8-2016

A Protocol to Recover the Unused Time Slot Assignments in Transmission Scheduling Protocols for Channel Access in Ad Hoc Networks

Vikas Bollapragada Subrahmanya
Clemson University, vbollap@clemson.edu

A PROTOCOL TO RECOVER THE UNUSED TIME SLOT ASSIGNMENTS IN TRANSMISSION SCHEDULING PROTOCOLS FOR CHANNEL ACCESS IN AD HOC NETWORKS

A Thesis
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
Electrical Engineering

---

by
Vikas Bollapragada Subrahmanya
August 2016

Accepted by:
Dr. Harlan B. Russell, Committee Chair
Dr. Daniel L. Noneaker
Dr. Kuang-Ching Wang

ABSTRACT

In mobile ad hoc networks without centralized control distributed transmission scheduling protocols for channel access are of interest. Many scheduling-based MAC protocols have been proposed to provide contention-free transmissions and to guarantee certain levels of performance. However, one of the major drawbacks of these protocols is that once a slot is assigned to a particular node if the node does not have a packet to transmit, then the slot is not utilized. This leads to a poor network performance. In our proposed protocol these assigned but un-utilized slots are recovered by other nodes.

We use custom computer simulations to compare our new protocol against two approaches that do not recover wasted slots. The simulation models the performance of the physical and link layers and includes a limited network layer that supports end-to-end forwarding of traffic. Through investigations of random networks with varying densities we conclude that our new approach results in an increase in the capacity of the network.

DEDICATION

To my parents, Dr. Ramana and Dr. Sreenivas for their love and support. To my brother, Varun, who has been a mentor and an inspiration.

ACKNOWLEDGEMENTS

I want to start by expressing my deepest gratitude to my advisor, Dr. Harlan Russell, for his regular brainstorming sessions, great advice, and quick feedback that has helped me immensely while conducting the research in this thesis. I would like to thank the faculty in the wireless group who have helped me gain the necessary background for performing this research. I would also like to thank Dr. Noneaker and Dr. Wang for serving on my committee. Finally, I want to thank Siddhartha Borkotoky, for the many discussions and for his help in getting my code running on the palmetto cluster.

TABLE OF CONTENTS

Table of Contents - Continued

LIST OF FIGURES

Page

List of Figures - Continued

Page

LIST OF TABLES

Chapter 1

INTRODUCTION

Mobile ad hoc networks (MANETs) are a key enabling service to support emerging networking paradigms such as vehicular networks, internet of things, disruption tolerant and opportunistic networks, intra-body networks, and under water networks. MANETs are a special type of wireless networks where the nodes exchange packet data without depending on a pre-planned infrastructure such as base stations or access points. Depending on the situation, every node that makes up the network plays one of the following three roles: a source, a destination, or as a router to relay packets from the other nodes. Another important feature is that the nodes can be mobile. These networks can be setup on demand, anywhere, and at anytime. In the past MANETs have primarily found application for military operations or during disaster management if the existing infrastructure is no longer operational.

Today almost every electronic device we use is equipped with some kind of wireless transceiver. All the smart devices are able to connect to the internet. This opens up endless possibilities for MANETs in our day to day lives. The authors in [1] describe the concept of ad hoc networking. They provide the historical background and overview some of the challenges facing future use of MANETs.

Forming an ad hoc network without depending on a centralized infrastructure requires that all of the nodes that participate in the network must be able to self-organize and self-configure. In particular, the link- and network-layer protocols must be designed to be distributed and robust. To limit the distribution of control information, decisions made by these protocols should be as localized as possible and not require excessive flooding of control information throughout the network. Furthermore, fairness is also a key requirement in the design of protocols that control channel-access and provide relay capacity for an ad hoc network.

Mobile networks require that the protocols for an ad hoc network are able to adapt to changes in topology efficiently. Even in networks that are envisioned to be static or slowly moving, there still is a requirement to adapt to dynamic changes in the network connectivity. For example, nodes may join or leave for various reasons, external sources of interference may arise at any time or be associated with mobile devices, shadowing or slow fading can occur depending on the deployment location, or in the case that cognitive radios are able to exploit white spaces there may be requirements to change RF assignments based on the availability of spectrum.

Many of the applications deployed on an ad hoc network will require that the nodes support relaying of traffic through multiple hops. To provide sufficient coverage with radios that inherently have limited communication range, nodes will need to discover their neighbors and building forwarding information so that packets can be routed to their required destinations or gateways. Many prior investigations have shown that cross-layer protocols are necessary to achieve good network performance [2, 3, 4, 5]. It is not sufficient to assume that links are simply good or bad, but the networking protocols need to account for the quality and capacity of the available links. Likewise, channel-access or transmission scheduling protocols can benefit from information at the network layer concerning demands for link capacity when establishing channel-access opportunities. Thus, the protocols must use cross-layer information to be able to robustly adapt to dynamic network conditions and demands. In including cross-layer information, care must be taken to ensure that the protocols make efficient utilization of the wireless resources and limit the overhead due to control information.

1.1 Medium Access Control (MAC)

The operation of the MAC protocol is critical to achieve good network performance in an ad hoc network, and the efficiency of shared access to the channel is the focus of this thesis. The MAC protocol controls the opportunities for the nodes in the network to transmit, and a key aspect is to coordinate the transmissions to ensure a high probability of reception and efficient use of the channel. When more than one node transmits simultaneously interference is created at the receivers. At a specific receiver, the signal that is received is a mixture of the signal from the intended transmitter plus the signals from the other transmitters. The presence of the signals from the transmitters other than the intended transmitter is called multiple-access interference. Depending upon the signal quality from the intended transmitter at the target receiver, some level of interference can be tolerated. The level of tolerance depends on the underlying physical-layer features such as the modulation scheme and type of error control coding.

A collision is said to occur when the interference at the receiver inhibits it from receiving the transmission that was intended for it. A good MAC protocol takes advantage of the physical-layer characteristics and channel properties and maximizes the concurrent transmissions while minimizing the packet loss due to collisions. The MAC protocol plays a much more significant role in a wireless network because the wireless links are inherently error prone and are less reliable when compared to wired links. For example, properties of a wireless channel that can lead to poor performance for the links include unique issues like channel fading, shadowing, mobility, the hidden terminal problem, the exposed terminal problem, and sources of interference from transmitters that are not participating in the MAC protocol. Wireless transmissions require more coordination than the wired counterparts and the resulting channel-access protocols are more complicated.

There are two classes of MAC protocols for wireless networks: contention based and schedule based. A survey of MAC protocols specifically for ad hoc networks can be found in [6]. Next, we briefly overview both protocol types.

1.1.1 Contention Based MAC Protocols

In contention-based MAC protocols the nodes contend with each other for each opportunity to transmit. In most of these protocols, a node attempts to access the channel as soon as possible after a packet is available. Frequently there are collisions and re-transmissions are required. A random delay is introduced before a retransmission to reduce the chances that there is another collision or that the intended destination for the transmission is not in receive mode (e.g., the intended node is itself transmitting, locked to a different transmission, or blocked). The frequency of attempts to access the channel and the success of the transmissions depend on the traffic load because the nodes initiate an access attempt whenever a packet is queued. Typically, the performance is good when the traffic load is low but it degrades rapidly when the traffic load is high. Unsuccessful transmissions and long delays occur when multiple nodes contend for access to the channel at the same time.

Some of the earliest examples of contention-based protocols are aloha [7], slotted aloha [8], and similar variations. These protocols are characterized by CSMA, and the nodes do not exchange any control information before transmitting a packet. An extensive discussion of fundamental stability concerns with protocols based on aloha is found in [9]. Many contention-based protocols employ additional signaling to address the limitations of

CSMA. One common approach is to use a collision avoidance (CA) strategy, and examples include MACA [10], MACAW [11], and the DCF mode found in the IEEE 802.11 standard [12]. While CSMA/CA approaches address the stability concerns of basic CSMA, they still under perform when traffic loads in the network are high [13]. One other major disadvantage of contention-based protocols is that they can be unfair to the nodes which have less traffic than the nodes with more traffic.

1.1.2 Contention Free MAC Protocols

In transmission scheduling MAC protocols the nodes do not contend with each other to access the channel but instead reserve the channel. Various strategies are possible including scheduling based on time in TDMA, frequency in FDMA, and code in CDMA. We focus on TDMA-based MAC protocols in this work. In traditional TDMA approaches the frame size is set equal to the number of nodes in the network and each node is assigned one time slot in the frame. Spatial TDMA (STDMA) is an improvement over TDMA in which nodes far away from each other can re-use the channel. We refer to STDMA protocols as schedule-based MAC protocols in this manuscript.

In schedule-based MAC protocols the nodes are scheduled to transmit in particular slots in a frame. Schedules are formed so that nodes transmitting simultaneously are separated from each other by enough distance to keep multiple-access interference at acceptable levels. Typically each node is assigned at the least one slot in the frame. Consequently, these protocols are collision free and fair. Schedule-based MAC protocols are also of interest in ad hoc networks as they can guarantee a certain level of quality of service (QOS).

However, schedule-based protocols often perform poorly when compared to contention-based MAC protocols at low traffic levels. The primary reason is because slots that are assigned to nodes that do not have a packet to transmit are not utilized. This increases the delay significantly as compared to the contention-based protocols, where the nodes attempt to access the channel whenever they have a packet to transmit. On the other hand, at high network traffic levels these protocols perform much better as the transmissions are scheduled to avoid contention and the need for randomized delays before retransmission attempts.

1.2 Thesis Statement

The focus of this thesis is on large ad hoc networks with a dense topology so that nodes typically have ten or more neighbors and packets often require multiple relays to reach their destinations. The network is expected to support periods with high levels of traffic, and the ability to efficiently utilize the network with some QOS support is desirable. The network is assumed to be static or have limited mobility during periods when it is heavily utilized. We assume a typical application of the network is to provide backbone connectivity service to various wireless applications. We envision that future generation routing and transport protocols will be designed to take advantage of cross-layer information about link performance. Support for hierarchical network organization, congestion control, capacity planning, and other features can be more easily supported if the channel-access protocol can provide reliable information about the available links. In this environment, scheduled channel access is preferred so that high levels of traffic can be efficiently supported.

One of the major drawbacks of a schedule-based MAC protocol is un-utilized time slots. Each node is assigned one or possibly more time slots in a frame. If a node does not have a packet to transmit during its slot in a frame, then that slot is wasted. This limits the capacity of the network and increases the delay. In this thesis we design and investigate a scheme that recovers these un-utilized slots while maintaining the advantages of a schedule-based MAC protocol.

We propose a new distributed scheduling protocol that recovers the un-utilized slots. Our protocol is described as an extension to Lyui's protocol (originally defined in [14]), however, our extension can be utilized with any schedule-based protocol. The key idea for our new protocol is to allow neighboring nodes to detect when a time slot is not utilized and to permit a different node with a packet to transmit during that time slot. The critical condition is to ensure that the substitute transmission creates multiple-access interference that is similar to the interference that would have been generated by the original node scheduled to transmit in this slot. This ensures that the transmission schedules of other nodes are still viable, and the interference environment for all transmissions in this time slot are still within acceptable levels.

A custom computer simulation is used to investigate the performance of our new channel-access protocol. The simulation models the performance of the physical and link layers, and includes a limited network layer that supports end-to-end forwarding of traffic. The simulation models direct sequence spread spectrum (DSSS) modulation. An advantage of DSSS is its ability to communicate in an environment with significant multiple-access interference. A transmission between two nodes is successful only if the signal to interference plus noise ratio (SINR) at the receiving node is greater than a threshold. We compare the performance of our channel-access protocol against Lyui's algorithm [14] and Seedex [15]. For simplicity, the positions of the nodes are static, and forwarding tables are calculated with a centralized routing algorithm. Results are reported for networks with various numbers of nodes and random topologies with various levels of neighbor density. Using extensive simulations we show that our protocol results in a considerable gain in the network performance.

The rest of the document is as follows. Related work is presented in Chapter 2 and a description of the system design is given in Chapter 3. Lyui's protocol is described in Chapter 4 and we develop our new protocol as an extension of Lyui's protocol in Chapter 5.

