TMAC: Timestamp-ordered MAC for CSMA/CA Wireless Mesh Networks

Faisal Nawab*, Kamran Jamshaid*, Basem Shihada*, and Pin-Han Ho†
* MCSE Division, KAUST, Saudi Arabia, {faisal.nawab, kamran.jamshaid, basem.shihada}@kaust.edu.sa
† Dept. of Electrical and Computer Engineering, University of Waterloo, Canada, pinhan@bbcr.uwaterloo.ca

***Abstract*—We propose TMAC, a timestamp-ordered MAC protocol for Wireless Mesh Networks (WMNs). TMAC extends CSMA/CA by scheduling data packets based on their age. Prior to transmitting a data packet, a transmitter broadcasts a request control message appended with a timestamp to a selected list of neighbors. It can proceed with the transmission only if it receives a sufficient number of grant control messages from these neighbors. A grant message indicates that the associated data packet has the lowest timestamp of all the packets pending transmission at the local transmit queue. We demonstrate that a loose ordering of timestamps among neighboring nodes is sufficient for enforcing local fairness, subsequently leading to flow rate fairness in a multi-hop WMN. We show that TMAC can be implemented using the control frames in IEEE 802.11 stack, and thus can be easily integrated in existing 802.11-based WMNs. Our simulation results show that TMAC achieves excellent resource allocation fairness while maintaining over 90% of maximum link capacity in parking lot and large grid topologies.**

***Index Terms*—Wireless Mesh Networks, 802.11, TCP, fairness**

I. Introduction

Wireless Mesh Networks (WMNs) have been proposed as a low-cost alternative for last mile access [2]. These dynamic, multi-hop networks can be built with commodity hardware (including off-the-shelf 802.11 radios) and open source software. A typical WMN is composed of distributed Mesh Points (MPs) that form a multi-hop backhaul. MPs may connect with multiple other MPs within their radio range. Some MPs have a wired back-channel to the public Internet; these act as gateway nodes and bridge traffic between the WMN and the Internet. End-users often communicate with their closest MP to access the Internet.

Recent research shows that 802.11-based WMNs have serious fairness challenges [4], [7], [9]. With backlogged TCP traffic, MPs which are two or more hops away from the gateway starve, while the one-hop MP saturates the channel with its own local traffic. The underlying cause of this starvation is information asymmetry [4]. This happens when transmitters outside the carrier sense range may inopportunely schedule their transmissions such that one of the receivers experiences collisions. With backlogged flows, a disadvantaged receiver experiences repeated collisions which builds-up the contention window for the corresponding transmitter, resulting in flow rate unfairness.

In this paper we improve flow rate fairness by proposing a new MAC scheduling protocol, called Timestamp-ordered MAC (TMAC). TMAC addresses the fairness and throughput degradation in WMNs using the age of a packet as a metric for prioritizing its scheduling. TMAC is based on the mutual exclusion algorithm of Lamport [10]. Lamport algorithm uses request timestamps to ensure that the node with the earliest request is the node that will be served next. The algorithm relies on an explicit exchange of control messages to make all nodes aware of the network state. These communication requirements are more suited for fully-connected wired networks, but may scale poorly in large WMNs. TMAC addresses these challenges by limiting the exchange of these control messages to a set of neighboring nodes that contend for channel access. It improves fairness by prioritizing the transmission of packets that are generated before others (i.e., have a larger age). We show that for backlogged TCP flows, scheduling packets according to their age when coupled with a specialized queuing discipline results in absolute[1] flow rate fairness.

