A Distributed Coordination Scheme to Improve the Performance of IEEE 802.11 in Multi-hop Networks

Fengji Ye, Haiming Yang, Hua Yang and Biplab Sikdar

***Abstract*—This paper investigates the performance of IEEE 802.11 in multi-hop scenarios and shows how its aggressive behavior can throttle the spatial reuse and reduce bandwidth efficiency. An adaptive, layer-2, distributed coordination scheme for 802.11 using explicit medium access control (MAC) feedback is then proposed to pace the transmissions on adjacent nodes, thereby assisting the MAC protocol to operate around its saturation state while minimizing resource contention. Simulation results show that the proposed scheme outperforms the original 802.11 MAC.**

***Index Terms*—Wireless LAN, MAC protocol, IEEE 802.11**

I. INTRODUCTION

Prior research has shown deficiencies in the performance of the IEEE 802.11 MAC protocol [1] in multi-hop scenarios due to its intrinsic flaws in combating the hidden and exposed terminal problems [2], [3], [7]. The *pipeline efficiency*, characterized by the simultaneous use of the same spectrum along the path of data flow, relies on the coordination of transmissions at each relay node and becomes a dominant factor affecting the throughput and latency as the hop count grows. In order to improve 802.11's performance, researchers have considered adaptive RTS/CTS (request to send/clear to send) schemes [4], rate-adaptation [5] or changing the carrier sensing range [6]. However, existing MAC protocols only govern single-hop data delivery based on the interference information collected within the scope of a single hop, and lack support for concerted transmissions among relay nodes in a larger area. Thus, MAC protocols such as IEEE 802.11 experience sub-optimal spatial reuse and performance degradation in multi-hop networks.

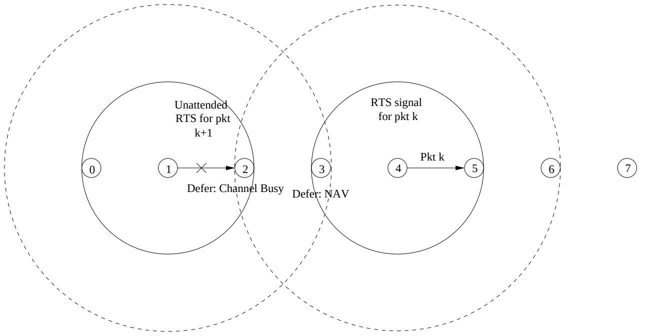

Fig. 1. A multi-hop flow traverses a chain and generates unattended RTS at node 1. The distance between successive nodes is 200m. The dotted line circle represents the carrier sense range (550m) while solid line circle represents the transmission range (250m). Node 2, in its deferral state when receiving the RTS from node 1, deliberately ignores the RTS and causes its failure.

The 802.11 MAC tends to over-utilize the spectrum by attempting more simultaneous transmissions than that are allowed in a neighborhood, and is thus prone to collisions and involuntary packet drops at the MAC layer. To observe this, consider the topology in Figure 1 in which a persistent flow traverses a chain from node 0 to node 7. Consider the instant when node 4 begins to forward the k-th packet of the flow to node 5, starting with a RTS frame. Assume that node 3 and 5 are within the transmission range of node 4 while node 2 and 6 are out of its transmission range but within its carrier sense range. As per 802.11 DCF, node 3 sets its Network Allocation Vector (NAV) according to the duration field of the RTS frame and starts to defer until the NAV counts down to zero [1]. Node 2 senses the signal in the channel but cannot correctly decode it, but defers while the channel is sensed busy. Suppose that packet $k+1$ now arrives at node 1, which senses the channel to be idle and thus sends an RTS to node 2. Since node 2 is in its deferral state and cannot reply with a CTS packet, node 1 would attempt the RTS repeatedly (after backoffs) until the RTS/CTS handshake eventually gets through.[1] We refer to the vain RTS attempts as *unattended RTS*, since its intended receiver purposely ignores the RTS owing to the deferral. Each unsuccessful RTS attempt reduces the throughput and increases the delay, and any node that overhears it may defer access unnecessarily. Furthermore, unattended RTS frames prompt the sender to repeat unsuccessful RTS attempts, making the situation even worse.[2]

