

D-LSMA: Distributed Link Scheduling Multiple Access Protocol for QoS in Ad-hoc Networks

Zhibin Wu and Dipankar Raychaudhuri

WINLAB, Electrical and Computer Engineering Department, Rutgers University
Piscataway, NJ 08854, USA

Abstract—This paper presents a novel medium access control (MAC) protocol for QoS support in multi-hop ad-hoc wireless networks. The proposed D-LSMA (distributed link scheduling multiple access) protocol uses an extension of the 802.11 CSMA/CA procedure as the basis for a distributed link scheduling algorithm which results in dynamic TDMA-like bandwidth allocation among neighboring wireless nodes without the need for global synchronization. In addition to supporting QoS, the proposed scheduling technique also solves the “exposed node” problem in ad-hoc 802.11, thus resulting in improved throughput in many scenarios. Simulation results from an ns-2 model are presented for a 15-node random ad-hoc network. The results demonstrate significant performance improvements relative to ad-hoc 802.11, with capacity increases typically ~20% for the example considered. Also, the D-LSMA network is shown to offer far better real-time packet delay and fairness properties than 802.11, particularly under overload and heavy contention conditions.

Index Terms — Ad-hoc networks, MAC, 802.11, D-LSMA

I. INTRODUCTION

In wireless ad-hoc networks, an important open problem is that of efficiently supporting quality-of-service (QoS) for real-time flows such as voice, video or aggregated infrastructure traffic. This problem has become increasingly relevant with the recent use of ad-hoc mesh networks for applications such as community networking or in-home distribution of digital media. Unlike best effort data, real-time flows and aggregated traffic between wireless routers need relatively deterministic bandwidths and bounded delays, usually requiring some combination of reservation, priority control, scheduling and dynamic resource management. However, supporting real-time traffic in the mobile ad-hoc network environment is difficult because of the lack of a centralized infrastructure, the shared radio medium and potentially rapid changes in topology. Ad-hoc network QoS can be achieved to some extent through a combination of techniques at the medium access control (MAC) and network (routing) layers. In this work, we focus on the MAC layer and propose a solution for handling real-time traffic in a shared-medium ad-hoc networking environment.

The IEEE 802.11 CSMA/CA protocol[1] has become the de-facto MAC standard for the current generation ad-hoc networks. Clearly 802.11 MAC, while generally available and simple to use, has serious deficiencies in a multi-hop environment in terms of throughput, delay variance and fairness (see for example,[2]). Moreover, 802.11’s ad-hoc mode does not provide features necessary to support real-time streams with

some level of QoS control. The 802.11 PCF (point coordination function) and the more recent 802.11e enhanced DCF[3] are both attempts to provide limited QoS control in WLAN’s operated in the infrastructure mode. However, these solutions do not apply directly to the ad-hoc 802.11 scenario, motivating alternative methods. Many of the proposed medium access control procedures for QoS in ad-hoc networks [4][5][6][7] are based on explicit scheduling of traffic in time and space using techniques such as time division multiple access (TDMA) in which data slots are allocated on a reservation basis. Such dynamic TDMA methods involve global synchronization and reservation procedures which cannot generally be implemented using existing 802.11 radios as the base platform.

These considerations motivate investigation of distributed 802.11 MAC protocol extensions that can support flow QoS in multi-hop ad-hoc networks. The distributed link scheduling multiple access (D-LSMA) protocol described here extends 802.11 with a flexible distributed scheduling mechanism that permits nodes to reserve periodic time slices for real-time flows, i.e. each node can request a specified flow bandwidth on links to neighboring nodes. The proposed scheme’s distributed scheduling algorithm also eliminates the exposed node problem in 802.11 by permitting parallel scheduling of traffic on links which do not interfere with each other. In contrast to other distributed schemes such as those in[8][9], scheduling in D-LSMA is loosely synchronized through channel events rather than with more rigid TDMA framing methods.