There are many distributed protocols designed to achieve fairness in wireless networks [14], [16], [18]–[20], including WMNs [4], [21]. Most of the proposed schemes achieve fairness by limiting flow rates to the fair share of the network capacity. This requires actively maintaining network global state and flow synchronization among distributed nodes. The proposed TMAC protocol, on the other hand, does not estimate the fair rate for each flow. Instead, it establishes priority scheduling for transmitting data packets at each node according to the local condition. The use of priority scheduling has also been investigated in the literature [12]–[14], [16], aiming to enforce service differentiation and ensure QoS. This is made possible by taking advantage of backoff increment functions or interframe spacing [13], [14]. However, solely relying on backoff timers may worsen the vicious impact due to the hidden terminal and masked node problems, which simply leads to starvation [11]. TMAC uses the packet age as the priority metric to resolve the above deficiency. Similar to [12], [17], TMAC uses request/grant control messages to achieve higher throughput and fairness.

This paper is organized as follows: In Section II, we introduce TMAC and describe its various functional blocks. In Section III, we describe the design challenges in implementing TMAC over 802.11 radios and propose optimizations for improving its behavior. In Section IV, we present TMAC

[1] Absolute fairness is the equal distribution of resources among competing nodes.

performance analysis using simulation results. We conclude in Section V and describe some directions for future work.

II. Timestamp-ordered MAC

In this section we provide an overview of TMAC, including a description of the proposed TMAC priority scheduling mechanism and its impact on flow rate fairness. We also describe the role of queueing discipline in improving fairness for TCP flows.

A. Overview

The fundamental idea behind TMAC is to schedule packets based on their age as identified in their timestamps. In wired networks, control messages can be used to achieve consensus between nodes. However, any message exchange requiring global co-ordination incurs a significant overhead in multi-hop WMNs. Our proposed TMAC protocol addresses this by limiting the exchange of control messages to a subset of direct neighboring nodes only, (*i.e.,* one-hop away). We argue that the single-sink property of WMNs allows us to limit ordering enforcement on nodes with a parent-child relationship[2]. This local ordering can be achieved by an explicit exchange of control messages between nodes. Each node maintains a table of its child nodes. Whenever a node has a packet to send, it advertises the priority (*i.e.,* age) of the packet in the head of the transmission queue by multicasting a *request* message to its child nodes. When a child receives this message it responds with a *grant* message only if the requesting node has a higher priority than any packet pending transmission at the child node. When grant messages are received from all children, the packet is transmitted.

B. Impact on starvation and global fairness

TMAC uses timestamps to measure the packet age and influence its scheduling priority. These timestamps enforce a local ordering between neighboring nodes. For example, a node cannot transmit a packet until the packet has a higher priority (*i.e.,* a lowest timestamp) than the packets of its child nodes. The mechanics of TMAC require a transmitter to poll its children and seek confirmation that they do not have older packets awaiting transmission. This explicit polling ensures that a node can not starve its children at the cost of its own transmission.

The local ordering enforced by TMAC creates a back-pressure that translates into global ordering in WMNs with a single gateway. Since all flows traverse this gateway, the local ordering enforced on one-hop neighbors of the gateway propagates to all flows traversing them. For example, suppose nodes N_1 and N_2 are one-hop away from the gateway and nodes N_3 and N_4 are two or more hops away. Suppose that there exist flows f_3 and f_4 originated from nodes N_3 and N_4 respectively. The local ordering between N_1 and N_2 creates a backpressure such that packets of f_3 and f_4 are relayed according to their priorities. The time for backpressure to propagate within the network is a function of the node depth. This determines the latency incurred in converging distant nodes to their fair rate. We evaluate these flow rate convergence characteristics of TMAC in Section IV.

C. TCP fairness and queuing discipline

TCP flow rate is clocked with its Round Trip Time (RTT). With a faster feedback loop, nodes closer to the gateway can quickly build up larger TCP congestion windows compared to distant flows. Thus, the buffers at one-hop and two-hop nodes are largely populated with packets originating locally. If we use a simple DropTail queue with FIFO discipline, distant flows will experience packet drops from queue overflows when they reach these two-hop and one-hop nodes. Thus, the queueing discipline is integral in improving the fairness of TCP streams in WMNs.