In the IEEE 802.11 DCF, a backlogged node always attempts to transmit whenever it considers the channel in its vicinity to be idle, through physical and virtual carrier sensing. Ironically, this may result in lower pipeline efficiency because of the unattended RTS problem. However, MAC layer pacing can solve the unattended RTS problem and improve the pipeline efficiency as shown by the following simulation study. Consider the same topology as in Figure 1 with constant bit rate (CBR) traffic carried by 512 byte user datagram protocol (UDP) packets pumped continually from left to right with a link bandwidth of 2Mbps. Figure 2 shows the end-to-end

This work was supported in part by NSF under grant 0313095.

Manuscript submitted on 27 November 2007, revised on 29 April 2008, 30 July 2008, 07 November 2008 and 23-Jan-2009. Accepted 10 February 2009.

Fengji Ye is with Cisco Systems, San Jose, CA USA.

Haiming Yang is with Cisco Systems, Milpitas, CA USA.

Hua Yang is with Intel Corporation, Shanghai, China.

B. Sikdar is with the Department of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY USA.

---

[1] In IEEE 802.11 DCF, an upper limit is imposed on the maximum allowed RTS retransmissions.

[2] Unattended RTS frames persist, though at a little lower rate, as the channel bit rate increases.

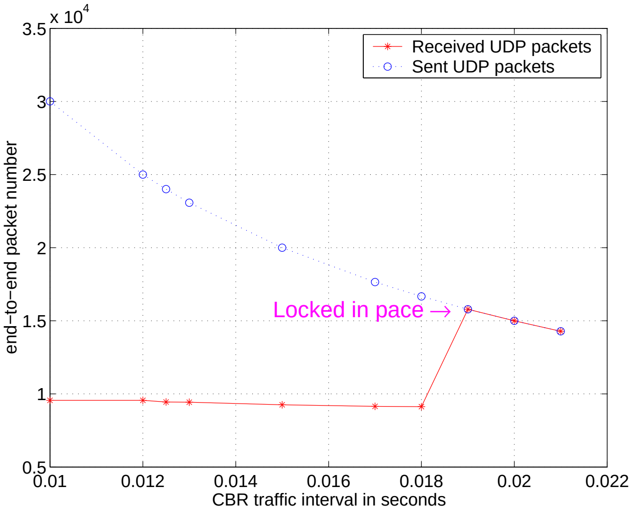

Fig. 2. The CBR traffic is pumped into an 8-node chain. The x-axis represents the CBR interval, and the y-axis represents end-to-end measurements in terms of transmitted and received UDP packets. At some point the CBR traffic rate is locked with the rate that achieves the maximum pipeline efficiency.

measurements in terms of transmitted and received packets as a function of the CBR traffic interval. While the throughput stays at a low value when the interval between data arrivals (and thus transmission attempts) is low, there is a phase transition when the rate of transmission attempts locks in with the rate that maximizes the pipeline efficiency. This is the rate at which the transmissions from the various nodes in the multi-hop path become coordinated so as to achieve the best tradeoff between alleviating the hidden and exposed terminals. It is interesting to note that a flow may achieve higher throughput if it properly yields channel access and maintains its pace in harmony, which in turn shows that aggressive transmission policies incur penalties.

To exploit the observations above, we develop an adaptive coordination mechanism to orchestrate the balanced transmissions of relay nodes in multi-hop scenarios, leading to performance improvements over the original 802.11 MAC. At the MAC layer of each node in the proposed scheme, traffic information in a multi-hop neighborhood is collected via a newly introduced MAC signal. The multi-hop traffic information is then conveyed to a link layer traffic shaper to dynamically control the rate of transmission attempts. Our layer-2 scheme is lightweight and transparent to upper layers, involving no cross-layer design issues. While the carrier sense range adjustment is an alternative to our approach [8], it requires the knowledge of the distance between nodes and is less effective in random networks. Note that a different notion of pipelining at the MAC layer is used in [9] where the contention resolution and data transmission are done in parallel using multiple channels.

