Multi-Objective Scheduling for MUD based Ad-Hoc Networks

M. Bouharras   Z.Dziong   F. Gagnon   M. Haidar

Department of Electrical Engineering,
Ecole de Technologie Supérieure
Montreal, Quebec, Canada, H3C 1K3

(Mohamed.bouharras.1, Mohamad.Haidar.1}@ens.etsmtl.ca, {Zbigniew.Dziong, Francois.Gagnon}@etsmtl.ca

ABSTRACT

Common channel multi-hop Ad Hoc networks have some inherent constraints related to throughput and Quality of Service (QoS). Multiuser detection (MUD) based Medium Access Control (MAC) can relax some of these constraints and provide significant gains in throughput and Quality of Service (QoS). These gains can be realized by implementing a distributed neighborhood scheduling algorithm that needs to choose one from several possible transmission configurations in each frame. This feature allows formulating different scheduling performance objectives such as delay minimization or throughput maximization. In this paper we focus on analysis and comparison of the system performance under different objectives including multi-objective formulations. First we implement a scheduling scheme that minimize delay using Start Time Fair Queuing (STFQ) algorithm and compare its performance with scheduling that maximises the throughput. Then we formulate multi-objective functions that are used to achieve a trade-off between delay and throughput performance. One of these formulations is based on the Nash arbitration scheme from cooperative game theory. The numerical results demonstrate the flexibility and efficiency of the proposed approach.

Keywords
Multiuser detection; Multiuser reception; MAC protocol; scheduling; fairness and wireless ad hoc networks.

1. INTRODUCTION

Wireless ad hoc networks are expected to support multimedia applications in emergency disaster management and military operations. This class of mission-critical applications demands a certain level of quality of services (QoS) for proper operation. Many MAC protocols have been developed for wireless networks which are based on a common channel shared by mobile hosts in the network. These protocols, such as IEEE 802.11 are referred to as single-channel protocols. Due to relatively high probability of contentions and collisions, the performance of single-channel MAC protocols deteriorates quickly while the number of mobile hosts increases, especially for muli-hop connections [1]. To mitigate this problem, one can consider utilizing multiuser detection (MUD) or multiuser reception (MUR) CDMA technology that allows the reception of multiple CDMA channels at the same time. In [2] and [3] the authors proposed a novel MAC and scheduling algorithms to take advantage of MUD and MUR. One of the key elements of their approach is a distributed neighborhood scheduling mechanism that is based on a protocol that exchanges information between neighbors. They presented numerical results indicating significant gains in the QoS (delay) and network throughput. The objective of their scheduling mechanism was based on a principle that priority was given to the voice packets that waited the longest time in the neighborhood's queues. This principle is not necessary optimal as shown in the studies of fair queuing mechanisms. Also, the network operation can have multi-objective formulation where the scheduling should balance network throughput maximization objective with traffic class fair access optimization.

In this paper, we first propose an approach for analyzing and comparing close to optimal distributed neighborhood scheduling schemes with different objectives. The approach is based on the notion of a flow contention graph that takes into account the topology of the network. Using this notion, we construct a dependence matrix of flows in each node that constitutes a base for selecting optimal configuration of transmissions in a given slot according to the chosen objective.

Then, using the above platform we implement several scheduling mechanisms. We start with single objective schemes such as scheduling that minimize delay using Start Time Fair Queuing (STFQ) algorithm and scheduling that maximizes the throughput. Then we formulate multi-objective functions that are used to achieve a trade-off between delay and throughput performance. Two concepts are considered. The first one is based on the Nash arbitration scheme from cooperative game theory where the objective is to maximize the product of two utility functions. The second one is based on maximization of the weighted sum of these utility functions.

The numerical results demonstrate the viability of the proposed scheduling platform. First we compare the delay and throughput performance of the scheduling scheme that utilizes Start Time Fair Queuing (STFQ) algorithm

with the suboptimal distributed scheduling algorithms presented in [2] and [3]. Then we analyse the performance of the multi-objective formulations that provide a trade-off between the delay and throughput objectives.

The paper is organized as follows: in Section 2, we present the related work. Section 3 describes the considered MUD based MAC framework. Our platform for implementation of scheduling with different objectives is described in Section 4. Section 5 provides description of different objectives implementation including STFQ based objective and Nash arbitration based multi-objective scheme. In Section 6, we present simulation based numerical results and their analysis. Finally, Section 7 concludes the paper.