Chapter 6 contains a detailed description of the other protocols against which we compare against and the modeling and parameter assumptions for the simulation implementation. The results are discussed in Chapter 7, and conclusions and future work are presented in Chapter 8.

Chapter 2

RELATED WORK

There has been considerable research on MAC protocols for ad hoc networks. An extensive survey of some recent protocols can be found in [6], though the majority of approaches covered in this survey focus on contention-based protocols. Contention-free approaches have received less consideration due to the increased complexity in forming reliable and efficient transmission schedules. In this chapter we describe related work that focuses specifically on channel-access approaches that schedule transmission opportunities. Many of these protocols have been shown to provide improved network capacity compared to common contention-based protocols.

One of the earliest investigations to consider scheduling broadcast transmissions is [16]. The authors show that finding the smallest frame size that supports a given traffic rate is NP complete. Both a centralized and distributed protocol is developed to provide an approximate solution. However, the approach requires global coordination among all nodes to agree on a fixed frame size and the resulting schedule cannot be easily updated.

Use of color numbers to establish slot assignments has received considerable interest, and [17] surveys the basic approach and describes a centralized algorithm, RAND, based on a random assignment of color numbers. An extension, called DRAND [18], is a distributed version of RAND. In DRAND nodes use a randomized contention based approach to form the schedules and do not need slot synchronizations during the schedule forming phase. In [18] it is shown that the performance of DRAND is equivalent to RAND. Lyui's algorithm uses a similar approach to assigning color numbers but employs a different algorithm to map color numbers to slot assignments. Investigations in [19] show that the performance of a protocol similar to RAND and Lyui's algorithm are similar, though Lyui's algorithm offers some advantages in supporting mobility and maximizing slot assignments.

The approaches described above assign one slot to each node in a frame regardless of the traffic level at the node. The investigations described next allow a node to request additional slots within a frame based on a long-term projection of the level of traffic it must support. The USAP [20] is one such distributed slot assignment algorithm where the nodes can choose slots from a pool of unassigned slots depending on their traffic requirements. The protocol has a mechanism to coordinate with nodes up to 2-hops away to detect and resolve any conflicts. In [21] a delay efficient TDMA based schedule is proposed. Here instead of assigning the slots to a node, the slots are instead assigned to a flow. A single node can be assigned multiple slots in a frame if it is involved in multiple flows.

Another general approach to scheduling transmissions is to use a contention phase to establish assignments for a short period of time and periodically recalculate the schedules. The five phase reservation protocol (FPRP) [22] is an example of this type of protocol where the nodes contend with each other to reserve slots. This protocol describes a five-phase dialogue between the nodes that are two hops away from each other. A recent extension [23] to the FPRP reduces the frame length.

Finally, SEEDEX [15] adopts a different approach in that the schedules are generated pseudo-randomly and the schedules are not periodic (i.e., there is no frame). Each node generates a pseudo-random number between zero and one with a uniform distribution in each slot. This number acts as the probability with which the node is a candidate to transmit in that slot. A node has the seeds of the pseudo-random generators of each node within two hops from itself. Hence, each node knows which of its two-hop neighbors are candidates in each of the slots. A candidate will transmit in a slot with a probability that depends on the number of possible candidates in this slot. A common problem with all of the transmission scheduling schemes described in this chapter is that once a slot has been assigned to a node, it is not utilized if the node does not have a packet to transmit in that particular slot.

In this thesis we present a transmission scheduling algorithm for multiple-hop ad hoc networks in which each slot is extended by a fixed and small number of mini-slots. From a set of nodes with the same size as the number of mini-slots, one of them can make a substitute transmission if the slot would otherwise be un-utilized by the node assigned to the slot. The nodes in the set form a clique. Work that is most closely related to ours is described in [24] and [25]. In [24] mini-slots are also investigated, however, only for a network in which every node is within line of sight of each other. In a network with N nodes, $N\text{-}1$ mini-slots are added to the beginning of each slot. Nodes are assigned priorities with algorithms such as round robin, random order, head of the line, and alternating priorities. At the beginning of each slot the node with the highest priority transmits an unmodulated carrier (or busy tone) for the duration of a mini-slot indicating that it has a packet to transmit. If the channel is detected to be idle in that mini-slot then the node with the next highest priority transmits the carrier. This procedure is repeated until one of the nodes transmits the carrier and reserves that slot. At the end of mini-slot $N\text{-}1$ the node that transmitted the carrier transmits its packet. A slot is idle only if none of the nodes have a packet to transmit. This approach works well when the number of nodes in the network is small but incurs significant overhead for large networks. Also, the approach does not consider multiple-hop networks, the hidden-terminal problem, or multiple-access interference, and the algorithm for assigning priorities do not generalize to multiple hop networks.

A protocol to permit more than one node the opportunity to utilize a slot is described in

[25]. The approach, called CAMA, is an extension of USAP [20]. The network is divided into multiple cliques, and slots are assigned to the cliques instead of individual nodes using USAP. In the slot assigned to a particular clique the nodes constituting that clique contend for the slot via mini-slots at the beginning using a non-persistent CSMA protocol [26]. However, the authors of [25] point out that the implementation is quite complex, requires careful tuning based on connectivity details, and has higher overhead than required by USAP.

Chapter 3

SYSTEM DESIGN

In this chapter we describe the modeling assumptions for the system we use throughout this work.

3.1 Channel Model

We assume that the communication between the nodes is half-duplex, that is, the nodes cannot transmit and receive at the same time. All nodes are equipped with omni directional antennae that have equal gain in all directions. A standard channel model is employed that is similar to one we previously used in [27]. In this work we assume that each node uses direct-sequence spread-spectrum (DSSS) modulation. We assume that the nodes are synchronized to the slot boundaries. See for example the discussions in [22] and [28] for methods to achieve slot synchronization. A packet is considered to be decoded correctly only if the SINR at the receiving node is greater than a threshold β. Specifically, if node x is transmitting a packet to node y, then the packet can be decoded correctly only if the SINR at the receiving node, denoted by $\xi_{x,y}$, satisfies

$$\xi_{x,y} = \frac{P_r(x,y)N_sT_c}{N_o + \sum\limits_{\forall z \neq x} P_r(z,y)T_c} \geq \beta. \tag{3.1}$$

where $P_r(x,y)$ is the power received at node y from node x, N_s is the spreading factor, T_c is the chip duration, and N_o is the noise at the receiver. The multiple access interference at the node y from nodes other than x is $\sum\limits_{\forall z \neq x} P_r(z,y)T_c$. The signal to noise ratio *SNR*, denoted by $\varepsilon_{i,j}$, is the *SINR* without multiple access interference.

3.2 Modeling Capture and Path Loss

Even though we assume that all transmissions are slot synchronous, because of the limitations of the hardware and propagation delays the transmissions could be slightly asynchronous. When there are multiple signals at a receiver that all satisfy equation (3.1), the receiving node will capture the transmission that reaches it the first. We model this capture in the following way. Each node will form a capture list, which has all the transmissions that have satisfied equation (3.1) in that slot. Then it will pick one of these transmissions at random.

We use a urban area cellular radio path loss model described in [29]. If the distance between the transmitter and receiver is d, then the received power, denoted by $P_r(d)$, is given by

$$P_r(d) = P_t \times (\frac{\lambda}{4\pi d})^{\alpha} \tag{3.2}$$

where P_t is the transmit power of a node, λ is the wavelength of the transmitted signal, and α is the path loss exponent.

We assume that all nodes transmit at the same power. We set the transmit power (P_t) such that if the distance between the transmitter and receiver is equal to R then the SNR is equal to β. The transmit power is given by

$$P_t = \frac{N_0 \beta}{N_s T_c} \left(\frac{4\pi R}{\lambda}\right)^{\alpha} \tag{3.3}$$

We call R the transmission range.

3.3 Network Layer Models

Let $C_{i,j}$ denote the cost of the link between nodes i and j, and $C_{i,j}$ is a function of the SNR of this link. Links with a better SNR have low cost as they have a higher probability of a successful packet transmission. However, if the SNR is sufficiently large, the cost is lower bounded by one. We assume each node maintains an estimate of the SNR to each of its neighbors, and Dijkstra's algorithm is used to find minimum cost routes. For $C_{i,j}$ we employ a similar function as used in [27], and it is given by

$$C_{i,j}(\varepsilon_{i,j}) = \begin{cases} \infty, & \varepsilon_{i,j} < \beta \\ 1 - log(\frac{\varepsilon_{i,j} - \beta}{\beta}), & \beta \le \varepsilon_{i,j} < 2\beta \\ 1, & \varepsilon_{i,j} \ge 2\beta \end{cases} \tag{3.4}$$

Chapter 4

LYUI'S PROTOCOL

Our protocol is built on an algorithm originally defined by Lyui [14] for assigning transmission schedules in a distributed manner in MANETs. A summary of Lyui's algorithm along with a discussion of its properties is given in [19]. The *1-neighborhood* of a node is defined as the node and all of its neighbors, where neighbors are the nodes with which communication is possible. The *2-neighborhood* of a node is defined as the node itself, the collection of neighbors of the node and neighbors of the neighbors. Each node chooses the smallest possible positive number as it color, making sure that no node in its 2-neighborhood has the same color. There are a number of distributed algorithms to assign color numbers [17]. Node u maintains a neighbor list, $\mathcal{N}_u$, which contains the identities of the nodes in its 2-neighborhood and their colors. This is the only information needed by the node to be able to determine whether it is a candidate to transmit in a slot. Details of how initial schedules are formed are given in [30] and a protocol to update color numbers in a mobile network is described in [31].

A node u with color number c_u is a candidate to transmit in slot t if there exists an integer n such that

$$t = c_u + n \times P(c_u) \tag{4.1}$$

where $P(c_u) = 2^k$ and k is the smallest integer such that $2^k \geq c_u$. In each slot t, node u uses (4.1) and its neighbor list, $\mathcal{N}_u$, to form a candidate list $\mathcal{C}_u(t)$ defined as,

$$\mathcal{C}_u(t) = \{x | x \in \mathcal{N}_u \textit{ and x is a candidate in slot t}\} \tag{4.2}$$

Node u will only be assigned to transmit in slot t if $u \in \mathcal{C}_u(t)$ and has the largest color among the other candidates. In this way Lyui's algorithm ensures that a node scheduled to transmit in a slot will be the only transmitter among the nodes in its 2-neighborhood. For node u the frame size is equal to $P(c_{max})$, where c_{max} is the maximum color in the 2-neighborhood of the node u.

Table 4.1 depicts the transmission slots for the first eight colors as assigned by Lyui's algorithm. An X in the table indicates that a node with that color number will be a candidate in that slot. As the table indicates there are multiple candidates in a slot. The node with highest color among these candidates gets to transmit in that slot. For example, consider a scenario where the 2-neighborhood of a node has a total of 5 nodes. Table 4.2 depicts a typical frame for each node in this 2-neighborhood. For simplicity we identify these nodes by their color numbers. Since there are five nodes in the 2-neighborhood, there are five distinct colors numbered from one to five. Also all the nodes belonging to a particular 2-neighborhood will have the same candidate list ($\mathcal{C}_u(t)$). The size of the frame is dependent on the 2-neighborhood of the node. In this example scenario, the frame size for the nodes in that 2-neighborhood is eight as 2^3 is the smallest power of two greater than equal to five.

Table 4.1: Transmission Slots In Lyui's Algorithm For First Eight Colors

	Transmission Slot Number (t)															
Color Number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
2		x		x		x		x		x		x		x		x
3			x				x				x				x	
4				x				x				x				x
5					x								x			
6						x								x		
7							x								x	
8								x								x

If a node is a candidate in a particular slot and it has the highest color amongst the candidate nodes from the candidate list it will transmit in that slot. From Table 4.1 we can see that nodes 1-5 transmit in the first five slots of the frame and nodes 2, 3, and 4 transmit again in the remaining three slots of the frame. In this manner Lyui's algorithm guarantees that each node has the oppurtunity to transmit at the least once in a frame. The nodes with smaller 2-neighborhoods often have a smaller frame size than nodes with larger 2-neighborhoods. This ensures a better spatial re-use than a traditional TDMA scheme. Advantages of Lyui's protocol are discussed in [19]. However, a significant disadvantage of transmission scheduling approaches like Lyui's algorithm is that if a node is scheduled to transmit in a slot but it does not have a packet, the slot is not utilized.