The rest of the paper is organized as follows. The proposed coordination scheme is presented in Section II and then evaluated by simulations in Section III. The paper is concluded in Section IV.

II. DISTRIBUTED COORDINATION FOR IMPROVED SPATIAL REUSE

The proposed coordination scheme is deployed at the link layer and consists of two steps: **(1)** information collection and **(2)** a pacing mechanism. The information collection step uses a new control signal to obtain explicit information on intentional RTS drops and the associated congestion. This information is then used by the pacing mechanism to control and coordinate the rate at which a node makes transmission attempts.

A. *Transmission Rate Control on the Link Layer*

We use a Token Bucket Filter (TBF) to pace transmission attempts and provide support for MAC coordination. In the proposed architecture, a TBF is inserted between the interface queue and the MAC function. The TBF controls the rate at which the MAC layer receives packets and initiates transmission attempts. The rate at which the TBF generates tokens is adaptive and changes based on the network conditions. This adaptive pacing is accomplished by issuing tokens at a dynamic pace set forth by a pace tuner. The tuner coordinates the transmission rate between neighboring nodes through explicit MAC feedback (discussed in Section II-B). The MAC feedback is provided in the form of information on the incidence of unattended RTS packets and the rate of token generation is inversely proportional to it.

It is worth mentioning that the optimal pace for persistent traffic usually exceeds the maximum backoff window allowed in IEEE 802.11. For example, in Figure 2 optimality is achieved when the CBR traffic interval is approximately 19 milliseconds. Hence we do not implement pacing inside 802.11 DCF by adding extra backoff time for outgoing packets.

B. *Introducing Explicit Feedback to 802.11 DCF*

Section I showed that an unattended RTS represents an early indication of throttled spatial reuse. We thus use it as a feedback for triggering pace adjustments at a sender so that it may probe for the optimal transmission rate in its neighborhood. We modify the 802.11 DCF to incorporate this MAC feedback.

1) Modifications on the MAC receiver side: The receiver is responsible for tracking any unattended RTS frames and conveying this information to MAC sender. Inside the 2-byte Frame Control field of the CTS frame, there are two unused subfields named More Fragments field and Retry field, respectively, with each taking up one bit and always set to zero. We use these two single-bit fields to deliver the pacing feedback to the sender while keeping our scheme compatible with the original 802.11. We introduce two new bits to replace the unused old fields: *EPF bit* for backward compatibility, and *SLW bit* for pace tuning.

- **EPF (Explicit Pacing Feedback)**: This bit is set to 1 if explicit pacing feedback is enabled on the receiver node. For backward compatibility, it is set to 0 on non-pacing nodes.
- **SLW (Slow Pacing)**: This bit is set to 1 by the receiver if it successfully receives but intentionally declines at

Algorithm 1 MAC sender pace tuning using explicit pacing feedback

```
$retryCount \leftarrow 0$
transmit RTS frame and initiate a timer
while CTS not received before timeout do
  if $retryCount < ShortRetryLimit$ then
    $retryCount \leftarrow retryCount + 1$
    retransmit RTS frame and restart the timer
  else
    abort transmission and notify upper layer
    QUIT the algorithm
  end if
end while
check the validity of the received CTS frame
if EPF bit is set to 1 then
  if SLW bit is set to 1 then
    decrease TBF token issue rate
  else
    increase TBF token issue rate
  end if
end if
proceed to transmit DATA frame
```

least one RTS request due to deferral since its last CTS transmission; otherwise it is set to 0. This bit informs the sender whether its transmission rate is too fast causing unattended RTS and thus if it should slow down. The SLW bit is always set to 0 on non-pacing nodes.