2. RELATED WORK

The most relevant work to our scheduling platform concept is presented in [4] where the authors have developed distributed scheduling approach that achieves weighted fairness while trying to maximize the throughput for IEEE 802.11 based ad hoc network. Their approach is based on the notion of a flow contention graph that takes into account the topology of the network. They have also developed a topology independent model for fair queuing in [5]. In relation to wireless ad hoc networks, little work has been reported for fair packet scheduling. The most relevant work to packet scheduling is presented in [6] where the authors proposed Distributed Fair Scheduling (DFS) for IEEE 802.11 WLAN. Their mechanism was based on Self-Clocked Fair Queuing (SCFQ) scheme which improves fairness in WLAN. However, it woul be very difficult to implement directly this approach in wireless ad hoc networks.

3. CONSIDERED MUD BASED MAC FRAMEWORK

Single-channel MAC mechanisms (like IEEE 802.11) have difficulty with providing required QoS for multimedia services in multi-hop Ad Hoc networks due to the large and variable delays of packet transmissions. One possible direction to accommodate the multimedia services in Ad Hoc networks is to increase the spectrum reuse by using CDMA multi-channel transmissions. Two basic architectures can be considered: parallel user transmissions (PUT) and multiuser reception (MUR). CDMA MUR is widely used in commercial cellular systems [7] while CDMA PUT are studied in full in [8]. The three categories of MAC platforms are illustrated in Figure 1.
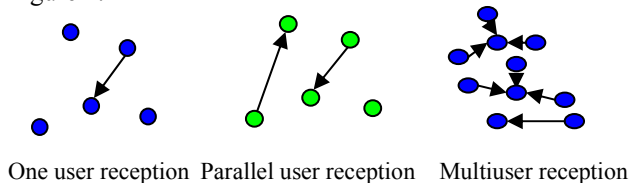

One user reception   Parallel user reception   Multiuser reception

Figure 1. MAC categories in wireless network.

The efficiency of MUR can be further increased by applying MUD [9]. In this case, the mutual interference of received signals is mitigated at the expense of increased complexity. Recent technological advances allow integration of a CDMA MUD based receiver on one chip. This development allows considering application of MUD for ad hoc networks in order to take advantage of both: spectrum reuse improvement due to MUR and capacity gain due to MUD. However, applications of MUD and MUR to ad hoc networks are not straightforward. In particular, the issue of MAC scheduling is quite challenging since the number of possible CDMA channel configurations in a node's neighborhood is large. Note that in general, an efficient solution to this problem requires a protocol that exchanges the necessary information among all the neighboring nodes without a large overhead in order to take advantage of CDMA spectrum reuse. A possible solution to this problem was proposed in [2] and [3]where a MAC mechanism based on synchronous frames was proposed as illustrated in Figure 2. In this approach the data frames are divided into a scheduling slot and a data transmission slot. In each signaling slot the information is first exchanged between the neighboring nodes and then the distributed scheduling decision is made based on a principle that priority is given to the voice packets that has the smallest timeout value (the time after which the packet is rejected). Since this principle is not necessary optimal and the operator may want to take into account other factors such as throughput maximization, in the next section we propose an approach for distributed scheduling in the presented MAC environment that can be used for testing and comparing different scheduling mechanisms with different objectives.
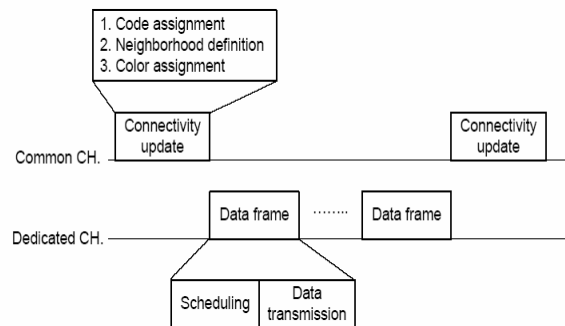

Figure 2. Synchronous frame structure.

4. DISTRIBUTED SCHEDULING PLATFORM

In this section we describe a scheduling approach that assumes that each node receives relevant information from all other nodes through the signaling stage prior to this stage (Note that in the scheduling protocol described in [2] and [3] only one hop exchange was taken into consideration.) The approach is comprised of three main components. First, the flow matrix and flow dependence matrix are constructed in each node. Second, the possible

2

configurations (also called cliques) of transmissions in the neighborhood are defined. Third, the best transmission configuration for the forthcoming data transmission slot is selected. In the following subsections we describe each of these components.