Table 4.2: An Example Frame in Lyui's Algortihm

Transmission slot t	Candidate list $\mathcal{C}_u(t)$	Transmitting node Max($\mathcal{C}_u(t)$)
1	{1}	1
2	{1,2}	2
3	{1,3}	3
4	{1,2,4}	4
5	{1,3,5}	5
6	{1,2}	2
7	{1,3}	3
8	{1,2,4}	4

Chapter 5

RECOVERING MINI-SLOT TRANSMISSION SCHEDULING (RMTS)

In our proposed protocol we recover the time slots that are un-utilized when a node scheduled to transmit does not. For our investigation each node uses Lyui's algorithm to find the slots in which it is a candidate to transmit. The transmissions arising because of these pre-assigned slots are called *primary transmissions* and the nodes are called *primary nodes* in that slot. In addition a node x nominates N_a of its proximate 1-neighbors as a set of *auxiliary nodes* called an *auxiliary set* and denoted by ς_x. One of the nodes in ς_x will try to transmit if the primary node x does not. This results in the second type of transmissions called *auxiliary transmissions*. Each node has its own auxiliary set.

Not every 1-neighbor of a node can be its auxiliary node. A node belonging to an auxiliary set of a primary node must satisfy two conditions. First, consider node x. A node y is a candidate for x's auxiliary set, ς_x, if the *SNR* at x for a transmission from y, $\varepsilon_{y,x}$ satisfies the following condition

$$\varepsilon_{y,x} \geq p \times \beta \tag{5.1}$$

Table 5.1: New Slot in RMTS

Mini slot #1	Mini slot #2	...	Mini slot #N_a	Original slot in the Lyui's algorithm

where p(»1) is a parameter which decides how close the auxiliary nodes are to the primary nodes. Lyui's protocol ensures that primary transmissions have a *SINR* value that satisfies (3.1). However, if an auxiliary node transmits in the place of primary node the *SINR* may not satisfy (3.1) at all of the auxiliary node's neighbors. Furthermore, the auxiliary transmissions create a different multiple access environment than the primary transmissions and this may disrupt other scheduled transmissions. So ideally we want the value of p to be such that the auxiliary node creates multiple access interference similar to that of the primary node. We show that the increase in multiple access interference is small when compared with the improvement in delay and capacity of the network. Once a node chooses the auxiliary nodes, it gives them auxiliary numbers, denoted by *A.No* which decides the order in which the auxiliary nodes attempt to transmit when the primary node does not.

Each slot is extended by N_a *mini-slots* as shown in Table 5.1. The reason we do this is because each of the auxiliary candidates listens for a transmission from the primary candidate during these mini-slots. By the end of the mini-slot 1 if the auxiliary node with *A.No* equal to 1 does not detect a transmission it will transmit its packet if it has one. By the end of mini-slot 2 if the auxiliary node with *A.No* equal to 2 still does not detect a transmission it will transmit its packet and so on. So it is essential that all the auxiliary nodes of a primary node are able to detect each others transmissions. In other words, the nodes of an auxiliary set of a primary node must form a clique. This is our second condition.

5.1 Neighbor Table

Next we describe the neighbor table. The typical row of the neighbor table, $\mathcal{N}$, is shown in Table 5.2. The first column is the neighbor, which is a in this case. The second entry is the color of the node a. Third column is the maximum color in the 2-neighborhood of node a. The fourth column indicates if node a is a 1-neighbor or a 2-neighbor.

Table 5.2: Typical Entry of the Neighbor List ($\mathcal{N}$) for a Node

Node	Color	Max Col	Type	SNR	A.No
a	2	10	1	95	1

The fifth column is a measure of the signal quality to the node a. Only 1-neighbors will have an entry in this column. The last column is the auxiliary number *(A.No)* that node a has assigned to this node. It has a value between 0 and N_a. The value is 0 if the node in the first column does not nominate the node forming the table as an auxiliary node. The value is between 1 and N_a if the node forming the table is an auxiliary node for the node in the first column.

5.2 Choosing the Auxiliary Nodes

Each node maintains an estimate of the *SNR* for the transmissions it receives from each of its 1-neighbors and stores this value in its neighbor table. Each node also stores a copy of the neighbor tables of their 1-neighbors. Utilizing the neighbor tables and equation (5.1), a node forms a list of eligible nodes, $\mathbb{S}$. Nodes in $\mathbb{S}$ satisfy (5.1). The node then builds a subgraph in which the vertices are the nodes in $\mathbb{S}$ and itself. An edge exists only if both the vertices list the other as a 1-neighbor in the set of stored neighbor tables. The second condition is that the nodes in the auxiliary set must form a clique in this induced subgraph.

Forming a maximum clique is a NP hard problem, but there are several algorithms with acceptable run times. For example, see [32] and [33] for algorithms that are efficient for graphs with hundreds of nodes.

Using the induced subgraph and one of the algorithms described above the node forms a maximal clique, $\mathbb{A}$. Once the maximal clique of auxiliary nodes is formed, the node chooses a random subset of N_a of these nodes from $\mathbb{A}$ and forms the auxiliary set, ς. The order in which the nodes in ς are chosen also decide their auxiliary number *(A.No)*. The node which is first in ς has *A.No* equal to 1 and the node which is last in the ς has *A.No* equal to N_a.

The node then includes its auxiliary set in its control packet (with details of other 1-neighbors). If a node finds itself in the auxiliary set of a primary node then it updates it neighbor table and includes its index in ς as its *A.No* value. For example, assume node x sends an auxiliary set that lists node y in position 2. Node y will update the entry for node x in its neighbor table to set the value of *A.No* equal to 2.

If the node is a primary node and it has a packet, it begins its transmission in the first mini-slot. A node is a auxiliary transmitter in a time slot if the node that is the primary transmitter, say x, has selected node y as part of *x's* auxiliary set (ς_x) and if the time slot is the primary time slot for the node x. For Lyui's algorithm, the *primary time slot* for a node is the slot number in its frame that is equal to its color number. Note that a node is a candidate to transmit in other slots within its frame, but it is guaranteed to be assigned to transmit only in its primary time slot. A node that is an auxiliary transmitter in a time slot uses its auxiliary number (*A.No*) to determine the mini-slot in which it is a candidate to transmit. If the auxiliary node does not detect a transmission in any of the earlier mini-slots and it has a packet, it begins its transmission in mini-slot *A.No*.

One characteristic that sets our protocol apart from other dynamic schedule based protocols is fairness. Here every node is guaranteed a slot in the frame. Only when a node does not utilize its assigned slot other nodes try to recover that slot. Also, our extension can be used with any distributed schedule based protocol which assigns slots to the nodes because our protocol does not depend on how these slots are assigned.

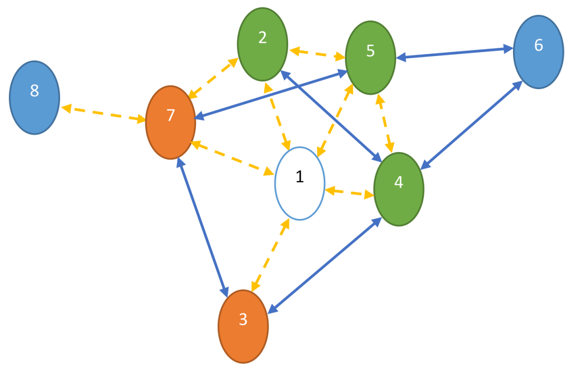

Figure 5.1: An Example Network

To illustrate how a primary node forms its auxiliary set consider the example in Figure 5.1 that shows the 2-neighborhood of node *1*. Assume that the value of N_a is five. For simplicity assume the node number and color number are the same. Nodes that are connected by bi-directional lines are 1-neighbors. Moreover, if a line is solid, the *SNR* of that link is less than $p\times\beta$ and if the line is dashed, the *SNR* of that link is greater than $p\times\beta$. Initially node 1 forms the set of eligible auxiliary nodes $\mathbb{S}_1 = \{2,3,4,5,7\}$. Node 1 then forms the maximum clique $\mathbb{A}_1 = \{2,4,5\}$. From the maximum clique node 1 selects N_a nodes at random and in a random order to form the auxiliary set ς_1. In this example there are fewer than five nodes in $\mathbb{A}_1$ so ς_1 is same as $\mathbb{A}_1$ except possibly the order of the nodes is permutated. Node 1 sends the auxiliary set to its 1-neighbors. When nodes 2, 4, and 5 receive ς_1, they will update their neighbor tables and for the entry for node 1, each node will set its *A.No* equal to its position in the auxiliary set.

Chapter 6

SIMULATION INVESTIGATIONS

A custom simulation program is utilized to investigate the performance of our new channel-access protocol. The simulation is time slotted and models the physical and link layers as described in the previous sections. The network layer is included to relay packets from their sources to destinations using fixed forwarding tables. A centralized routing algorithm [34] is used to create the initial tables, and the details of a distributed routing protocol are not included.

6.1 Simulation Parameters

We assume that there are 100 nodes in the network and the nodes are located randomly with a uniform distribution in a fixed area. All simulations are performed for three different densities and the density is modified by changing the size of the area. The densities are showed in the Table 6.1. The average hops is the average number of hops required to deliver packets to their destinations. We define the diameter of a network as the maximum number of hops that a packet travels to reach its destination.

Table 6.1: Types of densities used

Name	Density	Avg # of Hops	Diameter
D-1	1 node per 25 sq.m ($1/25^2$)	1.2	2
D-2	1 node per 50 sq.m ($1/50^2$)	1.9	4
D-3	1 node per 75 sq.m ($1/75^2$)	3.0	7

The transmission power is selected so that the *SNR* is equal to β if the distance between a transmitter and receiver is 200 m. At the start of each time slot, each node generates one packet with probability γ/N, where γ is the expected number of packets generated per slot in the network. We assume that size of the queue at each node has the capacity to store ten packets. If the queue is full when a packet is received or generated it is discarded. The value of the time to live (TTL) counter is set equal to 200 slots, and a packet not delivered to its destination within 200 slots from when it was generated is discarded.

No link or end-to-end acknowledgments are included in the simulation investigations with an exception for implementation of Seedex [15], where as the part of the MAC protocol the receiving node has to send an acknowledgment after it successfully decodes the packet and in the event that the transmitting node does not receive an acknowledgment at the end of the slot it will attempt to retransmit that packet in a later slot.

We compare three network metrics: end-to-end completion rate, end-to-end delay, and throughput. *End-to-end completion rate* is the percentage of packets that are successfully delivered to their final destinations. *End-to-end delay* is the average number of slots between the time at which a packet is generated and the time at which the packet reaches its final destination. *Throughput* is the average number of packets delivered to the final destination per slot.

Table 6.2: Simulation Parameters

Parameter	Symbol	Value
Number of Nodes	N	100
Number of Slots	Slots	4000
Numbers of Simulations	Sims	100
Time to Live	TTL	200 slots
Queue Size	Qs	10 packets
Radius	R	200 m
Number of Auxiliary Nodes	N_a	5
Multiplication Factor	p	10

Table 6.3: System Parameters

Parameter	Symbol	Value
Chip Duration	T_c	$2.9 * 10^{-7}$
Spreading Factor	N_s	128
Wavelength	λ	0.125
Path Loss Exponent	α	3.5
SINR Threshold	β	8
White Noise Power	N_0	$4 * 10^{-21}$

Statistics for each network are obtained over 4000 slots. Results are averaged over 100 random networks. Part of the performance evaluation investigates values for N_a and p, and we find values that result in good network performance for the scenarios considered here.

For all other simulations, N_a is set equal to five and p is set equal to ten. All the simulation parameters are listed in Table 6.2 and the sytem parameters used in these simulations are listed in 6.3.

6.2 Simulation of Other Protocols

We compare the performance of our protocol with three other protocols. The first one is Lyui's protocol. The other two protocols are described in this section.