TMAC uses a variant of Fair Queueing (FQ) by separating DATA packets from ACKs. Since TCP ACKs are cumulative, its congestion control mechanisms may not be triggered even when some ACKs are lost as long as an ACK with a higher sequence number gets delivered. Both DATA and ACK queues are sorted by timestamps such that the packets at the head of the respective queues are the oldest packets that are next scheduled for transmission. A locally generated packet is assigned a timestamp when it reaches the head of queue. This, however, leaves the locally generated traffic vulnerable to indefinite preemption by relay packets that have already been assigned a timestamp by their source nodes. We prevent this by partitioning our queue space into *rounds* such that relay packets from a round k cannot preempt packets generated locally in round $k + 1$.

III. TMAC design for CSMA/CA radios

In this section we describe our TMAC implementation for CSMA/CA radios using the control frames in IEEE 802.11. We also propose an optimization technique to reduce the overhead of these control messages.

A. TMAC implementation over 802.11

TMAC can be implemented through minor modifications to the IEEE 802.11 protocol. The modifications include the design of request/grant messages, as well as associating a timestamp with a data frame through its journey in the network.

Request/grant messages: A TMAC node requires request/grant messages to poll its neighbors about the state of packets pending for transmission. Our TMAC implementation uses modified RTS/CTS control frames to build this request/grant messaging framework. We introduce two modifications in the way RTS/CTS control frames are exchanged. First, the RTS frame is delivered to selected child nodes rather than a single designated receiver. This can be achieved either by transmitting RTS as a broadcast frame or by making the neighbors promiscuously capture the frame. Second, all neighbors receiving an RTS message should respond with a CTS message as long as the received RTS has a lower timestamp than any pending local transmission. The initial

[2] A parent node is the next node on the route towards the gateway.

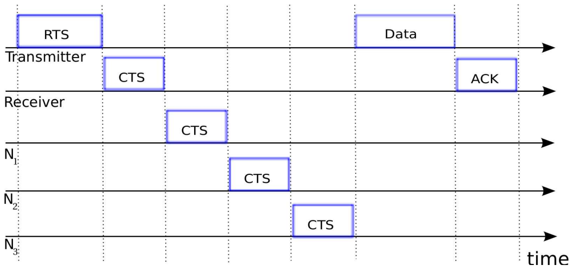


Fig. 1: Multicasting control frames in IEEE 802.11 radios

sender triggers data transmission only after receiving CTS frames from a sufficient number of children.

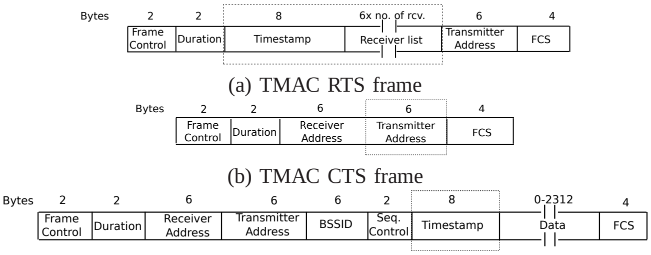
The proposed scheme may result in collision among CTS frames when multiple child nodes respond to an RTS. Therefore, these child nodes need to schedule their transmissions. We have implemented the scheme proposed in [6] using broadcasting and multicasting wireless transmissions and adapted it to control messages. The main idea is to append the neighbor addresses in the RTS in the order which they are expected to transmit CTS. Thus, a node responding to an RTS waits for a predefined amount of time T before launching the CTS message as follows:

$$T = (order - 1) \times (CTS\, transmission\, time)$$

These modifications to the RTS frame structure not only support the scheduling of CTS transmissions, but also enables the polling of specific neighbors for their CTS messages. Figure 1 shows the exchange of these control frames between a transmitter and a receiver with three neighboring nodes N_1, N_2, and N_3.

Timestamp generation: 802.11 radios achieve time synchronization by periodically exchanging timestamp-carrying beacons between neighboring nodes. We have implemented timestamps based on the synchronized clock among nodes. Our results in Section IV show that such synchronization is sufficient to ensure the ordering of packet transmissions required for the proposed TMAC protocol.