4.1 Construction of the Flow Matrix and Dependence Flow Matrix

4.1.1 Flow Matrix

Figure 3 displays the topology graph with arrows representing packets (also referred to as flows) selected for potential transmission in next data transmission slot. Each node selects at most one such packet according to the local scheduling. For this graph we can construct the flow matrix, $F = [f_{i,j}]$ for $1 \leq i,j \leq N$, where $f_{i,j}$ is a flow between nodes n_i and n_j, and N is the number of nodes. The entries of flow matrix are defined as follows:

$$f_{i,j} = \begin{cases} 0, & \text{if } i = j \text{ or nodes are outside the transmission range} \\ 1, & \text{if there are packets waiting for transmission from } n_i \text{ to } n_j \\ 2, & \text{if there are no packets waiting for transmission from } n_i \text{ to } n_j \end{cases}$$
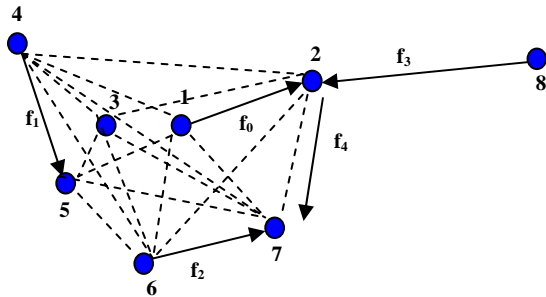

Figure 3. Flow graph.

The flow matrix corresponding to the flow graph from Figure 3 is represented in Table 1.

Table 1. Flow matrix

	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8
n_1	0	1	1	2	2	2	2	0
n_2	2	0	2	2	2	2	2	2
n_3	2	2	0	2	2	2	2	0
n_4	2	2	2	0	1	2	2	0
n_5	2	2	2	2	0	2	2	0
n_6	2	2	2	2	2	0	1	0
n_7	2	2	2	2	2	2	0	0
n_8	0	1	0	0	0	0	0	0

4.1.2 Flow Dependence Matrix

Let us define the flow dependence matrix $D = [d_{xy}]$ for $f_x, f_y \in M$ where M is the set of active flows ($f_x = f_{i,j} = 1$) and the entries are defined as follows:

$$d_{xy} = \begin{cases} 0, & \text{if } f_x \text{ and } f_y \text{ can be transmitted at the same time} \\ & \text{or } x = y \\ 1, & \text{if } f_x \text{ and } f_y \text{ cannot be transmitted at the same time} \end{cases}$$

Note that in our MAC environment, some pairs of flows cannot be transmitted at the same time due to half duplex operation of transceivers and not due to the channel contention. The flow dependence matrix corresponding to the flow graph from Figure 3 is given in Table 2.

Table 2 Flow dependence matrix

		$f_{1,2}$	$f_{4,5}$	$f_{6,7}$	$f_{8,2}$	$f_{2,7}$
		f_0	f_1	f_2	f_3	f_4
$f_{1,2}$	f_0	0	0	0	0	1
$f_{4,5}$	f_1	0	0	0	0	0
$f_{6,7}$	f_2	0	0	0	0	0
$f_{8,2}$	f_3	0	0	0	0	1
$f_{2,7}$	f_4	1	0	0	1	0

The flow dependence graph corresponding to the flow dependence matrix from Table 2 is depicted in Figure 4.
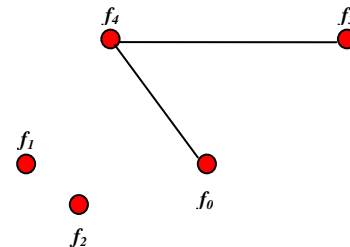

Figure 4. Flow dependence graph.

4.2 Construction of Cliques

Based on its flow dependence matrix each node can create several feasible flow configurations (also called cliques) that can be considered for transmission in the next data transmission slot. We assume that the feasible flow configuration, C_k, can exclude some flows only if these flows are in a conflict with some flows selected for this configuration. Then the algorithm to find the set of all feasible cliques (FC) is defined below:

Let us define the flow set, flow-set = M = {all active flows} and dependence set, dep-set = {flows are in a conflict with some flows} and n is number of active flows. We denote f_x is in conflict with f_y by $f_x \neq f_y$.

3

- The dep-set from flow dependence matrix is defined below:

  In the first, dep-set = { ϕ },

    For i = 1 to n

      For j =1 to n

        if $d_{xy} = 1$ then $f_x \neq f_y$

          dep-set = dep-set $\cap \{(f_x \neq f_y)\}$

  e.g. from table 2, dep-set = $\{(f_0 \neq f4), (f_3 \neq f4)\}$

- Create feasible cliques by creating all possible combinations from flow-set and dep-set, i.e. all possible flow sets witch the flows are not in a conflict.

For table 2 the above algorithm will create two feasible cliques: FC = $\{C1, C_2\}$, $C_1 = (f_0, f_1, f_2, f_3)$ and $C_2 = (f1, f_2, f_4)$.