6.2.1 Seedex

The key idea of Seedex [15] is to have a pseudo random schedule. In every slot a node is either in a possibly transmit (PT) mode with a probability p_s or in a listen (L) mode with a probability 1-p_s, where p_s is a predetermined parameter chosen to maximize the throughput. Thus, the modes of the nodes in a slot are determined by an i.i.d. Bernoulli sequence which is generated by a psuedo-random number generator. Nodes publish seeds of their psuedo-random number generators to their 2-neighbors. This is the only information needed by a node to calculate the modes of the nodes in its 2-neighborhood.

When node A wants to communicate with node B it waits for a slot in which A is in PT mode and B is in L mode. Node A calculates the number, n, of 1-neighbors of node B which are also in PT mode. Node A transmits in that slot with a probability $\min\{\frac{\alpha}{n+1}, 1\}$, where α is a parameter which controls how aggressively a node attempts a transmission. The values of p_s and α are taken from [15]. Accordingly we set p_s equal to 0.21 and authors in [15] suggest two different values for α (2.5 and 1.5). We set α equal to 2.5 because it results in better performance in our investigations.

6.2.2 Unlimited RMTS (u-RMTS)

In RMTS once a node forms the maximal clique $\mathbb{A}$ from the set of eligible nodes $\mathbb{S}$, it chooses N_a nodes at random to form the auxiliary set ς. Only the nodes that are in ς will attempt to transmit when the node forming the list does not transmit in its assigned slot. In the u-RMTS protocol an auxiliary set is are calculated as with our new protocol but the size of the auxiliary set is not limited (i.e., we set ς equal to $\mathbb{A}$). Note, we do not model the additional overhead due to the large number of mini-slots as compared to our system with N_a fixed. This modification maximizes the opportunity for some auxiliary node to transmit, and highlights the effect on network performance due to a practical limit on the size of an auxiliary set. For a network in which all links satisfy (5.1) a transmission is successful in each slot unless no node has a packet. As the network density decreases, so does the number of eligible nodes.

Chapter 7

RESULTS

First, consider the end-to-end completion rate. Figures 7.1, 7.2 and 7.3 show end-to-end completion rate for investigations with density D-3, D-2, and D-1 respectively.

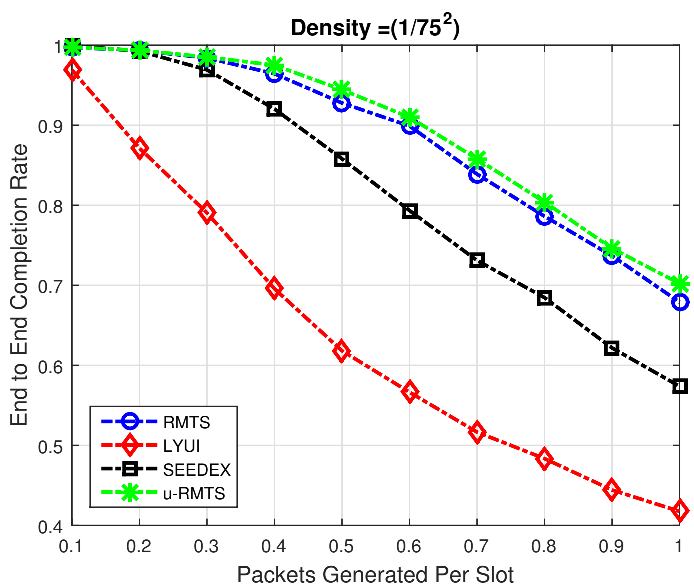

Figure 7.1: End-to-end completion rate for low density

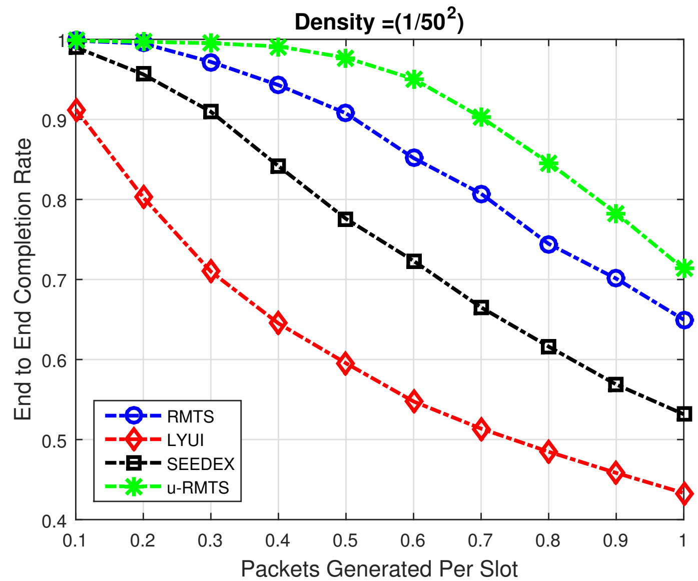

Figure 7.2: End-to-end completion rate for medium density

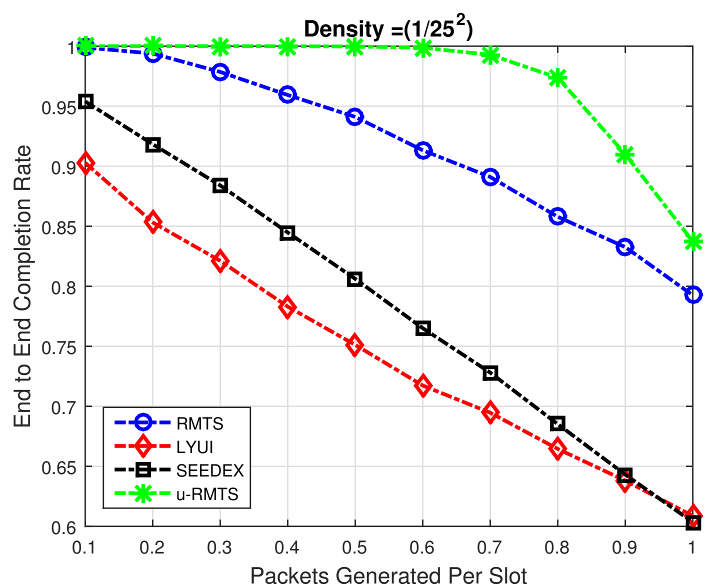

Figure 7.3: End-to-end completion rate for high density

We consider network performance to be poor when the end-to-end completion rate is low. We define the end-to-end completion rate threshold, Γ, as the maximum value of the packet generation rate, γ, for which the end-to-end completion rate is greater than 90%. Table 7.1 lists the values of Γ for each protocol. In Table 7.1, *L* denotes Lyui's protocol, *S* denotes Seedex, and *R* denotes RMTS. The last two columns in Table 7.1 show the percentage improvement in Γ of our protocol over Lyui's protocol and Seedex respectively. It is clear from these results that our approach leads to significant improvement over traditional scheduling algorithms like Lyui's protocol, in which the slot assignments are fixed. Our protocol also out performs Seedex in which slots are dynamically assigned depending on the traffic at the nodes.

Table 7.1: Values of Γ

Density	L	S	R	% imp in Γ	
				L	S
D-1	.10	.25	.66	520%	160%
D-2	.11	.31	.51	360%	63%
D-3	.17	.43	.60	250%	40%

Figures 7.4, 7.5, and 7.6 illustrate the end-to-end delay for investigations with density D-3, D-2, and D-1 respectively.

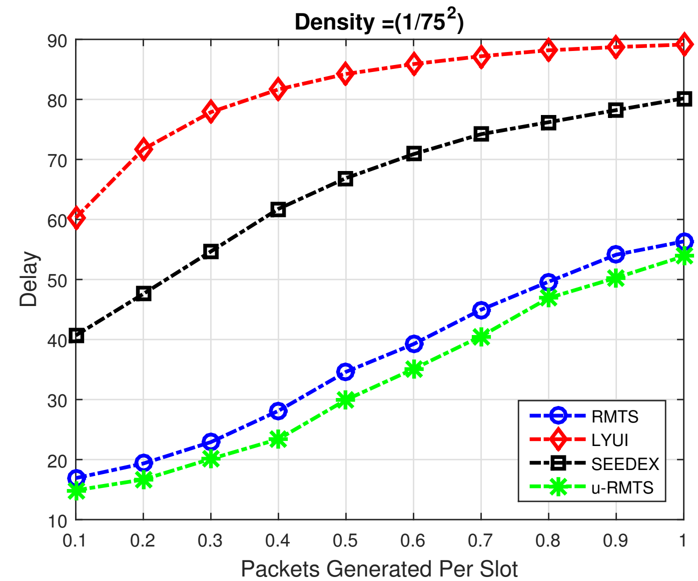

Figure 7.4: Delay for low density

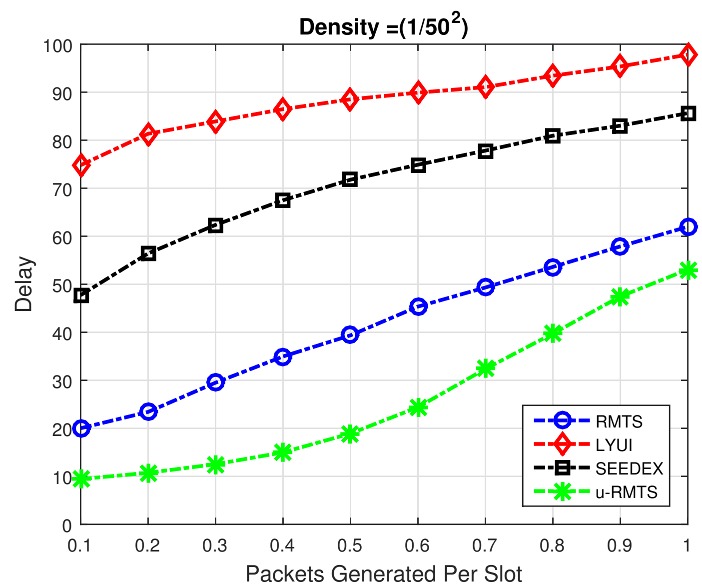

Figure 7.5: Delay for medium density

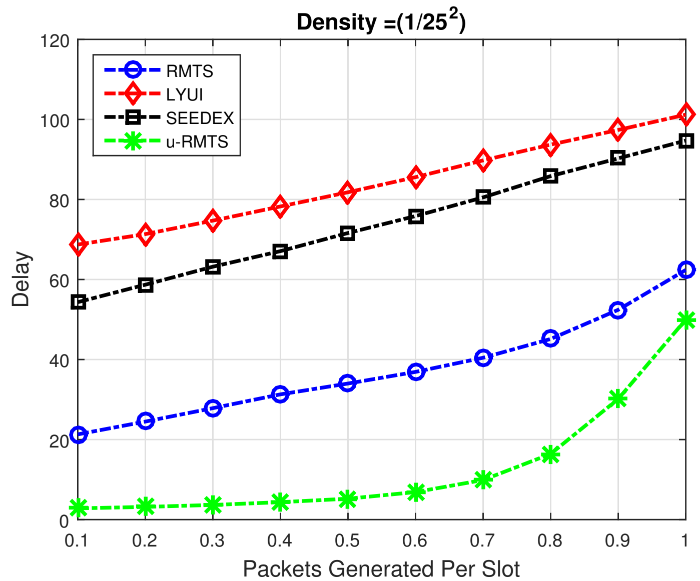

Figure 7.6: Delay for high density

From these graphs it can be noted that for investigations with density D-3, D-2, and D-1 the maximum delay (at $\gamma = 1$) in the case of our protocol is almost same as the minimum delay (at $\gamma = 0.1$) in the case of Lyui's protocol and our protocol provides significant improvement in delay over Seedex as well. We consider the performance of the network if the packet generation rate (γ) is greater than Γ as poor. We have already seen that RMTS has higher values for Γ than both Lyui's protocol and Seedex. We compare the delay of RMTS with delay of Lyui's protocol and Seedex at packet generation rate equal to their $\Gamma's$, respectively. For investigations with density D-3, Lyui's protocol has $\Gamma = .17$, and at this traffic generation rate there is about a 73% decrease in the delay of RMTS over Lyui's protocol. Seedex has $\Gamma = .43$, and at this traffic generation rate there is about a 53% decrease in delay of RMTS over Seedex. Table 7.2 lists these values in other scenarios. Our protocol provides a significant improvement in the average delay in all scenarios.