Revised RTS/CTS and Data frames format: We modified the 802.11 RTS/CTS and DATA frames to support TMAC protocol as shown in Figure 2.



Bytes	2	2	8	6x no. of rcv.	6	4
	Frame Control	Duration	Timestamp	Receiver list	Transmitter Address	FCS

(a) TMAC RTS frame

Bytes	2	2	6	6	4
	Frame Control	Duration	Receiver Address	Transmitter Address	FCS

(b) TMAC CTS frame

Bytes	2	2	6	6	6	2	8	0-2312	4
	Frame Control	Duration	Receiver Address	Transmitter Address	BSSID	Seq. Control	Timestamp	Data	FCS

(c) TMAC DATA frame

Fig. 2: 802.11 modified frame structure as used in our TMAC implementation

RTS frames have been modified as follows: a Timestamp field is appended (8 Bytes); this corresponds to the Timestamp field included in the Beacon frames per the 802.11 standard specifications. The Receiver address list (6 Bytes × no. of receivers) specifies the list of child nodes that are required to respond with a CTS. The Duration field is updated such that it reflects the time required for completing the transactions, including the additional CTS transmissions from selected children.

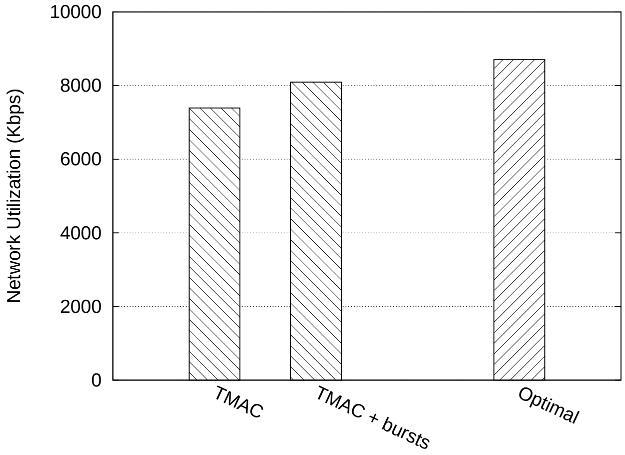


Fig. 3: Impact of TMAC optimizations on the utilization of a 5-hop chain with 12 Mb/s wireless link rates

CTS frames are appended with a Transmitter Address field (6 Bytes). This allows the RTS transmitter to differentiate between CTS frames from various children.

DATA frames are appended with a Timestamp field (8 Bytes). This allows the receiver to sort its transmit queue based on the age of the DATA packet.

B. Interface queue

An interface queue design satisfying the fairness requirements discussed in Section II-C is implemented as follows: packets arriving to a certain queue are either *fresh* (*i.e.,* locally generated) or timestamped packets (DATA or ACK). A *fresh* packet is placed at the tail of the queue. A timestamped packet is inserted in the queue sorted according to its timestamp. Note that *fresh* packets in the queue have not been assigned a timestamp yet. At this stage, *fresh* packets should first be placed between rounds of transmissions to prevent preemption by other flows packets. Consequently, if the tail of the queue has a *fresh* packet and a timestamped packet arrived with a timestamp larger than all the other timestamped packet, then it is placed in the tail of the queue. It is then scheduled in the next round of transmissions.

C. Mitigating TMAC control message overhead

The control message exchange required for TMAC may cause significant performance penalty. Recall, each RTS frame triggers CTS frames from child nodes. By default, these control frames are transmitted at the base rate, further increasing the impact of this overhead. We now propose an optimization technique to alleviate this overhead.