4.3 Clique Selection

Once the feasible cliques are defined, each node selects the best one, according to the network operator objectives, for scheduling in the next data transmission slot. This selection can be achieved by using a metric that characterizes each clique with respect to the operator objectives. In general, this metric can be formulated as a utility function that can also take into account multi-objective formulations. For example, throughput maximization's objective can use the number of flows in the clique as a metric. On the other hand, priority fairness's objective can use sophisticated formulas for tag calculations from fair queuing domain and then characterize each clique by the most critical tag among the flows as illustrated in Figure 5. For multi-objective formulations, one can use a heuristics algorithm or apply an approach based on Game Theory where the metric is based on some fairness criteria.
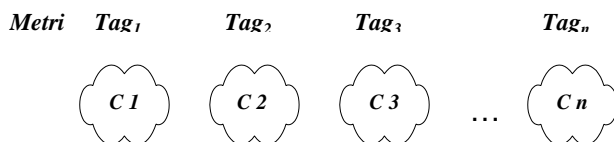

| *Metri* | *Tag₁* | *Tag₂* | *Tag₃* | | *Tagₙ* |

C 1   *C 2*   *C 3*   …   *C n*

Figure 5. Clique metrics.

In summary, the proposed approach allows to test and compare multitude of optimal and suboptimal scheduling algorithms with different objectives including multi-objective formulations.

4.4 Update flow dependence matrix

Whenever node n listens to a new service tag for any flow f_j from its D, it updates the entry for flow f_j and when node n transmits a head-of-line packet for flow "f", it updates flow "f" service tag in D and piggybacks the service tag in the handshake messages message of control (RTS, CTS, ACK).

5. SCHEDULING OBJECTIVES AND THEIR IMPLEMENTATIONS

This section describes the considered scheduling objectives and their implementations based on the platform described in the preceding section. Concerning the one objective formulations we present first the model based on STFQ (Subsection 1). Then the approach based on the timeout priority (TOP), that was used in [2,3], is presented in Subsection 2 followed by throughput maximization objective (Subsection 3). The considered multi-objective formulations include the Nash arbitration based scheme (Subsection 4) and the weighted sum approach (Subsection 5).

5.1 Start time Fair Queuing based scheduling (STFQ)

Many studies have been carried out on fair queuing algorithms to achieve a fair allocation of bandwidth on a shared link. Fair queuing algorithms in literature typically attempt to approximate the Generalized Processor Sharing (GPS) discipline [10]. STFQ, which is a GPS discipline is computationally efficient and allocates bandwidth fairly regardless of admission control and server variation. It schedules packets in the increasing order of start tags and uses two tags: a start tag denoted by $S(P^k_i)$, and a finish tag denoted by $F(P^k_i)$. These tags are associated with each packet, where i denotes the flow number and k denotes the round number. Our system assigns start tags using a virtual clock which plays the role of a "flow meter" driven by packet arrivals. According to the flow's specified average transmission rate, the difference between the Virtual-Clock and the real time clock will show how closely a running flow is following its claimed rate. Virtual Time $v(t)$ is defined as the start tag of the packet in service at time t.

- When a packet P^k_i arrives at time t, it is labeled with the start tag that is calculated as follow:

  a-In an active period: $S(P^k_i) = F(P^{k-1}_i)$

  b-In an inactive period: $S(P^k_i) = v(t)$

- Initially the Virtual Time of the server is zero. During an active period, the Virtual Time $v(t)$ at time t is defined to be the start tag of the packet in activity at time t, $v(t)=S(P^k_i)$. At the end of the busy period, virtual time is set to the maximum of finish tag assigned to any packets that have been serviced. Then, $v(t) = max^k_i F(P^k_i)$.

- The Finish time is computed as follows:

  $F(P^k_i) = S(P^k_i) + L_i / W_i$

where L_i is the packet size of flow i and W_i is its weight.

4

- Packets are serviced in the increasing order of the start tags and the ties are broken randomly

The implementation of the STFQ based scheduling is divided into two parts: tagging and scheduling. Tagging maintains a track of lead and lag in the amount of service each flow receives according to the algorithm presented in the previous subsection. Then, in each node the distributed scheduling mechanism, described in section 3, selects a clique that contains the smallest tag flow in order to preserve fair scheduling across all flows. The selected clique defines the function of the node (receiver or transmitter) and the packets (flows) to be sent in the next data transmission slot.

5.2 Time out priority scheduling (TOP')

The same metric as prescribed in [2] and [3] is used in our timeout priority (TOP) implementation. The distributed scheduling mechanism in each node selects a clique that contains the smallest values of timeout after which the packet is destroyed in order to preserve fair scheduling across all flows.

5.3 Throughput maximization scheduling (TM)

The maximum number of flows, T, in the clique is used as a metric in the objective of the throughput maximization. Then, the distributed scheduling mechanism selects a clique that contains maximum number of flows in each transmission slot.