2) Modifications on the MAC sender side: When a sender receives the CTS frame containing the pacing feedback, it uses a token bucket filter to update its transmission rate. The whole process is described in Algorithm 1. Since our scheme uses the same mechanism for contention-based access as in 802.11 DCF, all routine backoff or deferral operations are not shown in this algorithm. For each outgoing RTS, the sender starts a timer to wait for the corresponding CTS, and retransmits the RTS after a backoff in case of timeout. The total retransmission attempts should not exceed $ShortRetryLimit$, set to 7 in 802.11. Once the expected CTS is received, the sender proceeds to retrieve its EPF bit to check if pacing feedback is carried in this CTS frame. If feedback is available, the pace is decreased if the SLW bit is set to 1 and increased otherwise. The pace update method can be either linear or multiplicative. In the latter case, the TBF token interval is multiplied by a ratio every time a new feedback is received. The comparison of different pacing update methods is relegated to Section III.

For any overwhelmed node that is not able to answer every incoming RTS, a slow-pacing feedback (SLW=1) to any aggressive sender in its neighborhood is important, whether this sender generates the latest unattended RTS or not. A node thus sets the SLW bit even in CTS frames for nodes other than the one that sent the unattended RTS. This does not diminish the effectiveness of our scheme because we do not differentiate between flows at a specific node as spatial reuse is the sole concern.

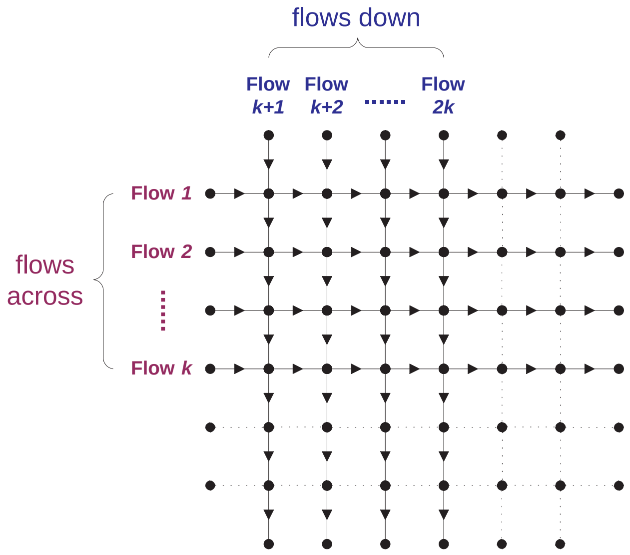

Fig. 3. 8 by 8 grid topology with 200m as the separation distance between neighboring nodes. In each simulation, k flows traverse across the network, while another k flows traverse down. Each flow has 7 hops.

C. Discussion

Any node using our enhanced 802.11 DCF with adaptive pacing, whether acting as a sender or receiver, can seamlessly work with any other non-pacing original 802.11 node, if any. The proposed scheme is thus backward compatible and can achieve a smooth technical migration. Also, our scheme applies pacing at the MAC layer, as opposed to transmission control protocol (TCP) pacing [10], [11]. The reason is three-fold: First, this implementation is independent of upper layer protocols, can handle hybrid traffic in an unified framework and does not require cross-layer design. Second, since TCP requires different end-to-end control strategies in wired and wireless environments, TCP pacing is not a universal solution effective in heterogeneous networks [10]. However, layer-2 pacing only applies to the wireless stations and thus evades this problem. Finally, layer-2 pacing can instantaneously use single-hop feedback for timely MAC coordination among neighbors, leading to faster convergence than in transport layer implementations.

III. SIMULATION RESULTS

This section evaluates the performance of our scheme via simulations in *ns-2*. All nodes have a single transceiver with 2Mbps bandwidth and all flows carry TCP traffic with packet size of 512 bytes. Ad hoc On-Demand Distance Vector Routing is used as the routing protocol. Four possible rate adjust mechanisms are considered: additive increase additive decrease (AIAD), additive increase multiplicative decrease (AIMD), multiplicative increase additive decrease (MIAD) and multiplicative increase multiplicative decrease (AIAD). All end-to-end measurements are made over 30 seconds at the MAC layer.