The D-LSMA protocol has some similarities with the recently proposed MACA-P scheme [10]. In MACA-P, each sender-receiver pair uses the initial RTS/CTS exchange to establish a future reference time instant at which the DATA and ACK phases will commence. This enables neighboring nodes to proceed with their RTS/CTS exchanges and synchronize their own DATA and ACK phases respectively with the already established schedule. The proposed D-LSMA scheme also allows parallel transmissions on multiple sender-receiver pairs. However, those transmissions use overlapped schedules rather than synchronized ones. Thus, each node has more flexibility to reserve and can avoid being locked by preceding neighbor schedules.

DCMA (Data-driven Cut-through Multiple Access) [11] also provides improved performance over 802.11 in ad-hoc network environments. DCMA sets up a “cut-through” path for end-to-end flows. It introduces a combined ACK/RTS frame to preemptively reserve resources for next-hop nodes for a flow. This helps to reduce intra-flow contention and generates more

Supported in part by NSF NRT ORBIT Testbed Project, ANI-0335244 and by a grant from Cisco University Programs, 2003

deterministic schedules for the flow. In contrast to DCMA, D-LSMA scheme provides an option for each node to set-up deterministic schedules with neighbors, but leaves end-to-end flow reservations to a different protocol.

The rest of paper is organized as follows. Section 2 describes the framework of “scheduler + basic MAC”. The basic principles of D-LSMA are described in section 3. Section 4 presents initial simulation results with comparison of 802.11 and other protocols. Finally, we present our conclusion in section 5.

II. D-LSMA MAC MODEL

The 802.11 CSMA/CA protocol has evolved from pure contention to a “first reserve-then send approach” using the request-to-send (RTS), clear-to-send (CTS) mechanism in the current standard. The use of the RTS/CTS procedure with its “network allocation vector (NAV)” introduces the ability to schedule longer packets in a contention-free manner, and also helps to solve the “hidden node” problem by distributing the NAV to the vicinity of both transmitter and receiver. In the proposed D-LSMA protocol, we further generalize scheduling in 802.11 channels by introducing the concept of segregation between an “upper MAC” and “lower MAC”. The lower MAC in D-LSMA is similar to that in the 802.11 standard with extensions to the control syntax necessary to support more general scheduling policies. The upper MAC can be thought of as an intelligent scheduler which monitors flows to neighboring nodes and makes decisions on “when to reserve?” and “how much to reserve?” depending on traffic volume and flow QoS requirements at that node. The upper MAC can incorporate algorithms that are aware of network topology and traffic load in order to maintain flow QoS. As shown in Fig 1, the upper MAC can be designed to support a mix of traffic types with separate packet queues and different scheduling policies. Note that this type of segregation by traffic type is an important feature in multi-hop ad-hoc networks where radio nodes have to handle both local and cross traffic with different service/bandwidth requirements.

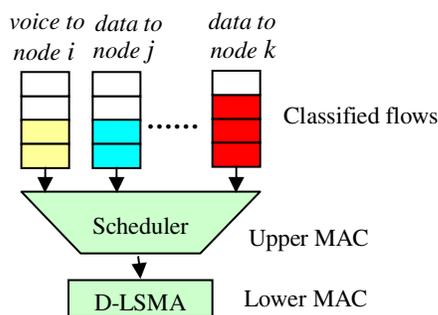


Fig. 1. New MAC structure: Scheduler + basic MAC

In the rest of this paper, we describe a specific realization of D-LSMA aimed at supporting QoS for real time traffic flows. It is noted here that the separation of scheduling from basic channel access means that the protocol described can be extended to other service objectives without changing the basic

architecture. The specific lower MAC protocol and syntax assumptions used in this work are as follows:

- Standard 802.11 access protocol with DCF and RTS/CTS functions.
- RTS and CTS packet payloads are extended to include more general forms of scheduling beyond the single-packet reservation in 802.11.
- NAV is reinterpreted to include only the RTS/CTS period, but does not cover the subsequent packet transmission.
- Reservation information carried in all RTS/CTS packets are aggressively collected.
- Actual transmission of packets is controlled by the D-LSMA upper MAC to be described in the next section.