5.4 Nash arbitration based multi-objective scheduling (NASH)

In the cooperative game theory framework the Nash arbitration solution [11], also called the NBS 'Nash Bargaining Solution provides a fair solution that balances the players satisfaction levels expressed by their utilities. This solution can be found by maximizing the product of players' utilities and we apply this concept to balance the network users' satisfaction from delay performance with the network operator satisfaction from the network throughput. In this case the inverse of the smallest start tag (from STFQ formulation) is used as a utility function for the delay satisfaction from each clique:

$$D' = 1/S(P^k_i)$$

Then we define the throughput satisfaction as the maximum number of flows, T, for each clique. Finally the clique with the maximum product of the utilities:

$$Max\ (T*D')$$

is selected for transmission.

5.5 Weighted sum based multi-objective scheduling (WSUM)

In this approach we use the same delay and throughput utility functions as defined for the NASH scheduling defined in the previous subsection. Nevertheless in this case, instead of maximizing the product, we select the clique that maximizes the weighted sum of these utilities:

$$Max\ (T + \alpha\ D')$$

This formulation can be seen as Thomson's solution [12] in the framework of cooperative game theory. The weight α can be used vary the relative importance of each utility [11].

6. ANALYSIS AND RESULTS

In this section, we compare and analyze the performance of the scheduling algorithms presented in the previous section (STFQ, TOP', TM, NASH, WSUM) and the scheduling proposed in [2] and [3] referred to as TOP..

The numerical results are obtained by means of a discrete event simulation that models an ad hoc network with parameters summarized in Table 3. Initially, the nodes are randomly distributed in the modeled circle area. Then, a mobility model is used which mimics human and vehicle movement behavior [13]. The voice packet arrival rate represents 20% of total arrival rate.

Table 3 Simulation parameters

Spreading gain	128
Modeled circle area	1000 m
The speed limit	50km/h
Maximum trans. power	7w
Types of traffic	voice and data
The number of nodes	60
Simulation time	100000 frames

The average voice packet delay and the total throughput (voice + data packets) are two performance metrics selected for analytical comparison. Throughput is defined as the total number of packets transmitted during the simulation or as the number of packets per frame per node. Average voice packet delay is defined as the average queuing delay in each node and is expressed in frame duration units (10ms). Figures 6 and 7 depict the performance characteristics as a function of packet generation rate factor, p, and MUD capacity that limits the number of simultaneously received CDMA signals and, therefore, also limits the number of transmission neighbors of each node.

Figure 6(a) compares the voice packet average queuing delays vs. offered traffic load factor, p, and Figure 6(b) presents the delays vs. MUD capacity M for $p=0.8$ that corresponds to a loaded network. Analogously Figure 7(a) compares the total throughput (expressed as the number of packets per frame per node) vs. varying traffic load p and Figure 7(b) presents throughput (expressed as the number of bits per second and per node)

5

vs. MUD capacity M for *p=0.8*. The results are shown for five scheduling schemes: TOP, TOP', STFQ, TM, and NASH, Here are the main conclusions from the result analysis:

a. The TOP' case gives a tangible but small improvement in voice delay when compared with the TOP model while the total throughput is comparable in both cases. This indicates that the neighborhood limited knowledge of the network state in the TOP model does not deteriorate significantly the performance.

b. Comparison of methods based on the platform presented in Section IV confirms the expected feature that there is a clear trade off between the voice delay and the total throughput performance. In other words improvement in delay metric is always achieved at the expense of throughput and vice versa.

c. STFQ model provides a significant improvement in the voice delay performance when compared with the TOP model especially for loaded network. In particular, for *p=0.8* the delay in the STFQ model was reduced by 11% while under overloaded case of *p=1.6* the delay reduction reached 47%. Although these gains are achieved at the expense of some moderate throughput reduction (~5%), these results indicate that the STFQ protects very well the voice traffic even under very heavy overloads which is not the case for other schemes.

d. As expected the TM model provides the best total throughput performance that for *p=0.8* exceeds TOP model by ~10% and STFQ model by ~14%. Nevertheless these gains are achieved at the expense of significant increase in the voice delay that is in the order of 12 % and 25 %, respectively.