Table 7.2: Values of Percentage Decrease in Delay

Density	% dec in delay over	
	Lyui's	Seedex
D-1	69%	57%
D-2	73%	52%
D-3	74%	53%

We examine jitter by measuring the standard deviation, std, of they delay. Figures 7.7, 7.8, and 7.9 show the standard deviation of the delay for the investigations with density D-3, D-2, and D-1 respectively.

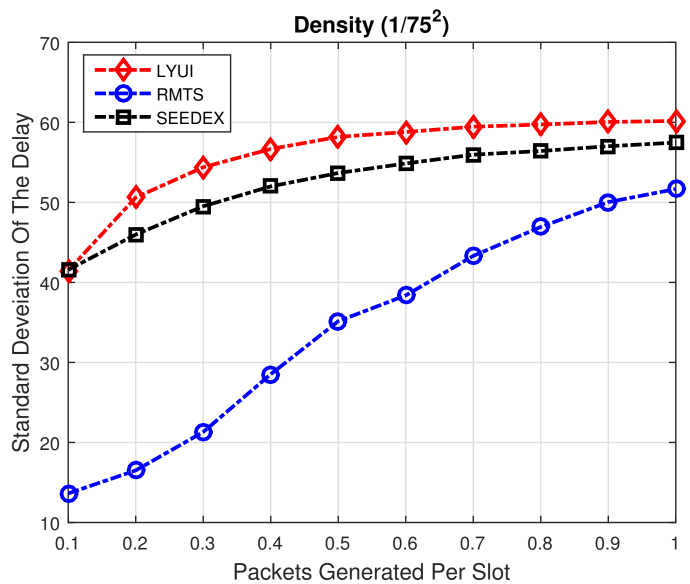

Figure 7.7: Standard deviation of the delay for low density

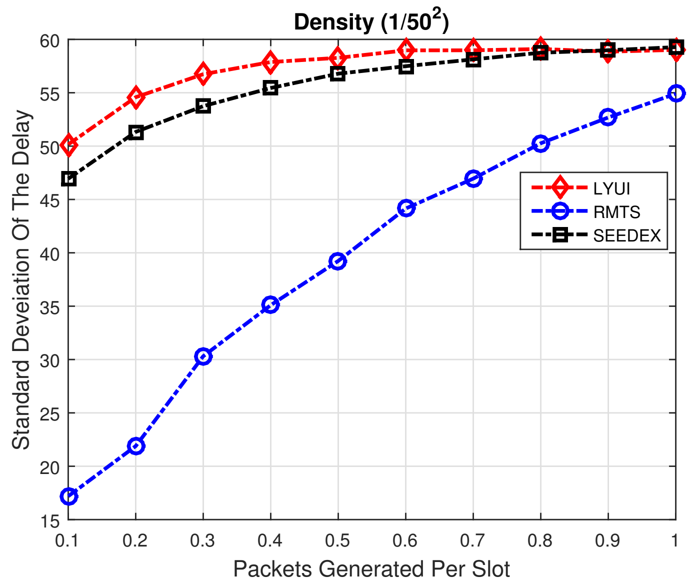

Figure 7.8: Standard deviation of the delay for medium density

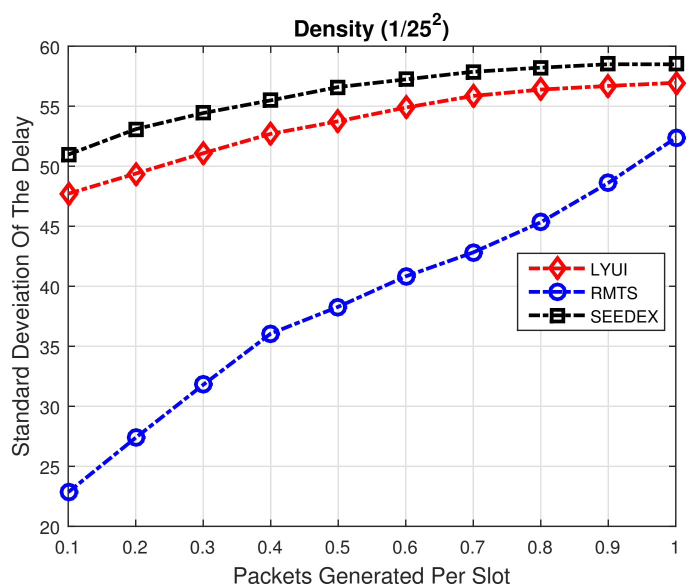

Figure 7.9: Standard deviation of the delay for high density

Once again we compare the std of RMTS with that of Lyui's protocol and Seedex at their $\Gamma's$, respectively. Table 7.3 lists the values of percentage decrease in STD of delay over Lyui's protocol and Seedex.

Table 7.3: Values of Percentage Decrease in STD of Delay

Density	% dec in std of delay over	
	Lyui's	Seedex
D-1	52%	45%
D-2	65%	43%
D-3	67%	42%

This improvement in delay leads to improvement in throughput. Figures 7.10, 7.11, and 7.12 show the throughput for investigations with densities D-3, D-2, and D-1 respectively.

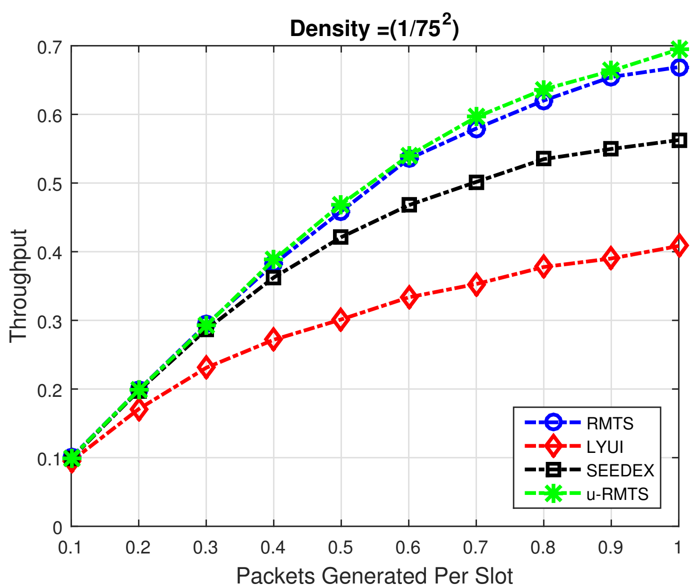

Figure 7.10: Throughput for low density

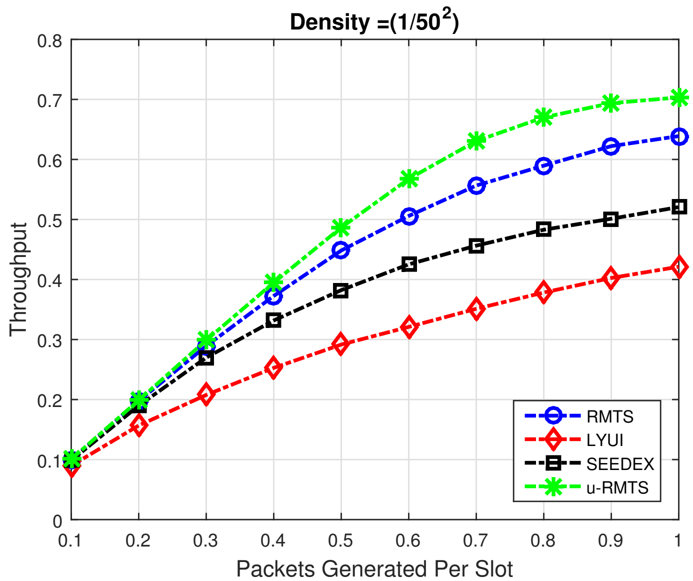

Figure 7.11: Throughput for medium density

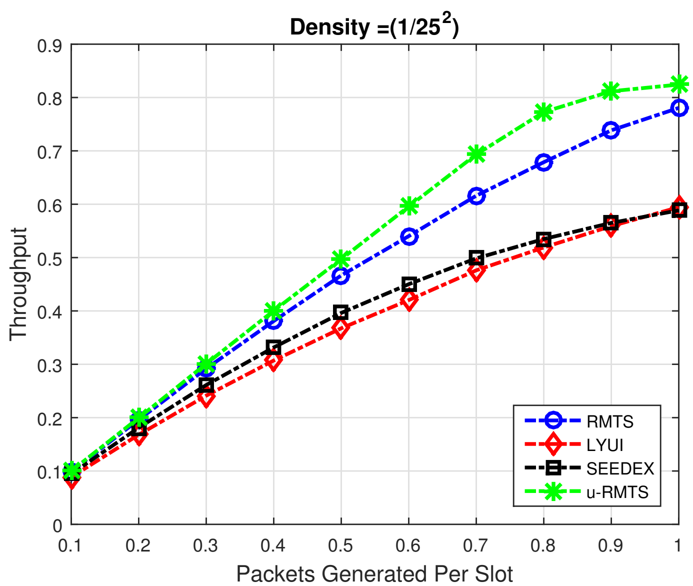

Figure 7.12: Throughput for high density

7.1 Slot Utilization

We define slot utilization as the fraction of the assigned slots in which packets are transmitted. For RMTS a slot assigned to a node is consider utilized when either the primary node or one of its auxiliary nodes transmit in that slot. Consider a node i and let the probability that it will not have a packet to transmit in its assigned slot be denoted by W_i. For simplicity we assume that all auxiliary nodes of node i have same probability that they do not have a packet to transmit in that slot. We can approximate the probability that node i will utilize the slot allotted to it by U_i.

$$\begin{aligned} U_i &\approx 1 - W_i + W_i \sum_{i=1}^{i=N_a} W_i^{i-1}(1 - W_i) \\ &= 1 - W_i^{N_a+1} \end{aligned} \tag{7.1}$$

The average slot utilization over all the nodes in the network is

$$U_{avg} \;=\; 1 - \frac{\sum_{i=1}^{N} W_i^{N_a+1}}{N} \tag{7.2}$$

Values of W_i are estimated in our simulations. We show the utilization predicted by 7.2 and the slot utilization obtained from the simulations for the investigations with density D-3, D-2, and D-1 in Figures 7.13, 7.14, and 7.15.