III. D-LSMA PROTOCOL

A. Link Scheduling and Exposed Terminal

The technical challenge of “link scheduling” in a multi-hop radio network is described in [12]. The problem is that of constructing a TDM (Time Division Multiplexing) schedule for unicast communications in a multi-hop radio network. From a centralized perspective, this is equivalent to solving an “edge coloring” problem in directed graph. The underlying idea is that two links could be assigned the same TDM slot as long as they are not adjacent and there is no third link from the transmitters of one link to the receivers of another link.

This rule provides parallel communication opportunities to be utilized in a multi-hop radio networks. However, when the IEEE 802.11 standard MAC is used, efficiency of parallel communication is limited by the well-known “exposed node” problem. Our solution to this is to modify 802.11 procedures to eliminate the ACK frame when transmitting real-time data, thus making it possible for exposed nodes to transmit in parallel. Omitting link-level retransmission is also helpful for real-time applications because retransmissions generally increase delay variance for received packets and may not provide an improvement in throughput relative to end-to-end error control alternatives.

In [12], each node may be assigned TDMA slots in the schedule table for each directed link to a neighbor node using a centralized algorithm with full knowledge of network topology. The D-LSMA, however, uses a distributed approach to form such a schedule.

B. D-LSMA Principle

The D-LSMA MAC has three main components: schedule table (ST), scheduler, and lower MAC. ST is generated by mapping information from 3 schedule classes (SC) according to the previously described link scheduling rule. It marks time slices with potential parallel communication opportunities. The scheduler (upper MAC) refers to the ST and sends a “reserve” command through CI (command interface) to the lower MAC which is responsible for sending out the RTS request. After it receives a CTS response from the receiver, the DATA frames will be passed down to the lower MAC for transmission at reserved time intervals as illustrated in Fig 2 below.

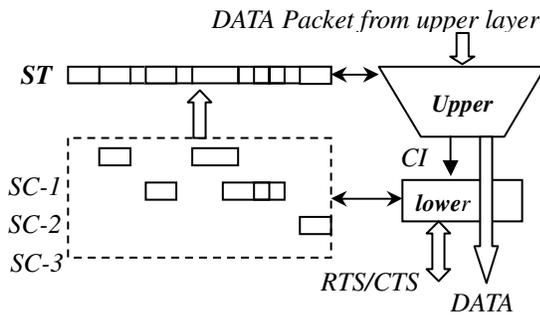


Fig. 2. D-LSMA Architecture

The ST divides the time axis into consecutive time slices, and each slice is marked with one of five “category tags”:

- *Free*. Open to make reservations on this period or use this time slice to send reservation requests.
- *Reserved*. Neither further TX nor RX is allowed
- *Transmit OK*. Node may transmit to nodes except to those in the “active transmitter list”. This list is formed by recording the senders of all the messages heard in neighborhood.
- *Receive OK*. Node may receive packets.
- *Semi-Reserved*. The node itself wants to reserve this slice, but has not been acknowledged by a complete RTS/CTS exchange. After that, it turns into “reserved”.

To label each time instant correctly, the D-LSMA MAC maps information from 3 classes of SC.

- SC-1 keeps schedules related to the node’s own activity. These should be mapped to “reserved” category.
- SC-2 maintains schedules of possible indirect activities of neighbors. Those schedules might overlap because of parallel communication and should be mapped to “Transmit OK” or “Receive OK” respectively.
- SC-3 records the nodes own schedule corresponding to a “semi-reserved” status.

After performing the above mappings from SCs, the remaining idle time slices are all marked “free”.

C. Distributed Algorithm for Link Scheduling

In a distributed manner, each node makes its own SCs and ST by collecting reservation information aggressively from the channel messages. The lower MAC extracts all schedule information from every heard RTS or CTS, no matter whether it is a normal reception (destined to the node’s own address) or an overheard one (destined to others). Whenever schedule information embedded in RTS or CTS is got, SCs and ST are updated. Outdated or invalid information are purged. It is possible that some RTS and CTS messages provide information contradictory to each other, or conflict with current schedules in the SC or ST. Thus, a major challenge is to set rules for proper schedule update. Here, we adopt a conservative policy to resolve conflicts based on following rules:

- CTS is taken to be more trustworthy than RTS. Information carried in CTS overrides that of RTS.
- Fresh information from a node overrides previous information provided by the same node and overrides

unacknowledged RTS request to that node.