We first revisit the topology in Figure 1 and now consider a TCP flow from node 0 to node 7. With just one flow in the

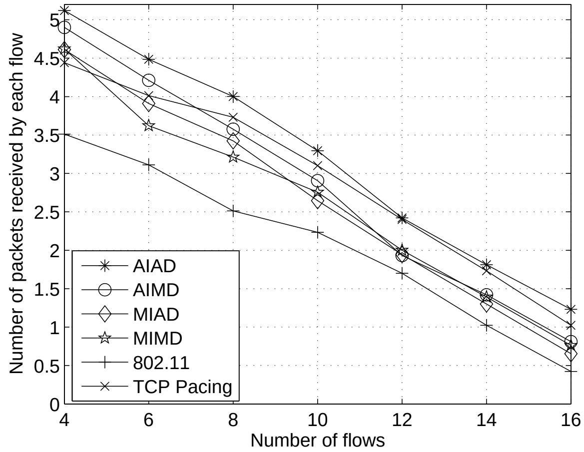

Fig. 4. Per flow end-to-end throughput measurements for TCP traffic in a random network.

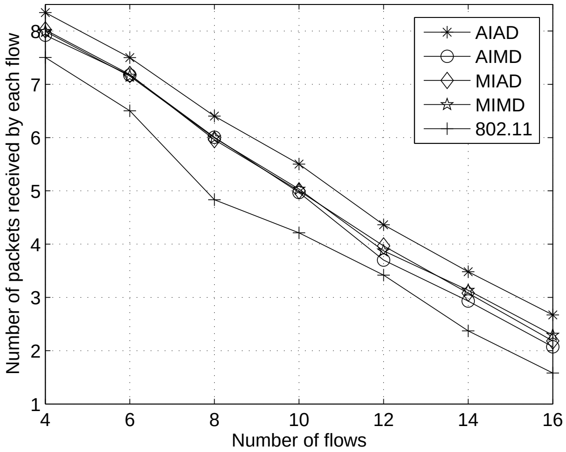

Fig. 6. Per flow end-to-end throughput measurements for UDP traffic in a random network.

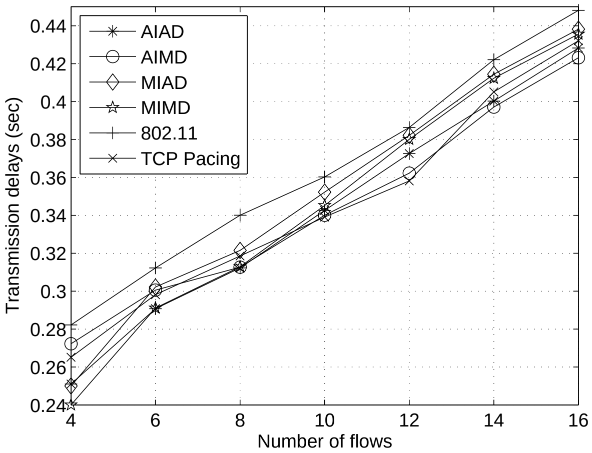

Fig. 5. Per flow end-to-end delay measurements for TCP traffic in a random network.

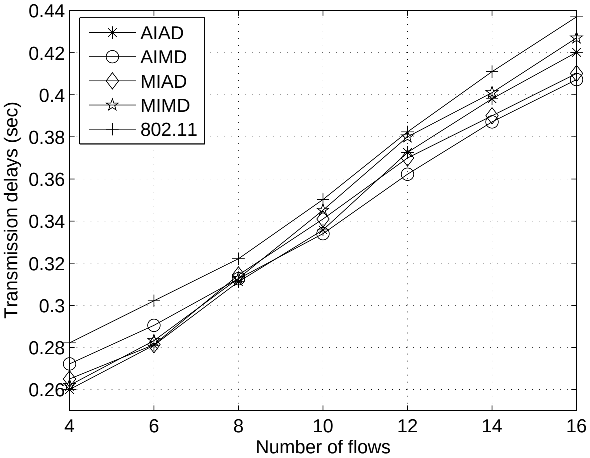

Fig. 7. Per flow end-to-end delay measurements for UDP traffic in a random network.

network, the proposed scheme achieves a throughput which is approximately 167% higher than that with plain IEEE 802.11. This suggests that pacing can significantly improve TCP performance by coordinating the traffic along the pipeline.