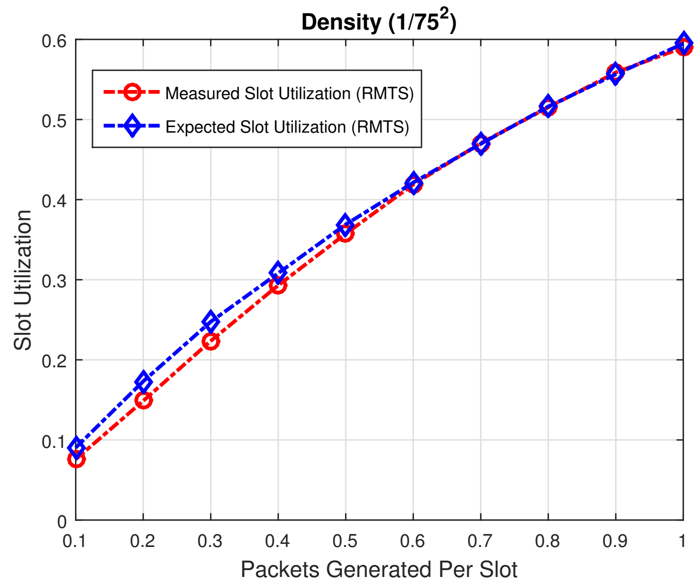

Figure 7.13: Slot utilization for low density

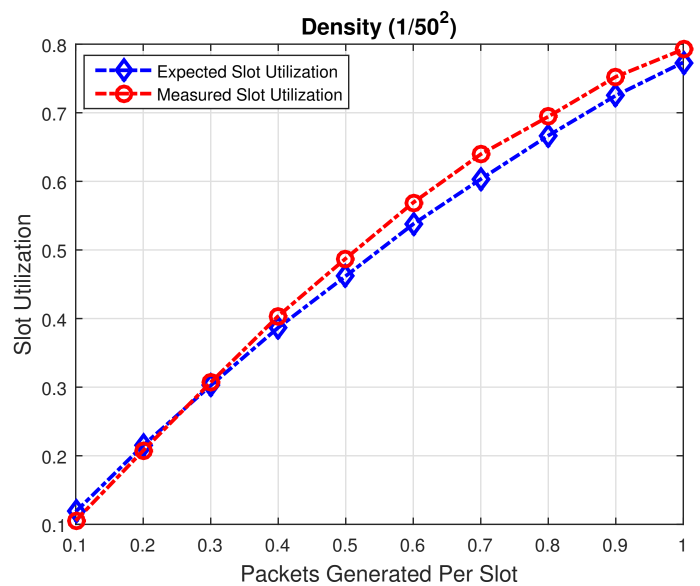

Figure 7.14: Slot utilization for medium density

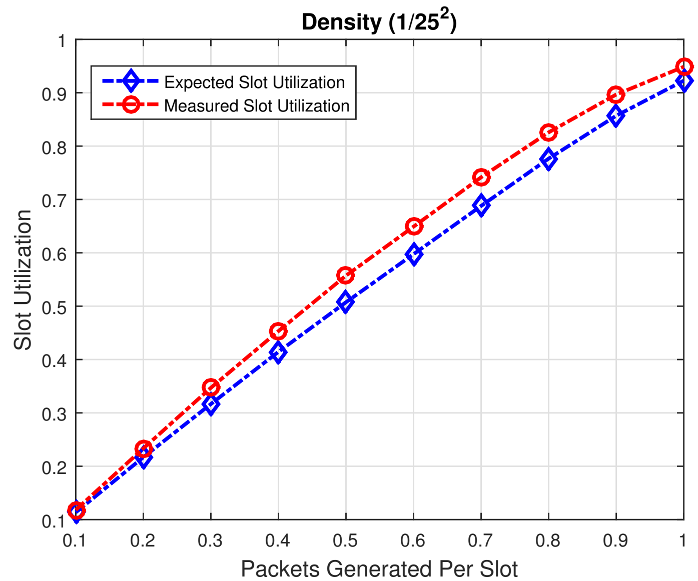

Figure 7.15: Slot utilization for high density

For investigations with density D-3, the value of Γ for RMTS is 0.61. At this packet generation rate slot utilization of RMTS is about 42%. Even though the slot utilization is low the network performance deteriorates because there are a few nodes which are acting like bottlenecks. In MANETs bottlenecks are a serious problem because these nodes effect all the traffic that is routed through them. One way to circumvent this problem is to utilize link metrics with the routing protocol to take into account the slot utilization of each node. Nodes with high slot utilization are prime candidates to become bottlenecks in a network. So the link metric should assign a higher cost for including that node in a route. Also the routes have to be calculated periodically. In the next section we present one such link metric.

7.2 Link Metric Using Slot Utilization

For the previous investigations we used the link cost function described in Section 3.3. Now we present a new link metric which is a variation of the one presented in [27] that takes into account the average slot utilization of each node. When the routes are calculated for first time the slot utilization of each node is zero, hence the new link metric is same as the old one. However, in the new method the nodes have to exchange average slot utilization information and update the link metric of every link periodically. The routes are re-calculated after every such update. We use a centralized algorithm to recalculate all routes after each *UPD* slots. The new link metric denoted by, $C_{i,j}^{new}$, is defined below.

$$C_{i,j}^{new}(\varepsilon_{i,j}, U_j) = \begin{cases} \infty, & \varepsilon_{i,j} < \beta \\ \{1 - log(\frac{\varepsilon_{i,j}-\beta}{\beta})\}(1+U_j), & \beta \leq \varepsilon_{i,j} < 2\beta \\ (1+U_j), & \varepsilon_{i,j} \geq 2\beta \end{cases} \tag{7.3}$$

Simulations are run for UPD equal to 64, 128, and 256. We compare the resulting three metrics end-to-end completion rate, delay, and throughput with the case with no periodic routing updates. Figures 7.16, 7.17, and 7.18 show end-to-end completion rates for investigations with densities D-3, D-2, and D-1 respectively.

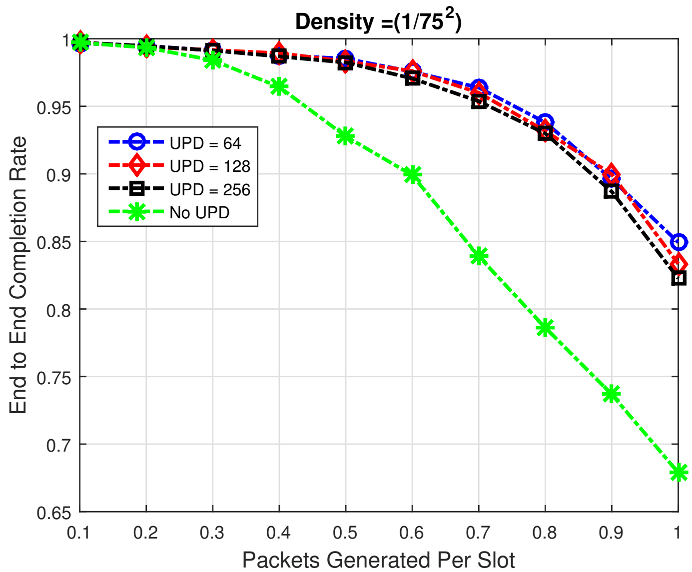

Figure 7.16: Comparison of end-to-end completion rate for different values of UPD for low density

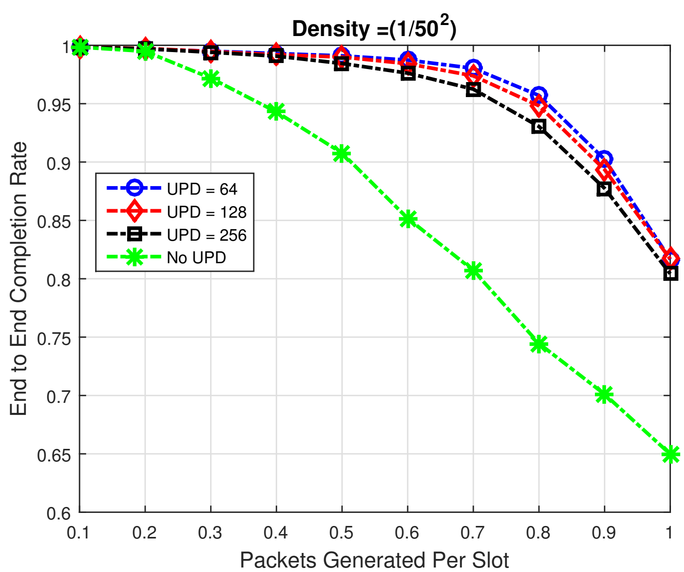

Figure 7.17: Comparison of end-to-end completion rate for different values of UPD for medium density

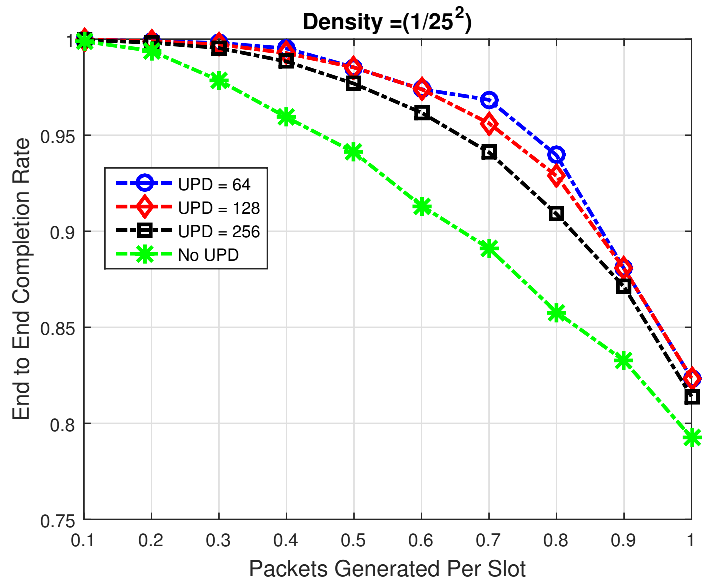

Figure 7.18: Comparison of end-to-end completion rate for different values of UPD for high density

In Table 7.4 we list the values of Γ for investigations with UPD equal to 64, 128, and 256. We also list the percentage improvement in the value of Γ over the scenario without periodic routing updates.

Table 7.4: Values of Γ

Density	No UPD	UPD = 64		UPD = 128		UPD = 256	
	Γ	Γ	% imp	Γ	% imp	Γ	% imp
D-1	.66	.87	31.4 %	.86	30.3 %	.82	24.9 %
D-2	.51	.91	75.7 %	.89	72.9 %	.86	66.9 %
D-3	.60	.89	49.3 %	.90	50.3 %	.87	45.7 %

Figures 7.19, 7.20, and 7.21 show the delay for investigations with densities D-3, D-2 and D-1 respectively.

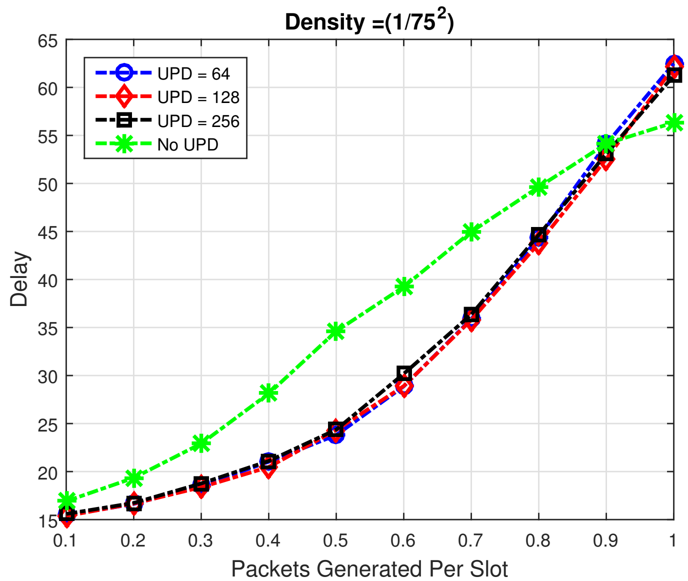

Figure 7.19: Comparison of delay for different values of UPD for low density

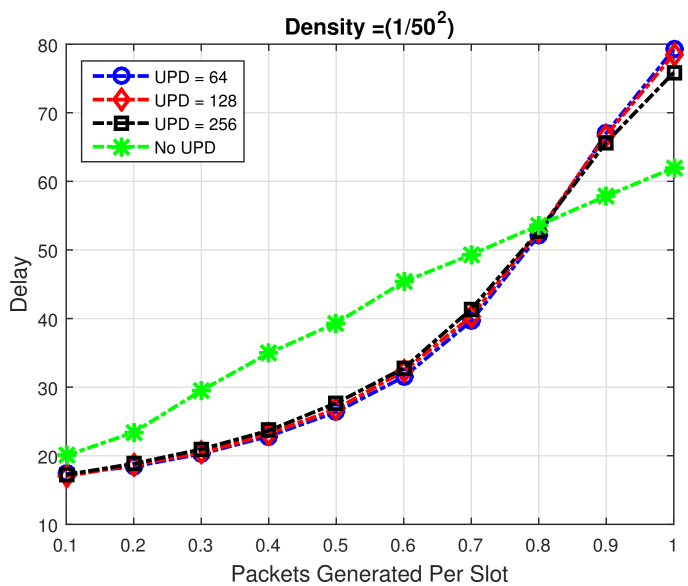

Figure 7.20: Comparison of delay for different values of UPD for medium density

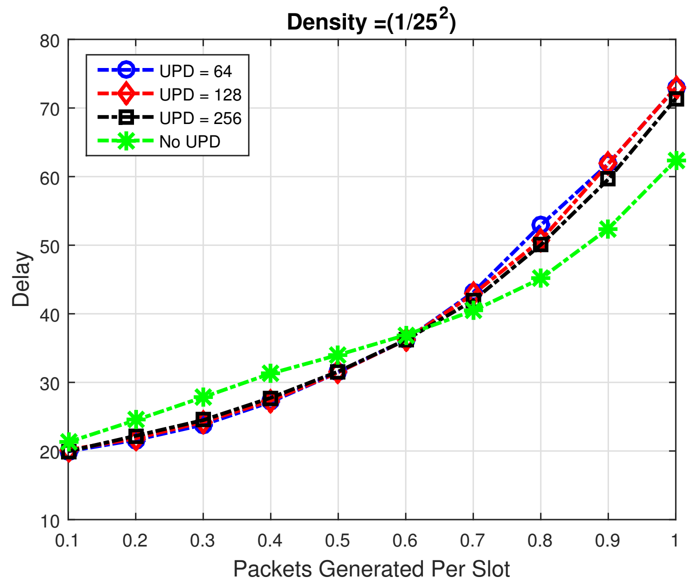

Figure 7.21: Comparison of delay for different values of UPD for high density

We compare the delay for investigations with UPD equal to 64, 128, and 256 with the delay for investigation without periodic routing at the packet generation rate equal to Γ of the latter case. Table 7.5 lists values of percentage decrease in delay for investigations with our new metric over the investigation with the old metric.