- Schedules in SC-1 overrides that of SC-2. Both SC-1 and SC-2 override SC-3.

The above rules are complementary and successfully eliminate the ambiguity of schedules and could help to discover an implicit reject of RTS.

Fig 3 shows the state machine diagram of lower MAC. State transition are triggered by three kinds of events: timer, RTS/CTS messages or command from upper MAC.

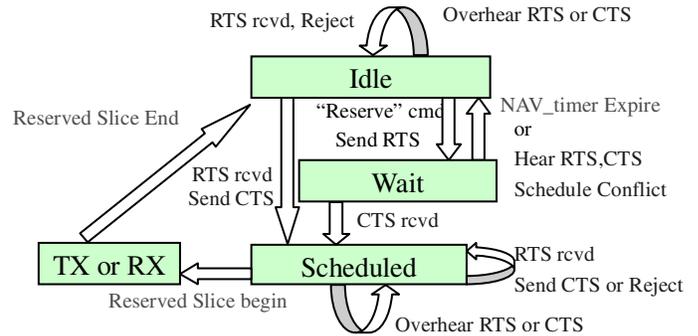


Fig. 3. State Machine Diagram of Lower MAC

From the figure, it can be seen that DATA transmission and reception are strictly delimited by reservation timers. This would help to satisfy the flow specification of real-time flows. And after a successful RTS/CTS exchange, there is an additional “scheduled” state before the actual data transmission. In this state, the node is still free to arrange additional parallel transmissions and handle RTS or CTS messages, as in the “idle” state. The only difference is that no more RTS requests will be sent in the “scheduled” state. This is a conservative way to avoid too many reservations from one node and encourage fair bandwidth sharing among neighborhood.

The scheduling algorithm in the upper MAC is responsible for appropriate downlink scheduling decisions to support QoS objectives. DATA packets are stored in different queues according to different flow specifications and destinations. For this preliminary work, we design a scheduler which chooses one available flow according to the available “Transmit OK” and “Free” schedules in ST, and the flows are served in a round-robin manner.

D. Implementation: Extending from IEEE 802.11

1) *Timing Synchronization*: In the 802.11 frame format, only relative timing information is included in the NAV field. In D-LSMA, packet transmission is reserved at a future time which is separated from the RTS. Similar to 802.11, the preferred transmit time is also relative to the RTS packet. This timing offset is carried in RTS frame. The only difference is that in 802.11, this value is a fixed value. In D-LSMA, however, this is a value determined by upper scheduler. As each node only describes its reservation requests to 1-hop neighbors, nodes that cannot hear each other do not need to have uniform timing descriptions. Thus, this local synchronization scheme is introduced to align neighboring nodes.

Each node holds a “timing offset” table to integrate schedule information from different neighbors and map them into the ST. The propagation delay issue can be solved as same as in 802.11 standard.

2) *Extending Reservation in RTS*: In D-LSMA, there is considerable flexibility in making reservation requests. Node may reserve in the same way as 802.11, or reserve back-to-back transmission for multiple packets, or reserve periodic time slots for packets from same CBR flow. The timeline of reservation procedure is depicted in Fig 4 (for simplicity, small time intervals like SIFS are not depicted).

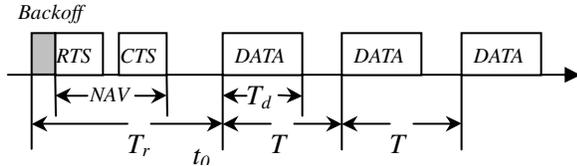


Fig. 4. Timing Relationships for Reservation

As seen in the figure, the following parameters are involved:

- t_0 . Starting time for a D-LSMA transmission procedure..
- T_d . Time needed for data transmission.
- N . Number of periodic time slices to reserve.
- T . Cycle time of a periodic packet flow if $N > 1$.
- T_r . Time between reservation and first data transmission.