e. As expected the multi-objective NASH model provides a compromise between the voice delay minimization and throughput maximization. It should be noted that when compared with TOP and TM this compromise is quite good since the relative gain in throughput over TOP exceeds corresponding increase of delay.
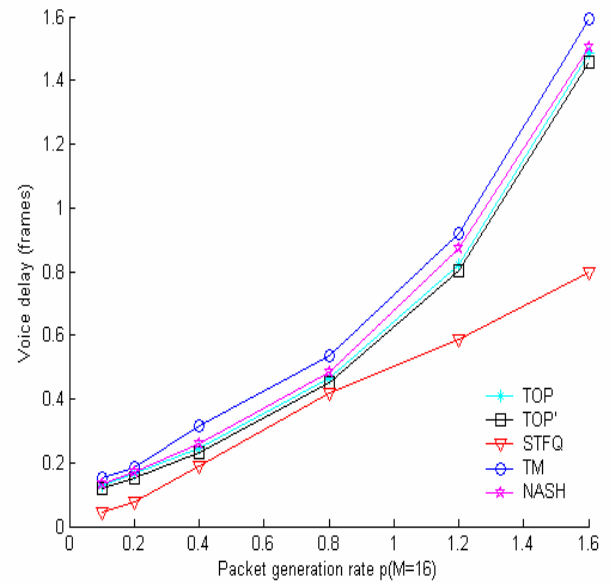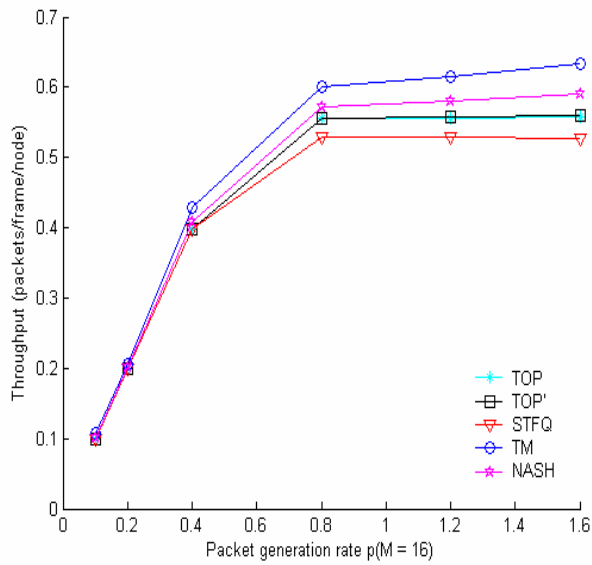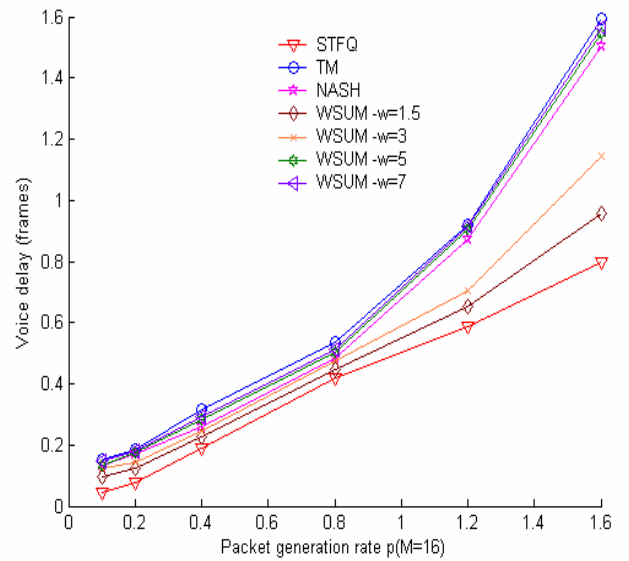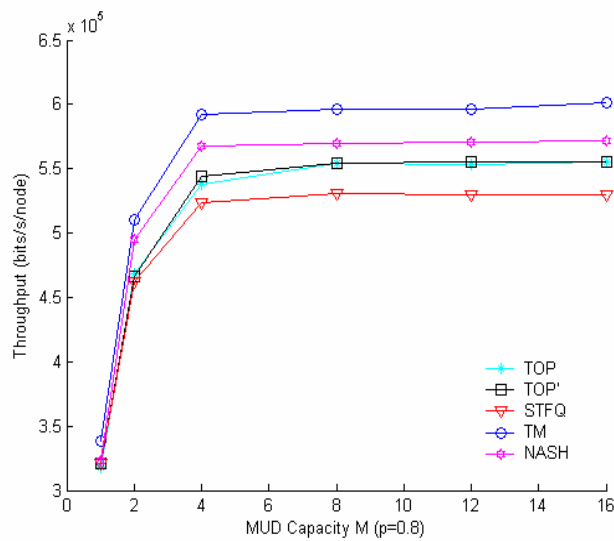