Table 7.5: Percentage improvement in delay with new cost metric

Density	UPD = 64	UPD = 128	UPD = 256
D-1	-3.3%	-2.98%	-1.5 %
D-2	32.6%	31.4%	29.4%
D-3	26.5%	26.4%	23.2%

Figures 7.22, 7.23, and 7.24 show the comparison between throughput for investigations with the new metric and throughput for investigations with the old metric for densities D-3, D-2 and D-1 respectively.

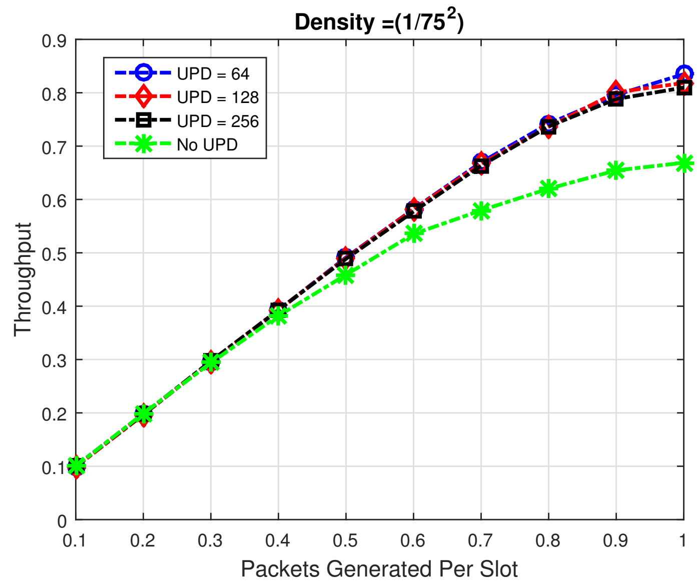

Figure 7.22: Comparison of Throughput for different values of UPD for low density

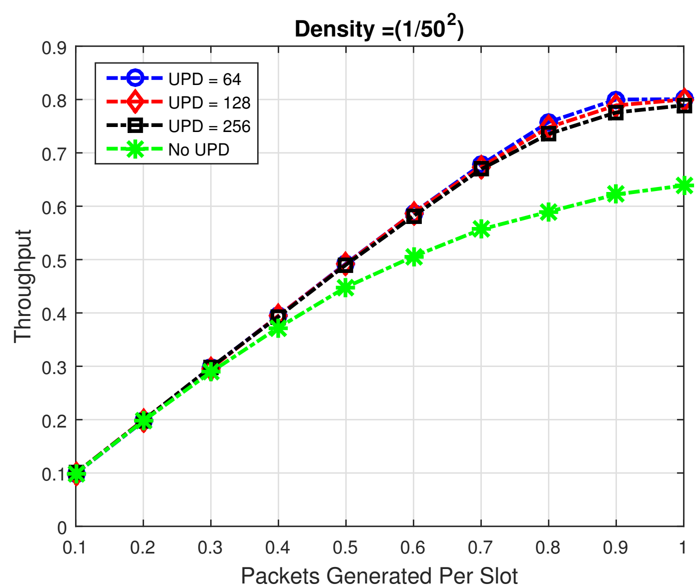

Figure 7.23: Comparison of Throughput for different values of UPD for medium density

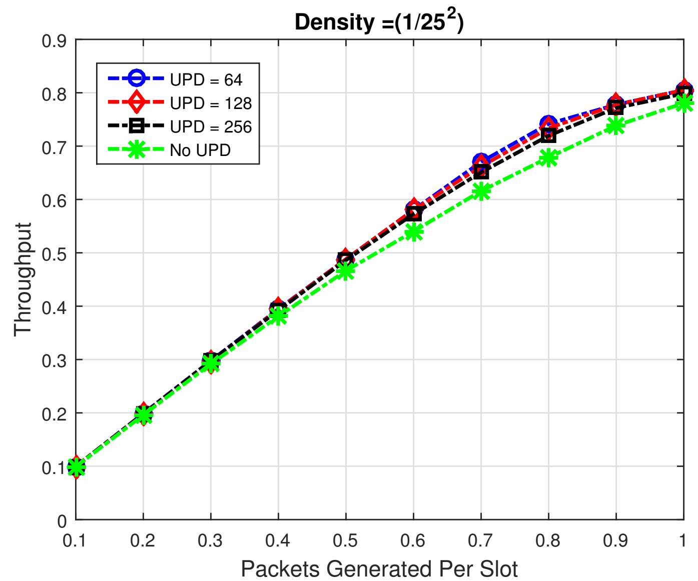

Figure 7.24: Comparison of Throughput for different values of UPD for high density

The new metric leads to significant improvement in end-to-end completion rate in all the cases. By using the new metric and calculating the routes periodically we are assigning higher costs to the links involving the bottlenecks and thereby reducing the effect of the bottlenecks in the network. There is considerable decrease in delay for investigations with densities D-2 and D-3 however in the case of D-1 the delay is almost same, with the new metric resulting in a small increase in the delay. In the investigations with density D-1 most of the packets only travel one hop hence there will not be much change in the delay even when we use the new metric. Also the performance of the network is comparable for three different values of UPD. Using UPD equal to 256 is recommended as this case incurs the least amount of overheard required to periodically calculate the routes.

7.3 Choosing the Values of the Parameters N_a and p

The key idea of RMTS is that each node selects auxiliary nodes such that one of them can transmit in its place if the node does not use its assigned slot. This improves the chance that an assigned slot will be utilized and is the reason for the improvement in the performance over a scheme where only the primary node is permitted to transmit. In RMTS there are two parameters the can be tuned. The first is N_a which is the limit on the number of auxiliary nodes a primary node can choose as candidates for a substitute transmission. The second is p which is a multiplication factor that controls how similar the multiple-access interference of the substitute transmission is compared to the primary transmission. We have designed RMTS for mulit-hop networks. Hence, we focus the investigations for selecting N_a and p on networks with density D-3.

7.3.1 Choosing the Value of N_a

We have run simulations for different values of N_a ranging from N_a=1 to N_a=N. Figures 7.25, 7.26, and 7.27 show end-to-end completion rate, delay and throughput of RMTS for different values of N_a. From these graphs it is clear that increasing the value of N_a beyond five does not significantly improve the performance. Hence for all the investigations shown in this manuscript we have set the value of N_a equal to five.

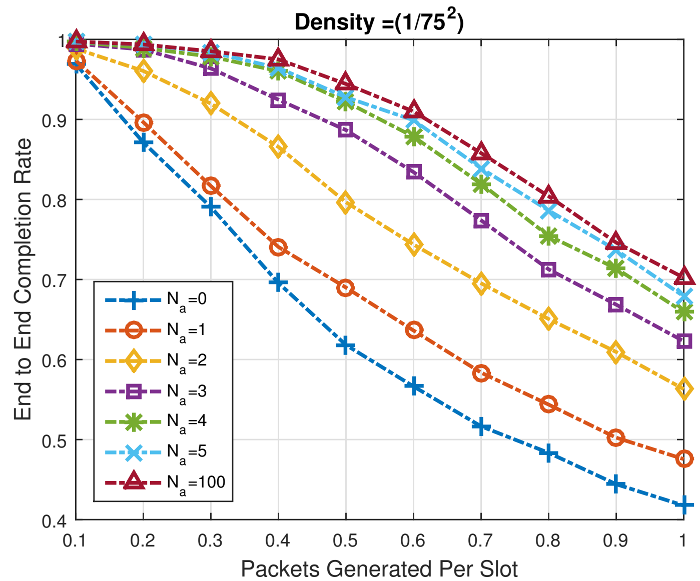

Figure 7.25: End-to-end completion rate for low density

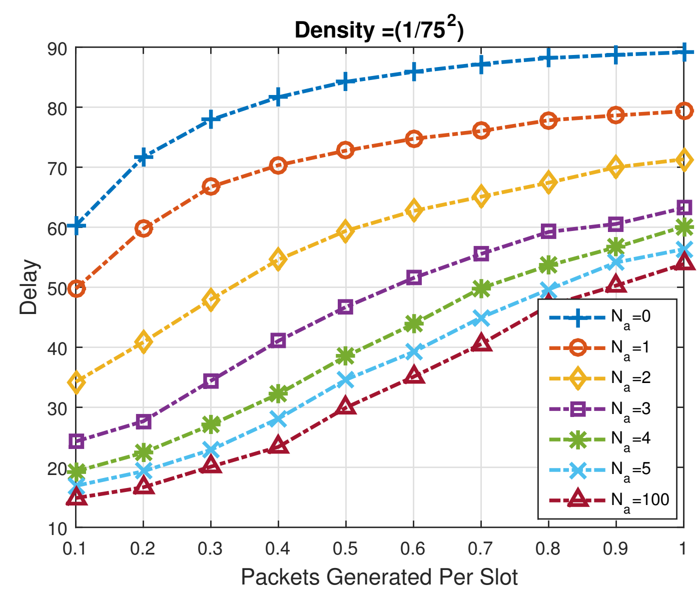

Figure 7.26: Delay for low density

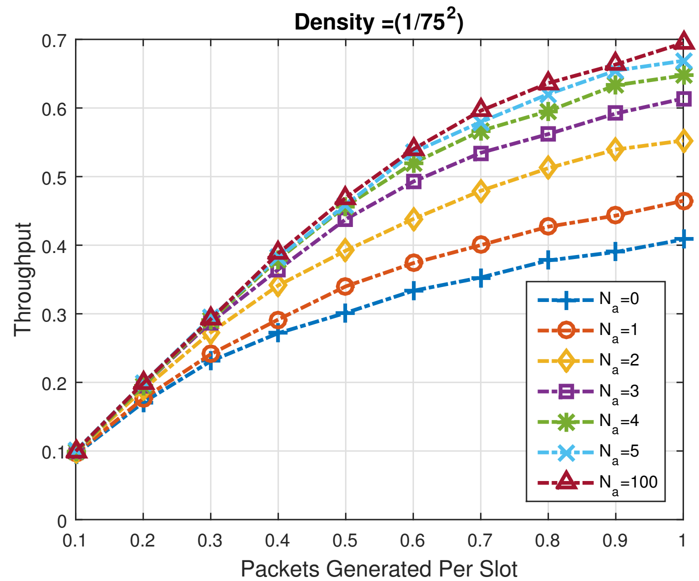

Figure 7.27: Throughput for low density

7.3.2 Choosing the Value of p

As described in Chapter 5, p is a factor which controls the multiple access interference caused by an auxiliary transmission instead of a primary transmission. With a large value of p we can ensure that the multiple-access interference caused by an auxiliary transmission is similar to that caused by a primary transmission. However this limits the size of the pool of 1-neighbors from which the auxiliary nodes are chosen. With a small value of p we can ensure that there is a large of pool of 1-neighbors from which the auxiliary nodes are chosen. However the multiple-access interference caused by these transmissions can be different from the one caused by the primary transmissions and can result in failure of other scheduled transmissions in that slot. We have investigated values of p ranging from 1 to 18. Network performance improves as the value of p increases until it achieves its maximum value for p approximately equal to 10. For large values of p the network performance decreases. In Figure 7.28 we show the value of Γ for different values of p, and Γ is maximum for p approximately equal to ten. For simulation results shown in this manuscript, p is equal to ten. Another point to note is that for all values of p the performance is significantly better than Lyui's protocol.