A modified RTS frame which includes above parameters will be sent after contention with backoff procedure. NAV is adjusted to duration only enough for RTS/CTS exchange. By exchanging RTS/CTS, the sending node and receiving node will reach an agreement that the sender will use the time slice as $[t_0 + iT, t_0 + T_d + iT], i = 1, 2, \dots, N - 1$ for transmission with a successful reservation. The scheduler is responsible for choosing appropriate above parameters. In the preliminary simulation results given in the next section, a fixed time offset of $T_r = 1ms$ is set when the scheduler makes reserve decisions and the nearest available feasible transmission time after this offset in ST is chosen as t_0 . Note that, for every possible retransmission of RTS, the t_0 has to be recalculated.

E. D-LSMA Example

Here is an example in Fig 5 which shows a network with 5 nodes, A, B, C, D and E. Nodes A and C have some traffic to B, while D has some traffic to C and E each. The procedure followed by the nodes to establish transmission schedules for periodic flows to neighbors is outlined below:

- 1) At the beginning (t_0), nodes A, C and D all want to make reservation requests for each of their traffic flows based on the desired bandwidth (or time slice referenced to a known periodic interval, T). Since the nodes still do not have information about other traffic in the network, they first enter their own request into their schedule table as “Semi-Reserved” shown at the bottom of the figure.
- 2) Following 802.11 carrier sense, collision and backoff procedures, Node A and D capture the carrier and successfully reserve channel with B and C respectively. So,

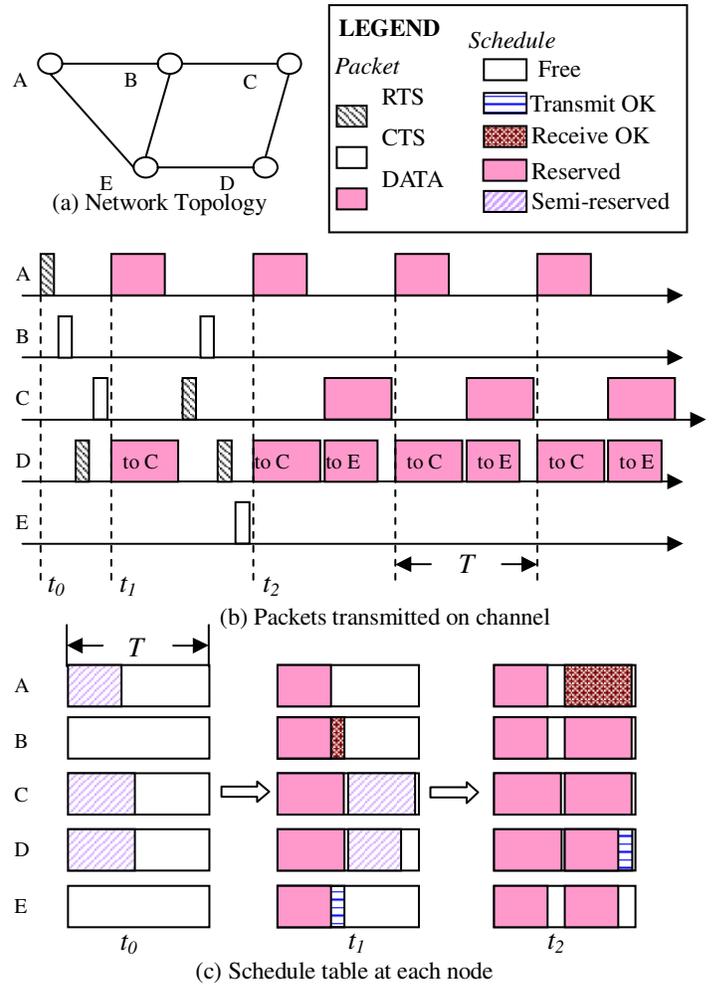


Fig. 5. Diagram for a D-LSMA Example

at time t_1 , the schedule table looks like that illustrated in part (c) of Fig 5.

