scenario

The x-axis of the above graph represents the number of nodes in a theoretical values table while the Y-axis represents energy gain in each scenario. From the above graph we can observe that the energy gain increases with an increase in the number of nodes.

By using even-parity network coding, opportunities are increased over normal blind flooding. This increase in coding opportunities leads to a decrease in the number of transmissions. A graph is plotted using these transmissions to number of transmissions without coding. Figure 4.3 represents a decrease in transmissions using even-parity scheme versus traditional flooding.

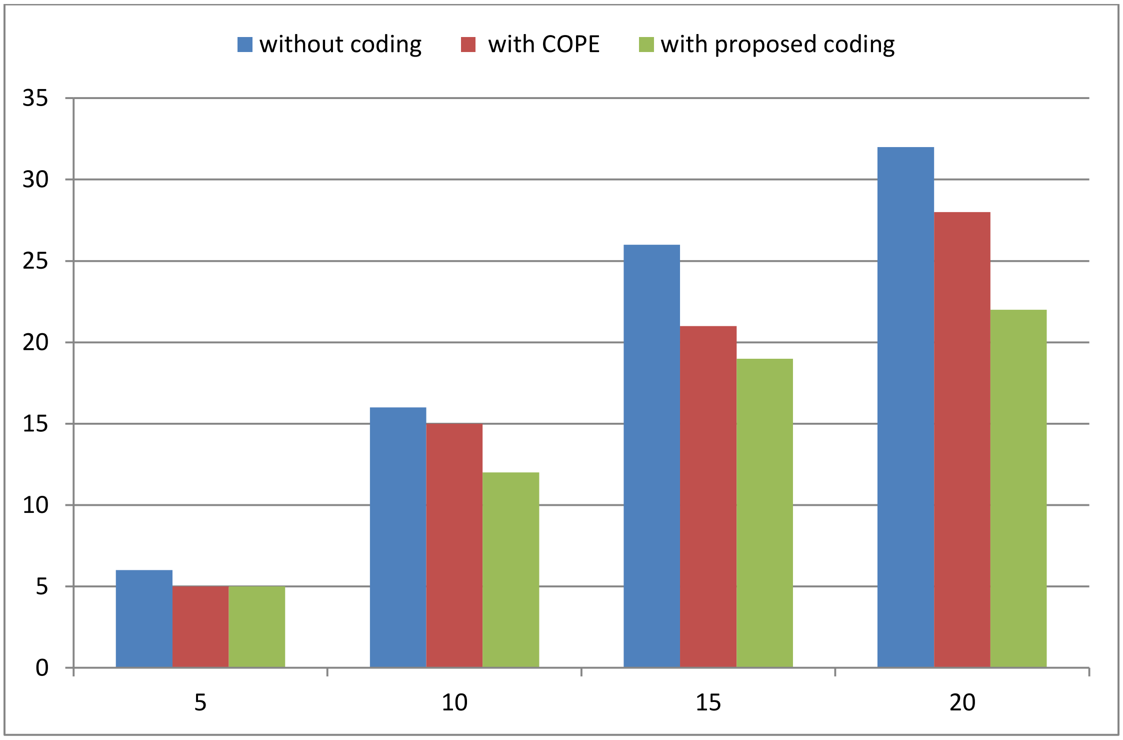



Figure 4.3. Comparison of Number of Nodes Scenario to Number of Transmissions data without coding, with COPE and with proposed coding

The x-axis of the above graph represents the number of nodes used for calculations and the y-axis represents the number of transmissions required to transmit data without coding, with COPE and with proposed coding.

# CHAPTER 5

# CONCLUSIONS

In this thesis, a scheme called even-parity network coding at relay nodes for relay network was presented. The main advantage of this scheme over traditional network coding is that even-parity can be performed for more than two different streams of data. Even-parity coding reduces processing time of nodes, thereby improving coding opportunities. In addition, we use a simple procedure for encoding and decoding packets that in turn decreases coding complexity.

We used Matlab for simulating different scenarios. It was observed that an increase in the number of nodes increases coding opportunities which in turn decreases the number of transmissions. Energy gain is defined as the total number of transmissions without coding to the total number of transmissions with coding. This metric is used to prove our algorithm works better than traditional network coding. On an average using our scheme, energy gain is increased by 1.34.

# REFERENCES

# REFERENCES


1. S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, “Xors in the air: practical wireless network coding,” in SIGCOMM, 2006.

2. S. Zhang, S. Liew, and P. Lam, “Physical Layer Network Coding,” CMMobicom, 2006.

3. R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, “Network information flow,” IEEE Transactions on Information Theory, vol. 46, no. 4, pp. 1204-1216, 2000.

4. W. Shuai, A. Vasilakos, J. Hongbo, M. Xiaoqiang, L. Wenyu, P. Kai, L.Bo, D. Yan, “Energy Efficient Broadcasting Using Network Coding Aware Protocol in Wireless Ad hoc Network,” Communications (ICC), 2011 IEEE International Conference on, Kyoto.

5. K. A. Agha, N. Kadi, I, Stojmenovic, “Proceedings of the 69$^{th}$ Annual IEEE VTC conference on Vehicular Technology, April 2009.

6. S. Sengupta, S. Rayanchu, and S. Banerjee, “Network Coding-Aware Routing in Wireless Networks,” IEEE/ACM Transactions on Networking, vol. 18, no. 4, pp. 1158-1170, 2010.

7. L. Li, R. Ramjee, M. Buddhikot, S. Miller, “Network Coding-Based Broadcast in Mobile Ad hoc Networks,” in INFOCOM, 2007.

8. C. Fragouli, D. Katabi, S. Katti, A. Markopoulou, M. Medard, and H. Rahul. Wireless network coding: Opportunities and challanges. *Proceedings of Military Communications Conference*, 2007.

9. P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, L. Viennot, “Optimized Link State Routing Protocol for Ad Hoc Networks,” Proceedings of INMIC Annual IEEE conference, 2001.