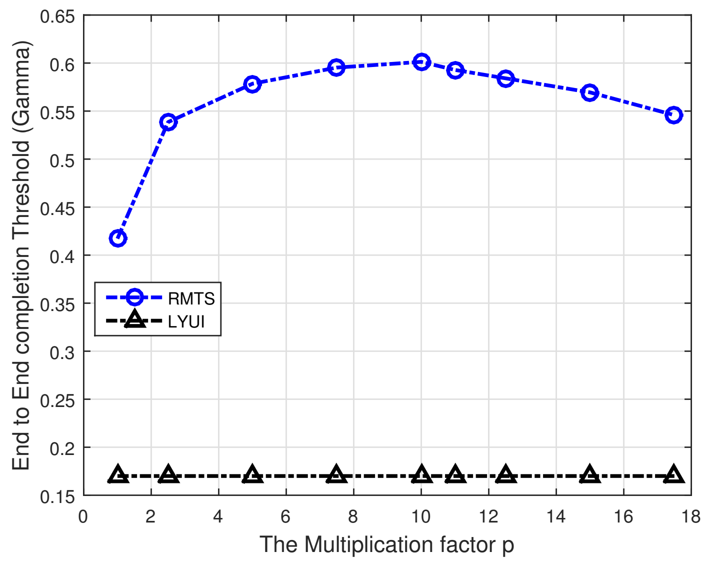

Figure 7.28: Values of Γ for different values of p

The values for N_a and p may not be optimum for densities D-1 and D-2. However RMTS still out performs Lyui's protocol and Seedex in the investigations with densities D-1 and D-2. In general we can choose conservative values for N_a (3-5) and p (10-15). No matter what the density of the network these values will result in significant improvement over a scheme in which the un-utilized slots are not recovered. However if we are interested in networks of a particular density then these parameters can be tuned to maximize the performance.

Chapter 8

CONCLUSION

We have designed and investigated a new mechanism for recovering un-utilized time slots in transmission scheduling protocols for channel access in a MANET. In most of the existing scheduling algorithms once a slot is assigned to a particular node if that node does not have a packet to transmit, that slot is not utilized. This leads to poor network performance. We propose a protocol to salvage these un-utilized slots by allowing a substitute transmission in place of the scheduled one. The node that is permitted to make a substitute transmission is carefully selected so that the multiple-access interference is similar to the level created with the original schedule. A modest amount of additional overhead is required in each time slot to include mini-slots for detecting which candidate node can utilize the opportunity. In addition, the size of the periodic control transmissions is slightly increased to support additional information about which nodes are selected as auxiliary transmitters. In this manuscript we use Lyui's algorithm to demonstrate our protocol. However, our approach can be used to modify any scheduling algorithm in which slot assignments are fixed but may go un-utilized due to fluctuations in traffic demands.

Our approach retains the advantages of a traditional scheduling algorithm. Namely, our scheme does not change the fairness of scheduling, that is, all the nodes have a guaranteed opportunity to transmit at least once in a frame. Also, our scheme maintains efficient channel access under high traffic loads and does not introduce contention and its associated stability concerns. The novel addition of our approach occurs in scenarios in which there is a high variability in traffic load and unequal queueing demands at the nodes. In a traditional scheduling algorithm, a node with a large queue of packets must wait for the next frame before another opportunity to transmit occurs. In our scheme, if such a node is found in the auxiliary set of one or more nodes it has the chance for additional transmissions. If a primary node is idle, the first node in the primary node's auxiliary set with a packet makes the replacement transmission. Our investigations show that allowing the recovery transmission from among a set of nodes significantly increases the probability that a packet is serviced. Furthermore, randomizing the selection and order of the nodes from the auxiliary set provides multiple opportunities for a node to appear as a candidate and ensures opportunities are fairly shared among the auxiliary sets. Queue overflow is reduced and this in turn leads to better end-to-end completion rates.

We use extensive simulations to show that our approach results in tremendous improvement in the network performance in investigations with random networks of varying densities. Additional investigations with a link metric that selects routes other than by simply minimizing the hop count confirms prior investigations that clever use of cross-layer information is essential to improving the network performance. Our investigations incorporate a slot utilization metric and show that even with the higher utilization with RMTS compared to prior investigations there is still substantial benefit in employing link utilization in the routing decisions. We also show that this design approach is robust enough to produce better performance for a range of values of the protocol parameters (N_a and p).

The demands to support higher data rates are expected to dramatically increase for future applications. One approach to achieve higher data rates in MANETs is to take advantage of links with high channel qualities. These links can support higher order modulation schemes and hence are capable of achieving higher data rates. We plan to develop a rate adaptive MAC that is integrated with RMTS to take advantage of these high quality links. Additional future work will consider innovative link metrics and routing protocols that utilize link-layer information in network-layer protocols. Preliminary investigations have revealed that a simple link metric that sets the link cost inversely proportional to the date rate selected by an adaptive modulation protocol out performs an approach in which all the links are assigned same cost irrespective of the modulation scheme.

In a network it is common to expect varying QOS demands by different nodes. In our RMTS protocol a node picks the auxiliary nodes randomly from a pool of eligible nodes. We propose to investigate mechanisms to provide different QOS levels for nodes by prioritizing the nodes with higher QOS requirements when selecting the auxiliary nodes.

Bibliography

[1] M. Frodigh, P. Johansson, and P. Larsson, "Wireless ad hoc networking: the art of networking without a network," *Ericsson Review*, vol. 4, no. 4, p. 249, 2000.

[2] M. Conti, G. Maselli, G. Turi, and S. Giordano, "Cross-Layering in Mobile Ad Hoc Network Design," *Computer*, vol. 37, no. 2, pp. 48–51, 2004.

[3] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Cross-Layer Congestion Control, Routing and Scheduling Design in Ad Hoc Wireless Networks," *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, pp. 1–13, 2006.

[4] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," *IEEE Communications Magazine*, vol. 41, no. 10, pp. 74–80, 2003.

[5] V. Srivastava and M. Motani, "Cross-layer design: A survey and the road ahead," *IEEE Communications Magazine*, vol. 43, no. 12, pp. 112–119, 2005.

[6] S. Kumar, V. S. Raghavan, and J. Deng, "Medium access control protocols for ad hoc wireless networks: A survey," *Ad Hoc Networks*, vol. 4, no. 3, pp. 326–358, 2006.

[7] N. Abramson, "The aloha system: another alternative for computer communications," in *Proceedings of the November 17-19, 1970, fall joint computer conference*, pp. 281–285, ACM, 1970.

[8] J. Zander, "Slotted aloha multihop packet radio networks with directional antennas," *Electronics Letters*, vol. 26, no. 25, pp. 2098–2100, 1990.

[9] D. P. Bertsekas and R. G. Gallager, *Data Networks*, vol. 2. Prentice-Hall International New Jersey, 1992.

[10] P. Karn, "MACA-a new channel access method for packet radio," in *ARRL/CRRL Amateur radio 9th computer networking conference*, vol. 140, pp. 134–140, 1990.

[11] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: a media access protocol for wireless LAN's," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 4, pp. 212–225, 1994.

[12] IEEE 802.11 Working Group, "Wireless lan medium access control (mac) and physical layer (phy) specifications," 1997.

[13] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on selected areas in communications*, vol. 18, no. 3, pp. 535–547, 2000.

[14] W. P. Lyui, "Design of a new operational structure for mobile radio networks," *Ph.D. dissertation, Clemson University*, 1991.

[15] R. Rozovsky and P. R. Kumar, "SEEDEX: A MAC Protocol for Ad Hoc Networks," *2Nd ACM International Symposium on Mobile Ad Hoc Networking &Amp; Computing*, vol. 7, pp. 67–75, 2001.

[16] A. Ephremides and T. V. Truong, "Scheduling broadcasts in multihop radio networks," *IEEE Transactions on communications*, vol. 38, no. 4, pp. 456–460, 1990.

[17] R. Ramanathan, "A Unified Framework and Algorithm for Channel Assignment in Wireless Networks," *Wireless Networks*, vol. 5, no. 2, pp. 81–94, 1999.

[18] I. Rhee, A. Warrier, J. Min, and L. Xu, "DRAND: Distributed randomized TDMA scheduling for wireless ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 10, pp. 1384–1396, 2009.

[19] J. L. Hammond and H. B. Russell, "Properties of a Transmission Assignment Algorithm for Multiple-Hop Packet Radio Networks," *IEEE Transactions on Wireless Communications*, vol. 3, no. 4, pp. 1048–1052, 2004.

[20] C. D. Young, "USAP: a unifying dynamic distributed multichannel TDMA slot assignment protocol.pdf," in *IEEE Military Communications*, pp. 235–239, 1996.

[21] J. H. Kim, J. R. Cha, and H. J. Park, "New delay-efficient TDMA-based distributed schedule in wireless mesh networks," *EURASIP Journal on Wireless Communications and Networking*, no. 1, p. 369, 2012.

[22] C. Zhu and M. S. Corson, "A Five-Phase Reservation Protocol (FPRP) for mobile Ad Hoc networks," *Wireless Networks*, vol. 7, no. 4, pp. 371–384, 2001.

[23] A. Bhatia and R. C. Hansdah, "DSLR: A Distributed Schedule Length Reduction Algorithm for WSNs," *2015 IEEE International Parallel and Distributed Processing Symposium*, pp. 365–374, 2015.

[24] L. Kleinrock and M. Scholl, "Packet switching in radio channels: New conflict-free multiple access schemes," *IEEE Transactions on Communications*, vol. 28, no. 7, pp. 1015–1029, 1980.

[25] C. D. Young and J. A. Stevens, "Clique activation multiple access (CAMA): A distributed heuristic for building wireless datagram networks," in *Proc. MILCOM'98*, Citeseer, 1998.

[26] R. C. Sunlin, “A hybrid distributed slot assignment tdma channel access protocol,” in *Military Communications Conference, 1990. MILCOM’90, Conference Record, A New Era. 1990 IEEE*, pp. 934–938, IEEE, 1990.

[27] B. J. Wolf and H. B. Russell, “Immediate neighbor scheduling (INS): An adaptive protocol for mobile ad hoc networks using direct-sequence spread-spectrum modulation,” *Ad Hoc Networks*, vol. 9, no. 3, pp. 453–467, 2011.

[28] A. Ebner, H. Rohling, M. Lott, and R. Halfmann, “Decentralized slot synchronization in highly dynamic ad hoc networks,” *Fifth International Symposium on Wireless Personal Multimedia Communications, WPMC*, vol. 2, pp. 494–498, 2002.

[29] T. S. Rappaport, *Wireless communications: principles and practice*, vol. 2. Prentice Hall PTR New Jersey, 1996.

[30] B. J. Wolf, J. L. Hammond, D. L. Noneaker, and H. B. Russell, “Distributed formation of broadcast transmission schedules for mobile ad hoc networks,” *IEEE MILCOM 2004. Military Communications Conference, 2004.*, vol. 3, pp. 1–7, 2004.

[31] P. K. Appani, J. L. Hammond, D. L. Noneaker, and H. B. Russell, “An adaptive transmission-scheduling protocol for mobile ad hoc networks,” *Ad Hoc Networks*, vol. 5, no. 2, pp. 254–271, 2007.

[32] R. Carraghan and P. M. Pardalos, “An exact algorithm for the maximum clique problem,” *Operations Research Letters*, vol. 9, pp. 375–382, nov 1990.

[33] P. R. J. Östergård, “A fast algorithm for the maximum clique problem,” *Discrete Applied Mathematics*, vol. 120, no. 1, pp. 197–207, 2002.

[34] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.