- 3) Nodes A and D transmit their data packets according to the established schedule. After that, nodes C and D want to reserve their respective flows again. In this case, node C gets the channel and sends the RTS first. Based on link scheduling rules, although node D hears node C's scheduled transmission to B, it finds that this does not affect its own transmission to E. Thus, D reserves a slice through RTS/CTS to E. The resulting schedule table at time t_2 is also shown in Fig 5. Once this schedule is established, the desired CBR flows are efficiently supported by the network.

IV. SIMULATION EXPERIMENTS

In this section, we present detailed performance results for the proposed D-LSMA access protocol using an *ns-2* simulator. In our simulation model, the data rate is 1Mbps (basic rate) and the effective transmission range is 250 meters. Each source generates constant CBR traffic flows. Note that all reported results are averages over 100-second simulations and all nodes are stationary during the simulation.

First, we consider a test scenario of 15 nodes randomly

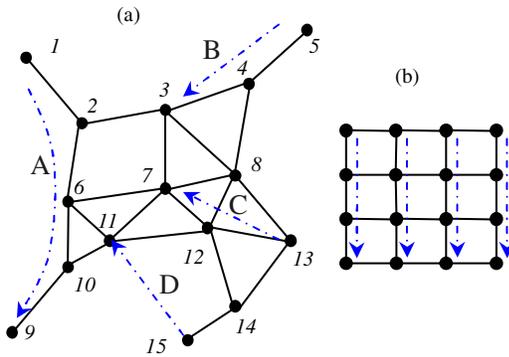


Fig. 6. Simulation Topologies

distributed in a 800×800 square meters area. The topology is depicted in the graph (a) of Fig 6. Four pairs of transmitting and receiving nodes are randomly chosen. The flows are labeled from A to D in that graph. Fixed routing paths are pre-specified for each flow as in the Table I. Three protocol options are considered for this scenario: 802.11, MACA-P and D-LSMA.

Flow	Hop Count	Path
A	4	1-2-6-10-9
B	2	5-4-3
C	2	13-8-7
D	3	15-14-12-11

TABLE I
RESPECTIVE FLOWS IN RANDOM TOPOLOGY

A. Comparison with 802.11

The simulation is conducted by supplying the same CBR packet rate for each flow. The packet size is 512 bytes. Fig 7 shows the throughput and delay values of each flow for both 802.11 and D-LSMA when the data rate for each is 110 Kbps (equivalent to 27.5 packets per sec.). It is obvious that the contention behaviors among different flows, especially between flow C and flow D cause throughput reduction and high end-to-end delay in 802.11. In D-LSMA, the performance is significantly improved both in terms of throughput-delay and fairness. Note that in 802.11, flow C has the least throughput because both node 7 and node 8 of this flow have to contend with their 5 neighbors. But D-LSMA protocol provides extra chances for those nodes to send or receive packets, thereby improving the throughput of flows C and D. The end-to-end delays in D-LSMA are all maintained at a low level (< 0.037 second) as desired for CBR flows.

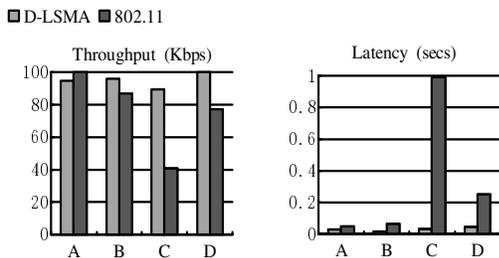


Fig. 7. Comparing Flows in Random Topology

Fig 8 shows the throughput gain measured as the gross throughput of all flows for 802.11 MACA-P and D-LSMA. In this test scenario, D-LSMA appears to provide ~ 20 -23% more throughput when network goes into saturation with increasing offered load.

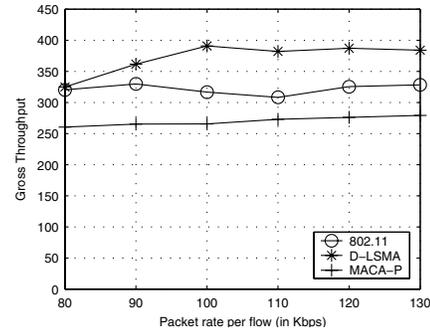


Fig. 8. Throughput Results for Random Topology

B. Comparison with MACA-P

In the above simulation, MACA-P protocol shows a saturated throughput less than that of 802.11. This is because if each node always tries to synchronize its schedule with neighborhood events, it may restrict its own capability to deliver packets in certain adversary circumstances [14]. Thus, certain flows get throttled while other flows take the advantage of synchronized parallelism in the MACA-P case.