We propose using data bursts to amortize the overhead associated with the control message exchange. This allows a node to forfeit requesting grant messages from its neighbors for a fraction of the transmissions, allowing the grants to be effective for more than one transmission. For example, a burst length of five indicates that each received grant is effective for five transmissions. Selecting the proper burst size is an important configuration parameter. Larger bursts can significantly reduce the control frame overhead, yet it may introduce short-term unfairness between flows. Figure 3 shows the effect of bursts on network utilization in a 5-hop chain topology. We define the *network utilization* metric as $\sum_{i=1}^{N} x_i \times l_i$, where N is the number of active flows, x_i is the goodput of the i^{th} node, and l_i is the number of hops along the routing path between the i^{th} node and the gateway. Figure 3 shows that bursts can increase the network utilization by 9.5% in a 5-hop parking lot topology. Our experiments in Section IV use TMAC with data bursts.

IV. Performance evaluation

We have implemented and evaluated the performance of our proposed TMAC protocol in ns-3. Our simulation parameters are shown in Table I. We discard the first 20 sec. of simulation trace as the initial transients for establishing routes and populating ARP tables.

Parameter	Value
Link rate	12 Mb/s
MAC protocol	IEEE 802.11a
Packet size	1500 B
Interface queue size	500 packets
Routing protocol	OLSR

TABLE I: Simulation parameters

We use *Jain's Fairness Index* (JFI) [5] to quantify the fairness of our measured rate allocation. We normalize the simulation results to the *optimally fair* flow rate distribution obtained with the collision domain network capacity model proposed by Jun and Sichitiu [8]. For 802.11 radios, the PHY and MAC layer overhead reduces the nominal MAC layer capacity to be much less than the corresponding link rate. TCP ACK overhead incurs an additional penalty. The achievable goodput of a single-hop TCP flow, shown in Table II, is used as a baseline for the 'optimal' (*i.e.,* no collisions) link capacity by the collision domain model.

Link rate	Optimal goodput (Mb/s)
12 Mb/s	8.5

TABLE II: Measured 'optimal' goodput for a TCP flow in a 1-hop network

A. Parking lot topologies

We evaluated TMAC with several variations of parking lot and grid topologies. The spacing between nodes is 200 m. Using the default ns-3 radio parameters, only adjacent nodes in the chain are within transmission range and nodes upto two-hops away are within interference range. Each node initiates an uplink TCP flow to the gateway.

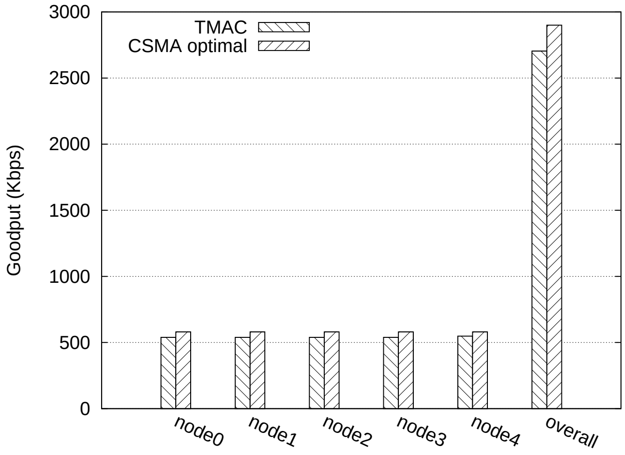


Fig. 4: Per-node goodput for a 5-hop parking lot topology

TMAC fairness: First, we analyze flow rate fairness characteristics of TMAC. We first describe our results for a 5-hop chain. Nodes are numbered from 0 to 4, where node 0 is the farthest node from the gateway. Figure 4 shows the throughput obtained by TMAC compared to the reference optimal results discussed earlier in this section. TMAC registers a capacity drop of approximately 7% compared to these reference results.

We have extensively evaluated TMAC over a number of additional parking lot topologies, varying the size from 2-hops up to 6-hops. Our results are tabulated in Table III. For network utilization, we list the values normalized to the reference optimal results. TMAC achieves a minimum JFI of 0.99 and an average network utilization of around 93%.