(a)
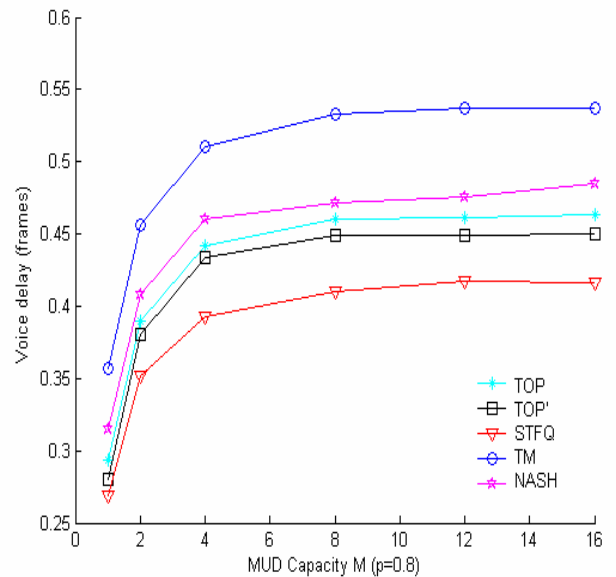
(b)

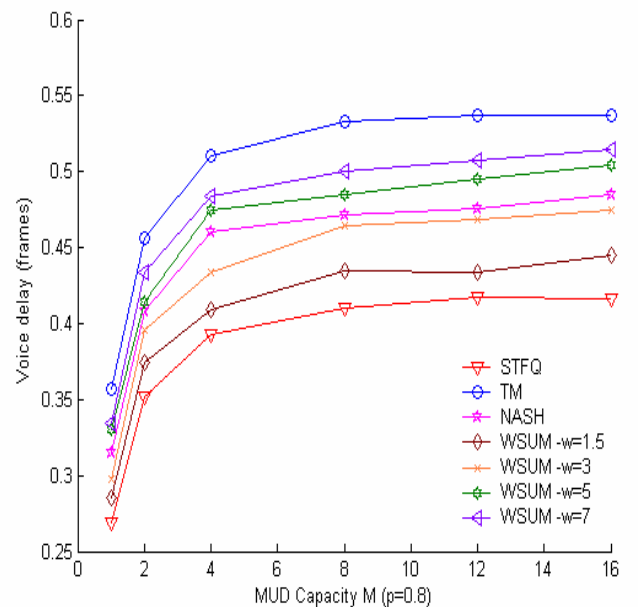Figure 6.   Voice packet delay for different scheduling models.

6

(a)

(b)

Figure 7. Throughputs for different scheduling models.

(a)

(b)

Figure 8. Voice packet delay for WSUM model.

The performance of the second multi-objective function, WSUM, is illustrated in Figures 8 (a & b) and 9 (a & b) for different weight parameters and compared with the relevant single objectives: STFQ, NASH and TM. The results indicate that the WSUM model gives the flexibility of achieving an arbitrary tradeoff between STFQ and TM. On the other hand the NASH solution avoids optimization of the weight value.

7
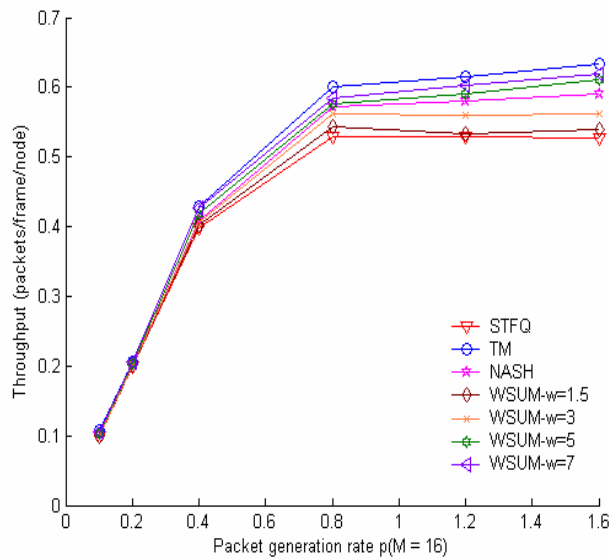
(a)
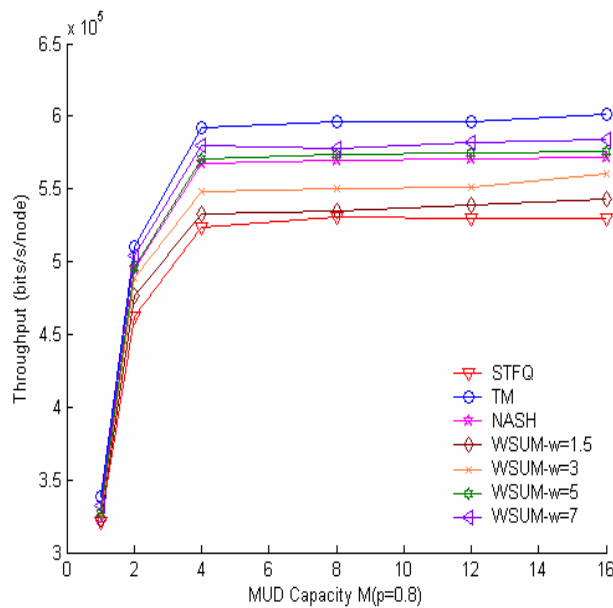
(b)

Figure 9. Throughputs for WSUM models.