C. Fairness

Fig 9 plots the changes of fairness index¹ with increasing offered load for three MAC schemes.

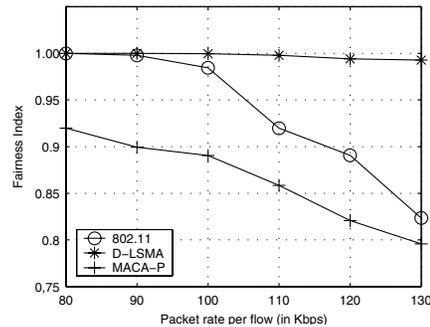


Fig. 9. Comparing Fairness Index for Random Topology

From Fig 7, we can see that 802.11 has a tendency to favor certain low contention flows and starve other heavy-contention flows. However, because nodes with D-LSMA have more efficient packet transmission schedules, flows in heavy contention neighborhoods can be better served. Moreover, in D-LSMA, as nodes with multiple flows will reserve on a round robin basis, the reservation tends to be fairer. Those factors lead to some improvement in fairness. Note that the fairness issue is closely related to scheduler design and can

¹In [13], Given n flow throughputs x_1, x_2, \dots, x_n , the fairness index is $(\sum_{i=0}^n x_i)^2 / (n \sum_{i=0}^n x_i^2)$

be further improved. Even for the simple scheduler used here, the improvement in fairness with D-LSMA is significant.

D. Simulation with Asymmetric Traffic

The above simulations were carried out with equal flows. In order to see the performance with asymmetric traffic, which is more realistic in practice, we divided four flows into two distinctive groups. Flow A and C are grouped in group I, Flow B and D are in group II. Flows within each group have an equal portion of offered load, while the total load is divided in Group I and II with different distribution percentage ranging from 0-100 to 20-80 and 40-60 and 50-50. With overall offered load as a constant (480Kbps), we measured the total throughput of all flows with different portions from each group. Note that when Group I presents 100% of load, it is a typical "hidden terminal" scenario where two flows in Group I contend with each other. When Group II presents 100% of load, there are two separate multi-hop flows and there are no contentions between them. The results in Fig 10 show that throughput increases when contention is reduced as might be expected independent of the MAC protocol used. For all load distribution between Groups I and II, D-LSMA provides better throughput than 802.11 MAC. And especially when contention is heavy (i.e., in the left half of the figure), D-LSMA outperforms 802.11 MAC by a significant margin.

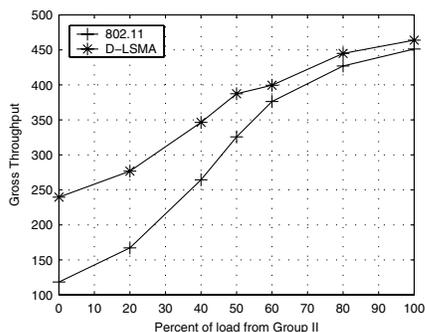


Fig. 10. Comparing Throughput with Asymmetric Traffic Load

E. Comparison with DCMA

Finally, we provide a brief comparison of D-LSMA with the DCMA scheme. The same 4 by 4 grid topology in [11] has been simulated with D-LSMA. As depicted in graph (b) of Fig 6, four greedy nodes in first row send vertical streams of 1024 byte sized UDP packets to the nodes in last row. Flows are labeled as 1 to 4 from leftmost column to rightmost.