Scenario	Norm. Network Utilization	JFI
2-hops	93.38%	0.999
3-hops	93.35%	0.999
4-hops	94.20%	0.999
5-hops	93.04%	0.999
6-hops	93.80%	0.998

TABLE III: TMAC performance analysis in parking lot topologies with 12 Mb/s wireless links

TMAC convergence rate: We use a 4-hop parking lot topology to characterize the convergence time for various TCP flows. This experiment aims to obtain the time required for a new flow to converge to its fair rate allocation. At the beginning of the simulation, all flows are inactive. At time 10 sec. the 1-hop node (node 3) initiates a flow destined to the gateway. Every 10 sec, each of the other nodes initiates a new flow destined to the gateway. Figure 5 shows the instantaneous throughput of each flow (averaged over a 1 sec. interval). We observe that all flows successfully converge to their fair share assignments within 1 sec. from initiation.

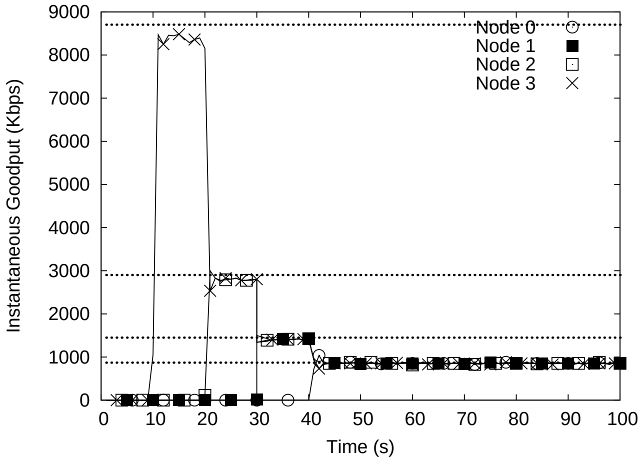


Fig. 5: The instantaneous goodput of a 4-hop parking lot topology. New flows are initiated every 10 s. starting with the 1-hop flow from (node 3) at time 10 s. Dotted lines show the optimal throughput.

B. Grid topologies

We extend our experiments to grid topologies as shown in Figure 6. The vertical and horizontal spacing between nodes is 200 m; thus, nodes in a 4x4 grid topology have up to four neighbors in the transmission range and up to 11 neighbors in the interference range. The number inside a node indicates its hop-count number along the shortest path to the gateway. All nodes in the network have an active TCP connection with the gateway.

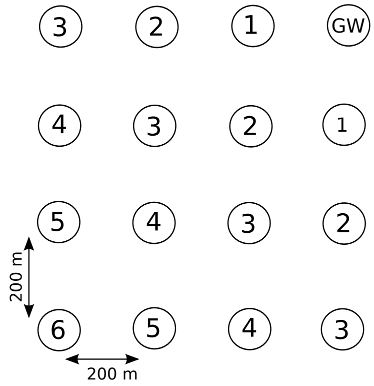


Fig. 6: A 4x4 grid topology. Node numbers represent the number of hops along the shortest path to the gateway

Grid size	Norm. Network Utilization	JFI
2x2	92.42%	0.999
3x3	91.17%	0.992
4x4	90.18%	0.993

TABLE IV: TMAC results for grid topologies with 12 Mb/s links

We performed several simulations while varying the grid size from 2x2 to 4x4. Our results are shown in Table IV. TMAC achieves a network utilization higher than 90% while maintaining a JFI fairness of at least 0.99. Figure 7 shows detailed results for TMAC compared to CSMA and CSMA/CA for the 4x4 grid. Both CSMA and CSMA/CA have nodes experiencing unfairness and starvation; approximately 45% and 65% of the nodes starve with CSMA and CSMA/CA, respectively. TMAC (both with and without data bursts) improves fairness amongst nodes. Using data bursts further increases average flow rate by approximately 15–20%. Thus our conservative choice of burst size parameters maintains a balance between the throughput and fairness requirements of this network.