7. CONCLUSIONS AND FUTURE WORKS

In this paper, we addressed the issue of scheduling optimization for wireless ad hoc networks with MUD based MAC. As shown in [2,3], MUD can give significant gains in throughput and QoS performance. Nevertheless, achieving these gains requires optimization of the distributed neighborhood scheduling in order to obtain the desired performance objectives. To achieve this goal, we proposed a platform that allows analyzing and comparing optimal or suboptimal distributed neighborhood scheduling schemes with different objectives. The approach is based on the flow and flow dependence matrices that are used to create a set of possible scheduling configurations (also refered to as cliques). Then, the selection of a configuration used for transmission is based on the chosen scheduling objective. To demonstrate the viability of this approach we implemented, using the proposed platform, several scheduling algorithms with different single objectives based on STFQ, throughput maximization and timeout priorities. Moreover we also proposed and implemented two multi-objective functions with the aim of providing a trade off between the delay minimization and traffic maximization objectives. One is based on Nash arbitration and the other on weighted sum of the respective utilities. The numerical results showed that the STFQ implementation can improve significantly the voice packets' average delay, especially under overloaded conditions, at the expense of some reduction in the total network throughput. The results for multi-objective scheduling schemes confirmed their ability to provide efficient compromise between the conflicting objectives.

Future work is undergoing in several directions. The proposed approach assumes that each node receives scheduling information from all other nodes, which is difficult to achieve in each frame. Therefore, we plan to augment the one hop signaling protocol developed in [2] and [3] to a two-hop protocol that should be sufficient for the proposed approach. At the same time, reduction of the scheduling signaling loads by exchanging information with larger cycle than one frame and only when needed is being considered.

8. REFERENCES

[1] S. Kumar Sarkar, T.G. Basavaraju, C. Puttamadappa. "Ad hoc Mobile Wireless Networks, Principles, protocols, and application, " Auerbach Publications edition, 2008.

[2] J. Zhang, Z. Dziong, F. Gagnon, M.Kadoch, "Multiuser Detection Based MAC Design for Ad Hoc Networks," accepted for publication in IEEE Transactions on Wireless Communications (May 5, 2008).

[3] J. Zhang, Z. Dziong, M. Kadoch, F. Gagnon, " Performance Evaluation of A Multiuser Detection Based MAC Design for Ad Hoc Networks, " Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th Volume, Issue, Sept. 30 2007-Oct. 3 2007 Page(s):294 − 298.

[4] L. H, Medvedev. P, Cheng. J, Lu. C,"A Self-Coordinating Localized Fair Queueing in Ad Hoc Wireless Networks, " IEEE Transactions on Mobile Computing, vol., vol. 3, NO. 1, January/March 2005.

[5] L. H, P, LU. S, "A topology–Independent Wireless Fair Queueing in Ad Hoc Wireless Networks," IEEE Journal on Selected AREAS In communications, Vol. 23. No 3, March 2005.

8

[6] N. Vaidya, A. Dugar, S. Gupta, and P. Bahl,"Distributed Fair Scheduling in a Wireless LAN," IEEE Transactions on Mobile Computing, vol. 4, NO. 6, November/December 2005.

[7] J. Zhang, J.M.Mark, and X. Shen, "An adaptive handoff priority scheme for wireless MC-CDMA cellular networks supporting realtime multimedia applications," Computer Communications, vol. 28, pp. 1240–1250, 2005.

[8] K. T. Jin and D. H. Cho, "Multi-code MAC for multi-hop wireless Ad Hoc networks," in Vehic. Tech. Conf. IEEE, vol. 2, pp. 1100–1104, 2002.

[9] Z. Xie, R. T. Short, and C. K. Rushforth, "A family of sub optimum detectors for coherent multi-user communications," IEEE Journal on Selected Areas in Communications, vol. 8, pp. 683–690, May 1990.

[10] A. K. Parekh, R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the single-node case," IEEE/ACM Transactions on Networking (TON), vol. 1, no. 3, pp. 344-357, 1993.

[11] J. Nash, "Two-person cooperative games," Econometrica, vol. 21, pp. 128–140, January 1953.

[12] Z. Dziong, ''ATM Network Resource Management, Appendix C, Cooperative Game Theory'', 1997 McGraw-Hill.

[13] J. Zhang and J. M. Mark and X. Shen,"An adaptive resource reservation strategy for handoff in wireless cellular CDMA networks", Can. J. Electr.Comput. Eng., vol. 29, no. 1/2, pp. 77–83, Jan./Apr. 2004.