The performance comparison in Fig 11 shows that D-LSMA has higher throughput and lower delay in the two middle columns. Basically, this is due to the fact that nodes in D-LSMA scheme get extra chances to transmit even when carrier sense is busy, so middle nodes perform better and flows are not as starved as in DCMA. While DCMA introduces a cut-through method to reduce contention and expedites forwarding in the same flow, D-LSMA explores the packet forwarding opportunities from multi-flow diversity. Both these schemes indicate a potential for significant improvements in ad-hoc

■ DCMA ■ D-LSMA

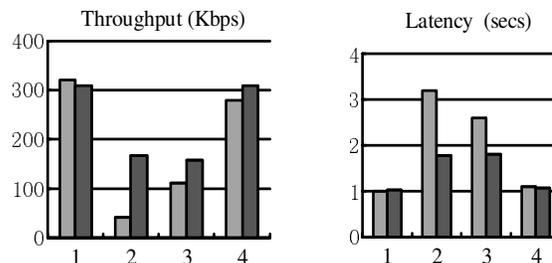


Fig. 11. Comparing D-LSMA with DCMA

network MAC performance with suitable extensions to widely available 802.11 radios.

V. CONCLUSIONS

D-LSMA, a novel 802.11-compatible medium access control protocol for real-time flows in ad-hoc networks, was presented in this paper. The protocol enables nodes to form a distributed packet transmission schedule that provides deterministic service for constant bit-rate flows while taking into account transmission opportunities at "exposed nodes". The performance of D-LSMA was evaluated using an ns-2 simulation model and compared with 802.11 and DCMA. The results show that D-LSMA achieves throughput gains of up to 23% when compared with IEEE 802.11, while providing bounded delay and improved fairness. The performance was also found to be competitive with DCMA and MACA-P, two other recent proposals for improving 802.11 performance in multi-hop scenarios. For future work, we intend to carry out a more complete comparative evaluation of D-LSMA and alternative protocols such as DCMA for a variety of realistic traffic models. A proof-of-concept prototype implementation on an 802.11/Linux ad-hoc network testbed is also planned.

REFERENCES

- [1] <http://standards.ieee.org/getieee802/802.11.html>
- [2] Hung-Yun Hsieh and Sivakumar, R., *IEEE 802.11 over multi-hop wireless networks: problems and new perspectives*. VTC 2002-Fall. pp.748-752
- [3] D. Gu and J. Zhang, *QoS enhancement in IEEE802.11 wireless local area Networks*. IEEE Comm. Magazine, June, 2003, pp.120-124
- [4] T-W. Chen, et al., *QoS routing performance in multihop, multimedia, wireless networks*. In Proceedings of IEEE ICUPC'97, 1997.
- [5] A. Michail et al., *Algorithms for routing session traffic in wireless ad-hoc networks with energy and bandwidth limitations*. In PIMRC 2001
- [6] Chenxi Zhu and M. Scott Corson, *QoS routing for mobile ad hoc networks*. In Proc. IEEE Infocom, June 2001.
- [7] Chunhung Richard Lin, *On-demand QoS routing in multihop mobile networks*. In Proc. IEEE Infocom, April 2001.
- [8] C. X. Zhu and M. S. Corson, *A five-phase reservation protocol for mobile ad-hoc networks*, in Proc. IEEE INFOCOM, 1998 pp.322-331
- [9] S. Jiang, et al., *A simple distributed PRMA for MANETs*. Vehicular Technology, IEEE Trans. on, Vol. 51 Iss. 2. pp.293-305, Mar. 2002
- [10] Acharya, A. et al., *MACA-P: a MAC for concurrent transmissions in multi-hop wireless networks*. in Proc. IEEE PerCom 2003, pp.505-508
- [11] Acharya, A. et al., *Mobile Ad Hoc Networks: A label-switching packet forwarding architecture for multi-hop wireless LANs*. Proc. of the 5th ACM inter. workshop on Wireless mobile multimedia, pp.33-40 2002
- [12] Ramanathan, S. et al., *Scheduling algorithms for multihop radio networks*. IEEE/ACM Trans. on Networking, Vol:1(2) pp.166-177, Apr. 1993
- [13] L.L. Peterson and B.S. Davie, *Computer Networks, A System Approach*, Second Edition, 2000
- [14] A. Acharya, et al., *Design and Analysis of a Cooperative Medium Access scheme for High-Performance Wireless Mesh Networks*. Broadnets, 2004