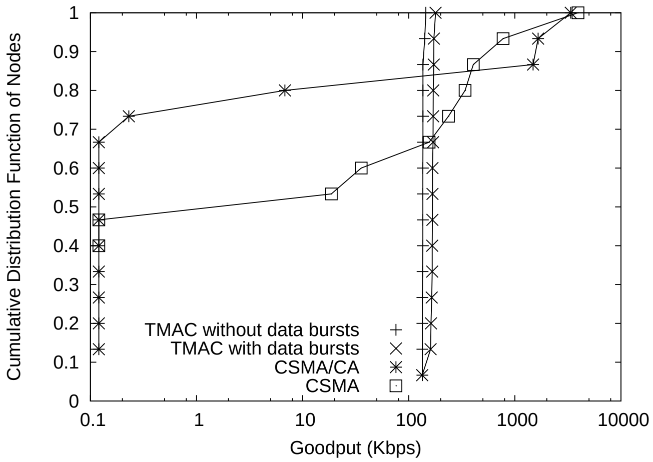


Fig. 7: The CDF of goodput values in a 4x4 grid topology

The TMAC convergence time, which is the time where the instantaneous throughput (we use granularity of 1 sec.) reaches its fair share, for the presented grid topologies is shown in Figure 8. We observe that all nodes converge to their fair allocation within 7 sec. of the simulation run.

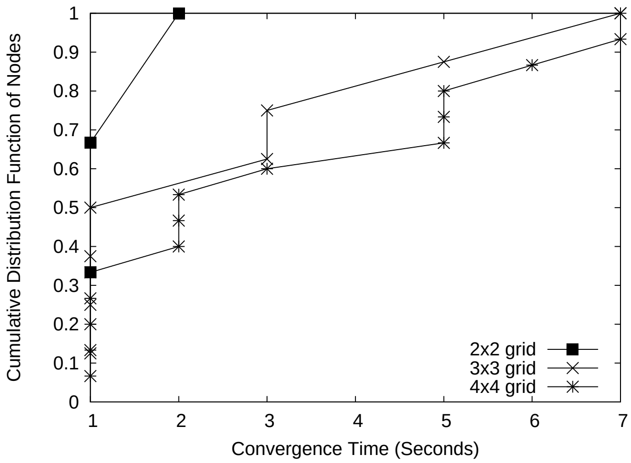


Fig. 8: The CDF of convergence time in grid topologies

V. Conclusions and Future Work

We presented TMAC, a timestamp-ordered extension for CSMA/CA protocol. TMAC enforces fairness through prioritizing the transmission of packets based on their age. We implemented TMAC message exchange using RTS/CTS control frames in the 802.11 radios. We proposed data bursts to overcome the RTS/CTS exchange overhead in WMNs. TMAC successfully achieves resource allocation fairness and maintains over 90% of maximum link capacity in parking lot and grid topologies.

We are further investigating optimizing the performance of TMAC by inverting the use of grant messages. For example, in response to a request message from a sender S, a neighbor N instead responds with a deny message if it has a packet with an older timestamp pending for transmission. This can also suppress further transmission of messages from neighbors which have pending transmissions with a lower timestamp than S but higher than N. However, the challenge is accounting for lost messages; for example, the loss of deny-messages can inherently be interpreted as a grant. This is problematic for wireless networks with high loss rates and needs to be further investigated.

References

[1] A. Acharya, A. Misra, and S. Bansal, "MACA-P: a MAC for concurrent transmission in multi-hop wireless networks", *IEEE Journal on Selected Areas in Communication*, Vol. 24, No. 11, pp. 1995–2004, 2006.

[2] I. Akyildiz, X. Wang, and Weilin Wang,"Wireless mesh networks: a survey", *Computer Networks and ISDN Systems*, Vol. 47 No. 4, pp. 445–487, 2005.

[3] B. Bensaou, Y. Wang, and C. C. Ko., "Fair medium access in 802.11 based wirless ad-hoc networks", *In Proc. of the ACM MobiHoc*, pp. 99–106, 2000.

[4] V. Gambiroza, B. Sadeghi, and E. W Knightly, "End-to-end performance and fairness in multi-hop wireless backhaul networks", *In Proc. of the ACM MobiCom*, pp. 287–301, 2004.

[5] R. K. Jain, D-M. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer system", *DEC Research Report TR-301*, 1984.

[6] S. Jain and S. Das, "MAC layer multicast in wireless multi-hop networks", *In Proc. of Comsware*, pp. 1–10, 2006.

[7] L. Jiang and SC Liew. Proportional fairness in wireless LANs and ad hoc networks, *In Proc. of the IEEE WCNC*, pp. 1551–1556, 2005.

[8] J. Jun and M. L. Sichitiu, "The nominal capacity of Wireless Mesh Networks", *IEEE Wireless Communications*, Vol. 10, No. 5, pp. 8–14, 2003.

[9] K. Jamshaid and P. Ward, "Centralized feedback-driven rate allocation mechanism for CSMA/CA-based Wireless Mesh Networks", *In Proc. of the ACM MSWiM*, pp. 387–394, 2008.

[10] L. Lamport, "Time, clocks, and the ordering of events in a distributed system", *Communications of the ACM*, Vol. 21, No. 7, pp. 558–568, 1978.

[11] S. Ray, J. B. Carruthers, and D. Starbinski, "Evaluation of the masked node problem in ad hoc wireless LANs", *IEEE Trans. on Mobile Computing*, Vol. 4, No. 5, pp. 430–442, 2005.

[12] C. Cicconetti, I. F. Akyildiz, L. Lenzini, "FEBA: a bandwidth allocation algorithm for service differentiation in IEEE 802.16 mesh networks", *IEEE/ACM Trans. on Networking*, Vol. 17, No. 3, pp. 884–897, 2009.

[13] I. Aad and C. Castelluccia, "Differentiation mechanisms for IEEE 802.11", *In Proc. of the IEEE INFOCOM*, pp. 209–218, 2001.

[14] N. H. Vaidya, A. Dugar, S. Gupta, P. Bahl, "Distributed fair scheduling in a Wireless LAN", *IEEE Trans. on Mobile Computing*, Vol. 4, No. 6, pp. 616–629, 2005.

[15] V. Bharghavan, A. J. Demers, S. Shenker, L. Zhang, "MACAW: A Media Access Protocol for Wireless LANś", *In Proc. of the ACM SIGCOMM*, pp. 212–225, 1994.

[16] H. Luo, S. Lu, V. Bharghavan, "A new model for packet scheduling in multihop wireless networks", *In Proc. of the ACM MobiCom*, pp. 76–86, 2000.

[17] J. I. Choi, M. Jain, M. A. Kazandjieva, P. Levis, "Granting silence to avoid wireless collisions", *In Proc. of the IEEE ICNP*, 2010.

[18] L. Tassiulas, S. Sarkar, "Maxmin fair scheduling in wireless ad hoc networks", *IEEE Journal on Selected Areas in Communications*, Vol. 23, No. 1, pp. 163–173, 2005.

[19] Z. Fang, B. Bensaou, "Fair bandwidth sharing algorithms based on game theory frameworks for Wireless Ad-hoc Networks", *In Proc. of the IEEE INFOCOM*, pp. 1284–1295, 2004.

[20] X. L. Huang, B. Bensaou, "On max-min fairness and scheduling in wireless ad-hoc networks: analytical framework and implementation", *In Proc. of ACM MobiHoc*, pp. 221–231, 2001.

[21] J. Lee, W. Liao, M. C. Chen, "An incentive-based fairness mechanism for multi-hop wireless backhaul networks with selfish nodes", *IEEE Trans. on Wireless Communications*, Vol. 7, No. 2, pp. 697–704, 2008.