For more realistic scenarios, we next consider two types of topologies. In the first, a two dimensional grid as shown in Figure 3 was used, where the distance between two nodes along the horizontal and vertical lines along the grid was 200 meters. Each simulation run consisted of a given number of flows (k) in both the horizontal and vertical directions, as shown in Figure 3. In the second topology, 228 nodes were randomly distributed in a 1600m by 1600m region with uniform density. The source and destination pairs are chosen randomly for each flow. The averaged result of 30 runs is reported for each simulation setting.

A. Random Topologies

Figures 4 and 5 show the per flow TCP packets received and the average end-to-end delay as a function of the number of flows in the network. We note that the proposed scheme has higher throughputs than the original 802.11, with AIAD resulting in the greatest improvement. This suggests that a gentler adjustment of pace is more beneficial for the throughput of flows. We also note that the average delays are lower with the proposed scheme with AIMD typically having the lowest delays as the number of flows increases. This is because the reduction in the delays resulting from collisions and unattended RTS packets more than compensates for the delay introduced due to pacing. We also note that while TCP pacing can improve the performance, MAC layer pacing leads to greater improvements.

Figures 6 and 7 show the end-to-end throughput and delays when UDP is used at the transport layer. As in the TCP case, the proposed scheme achieves higher throughput compared to the original 802.11. Among the four increase-decrease policies, we again observe that AIAD leads to the best performance. Also, the end-to-end delays with the proposed scheme are lower than those achieved with the original 802.11.

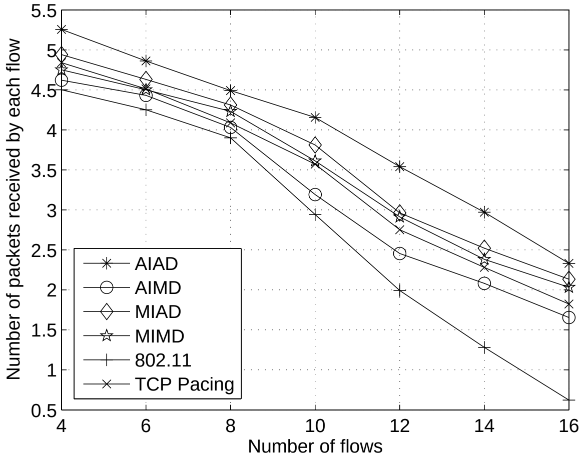

Fig. 8. Per flow end-to-end throughput measurements for TCP traffic in a grid network.

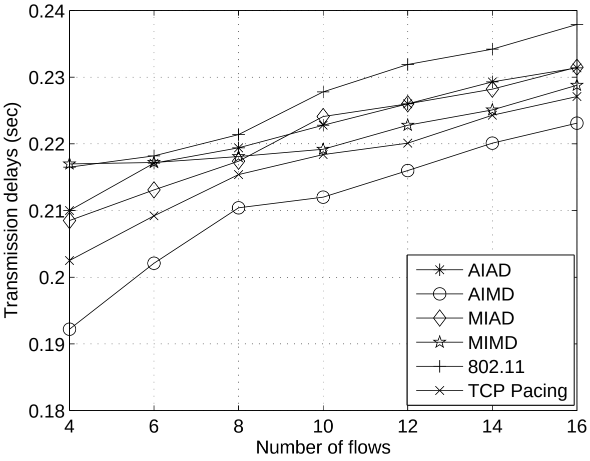

Fig. 9. Per flow end-to-end delay measurements for TCP traffic in a grid network.

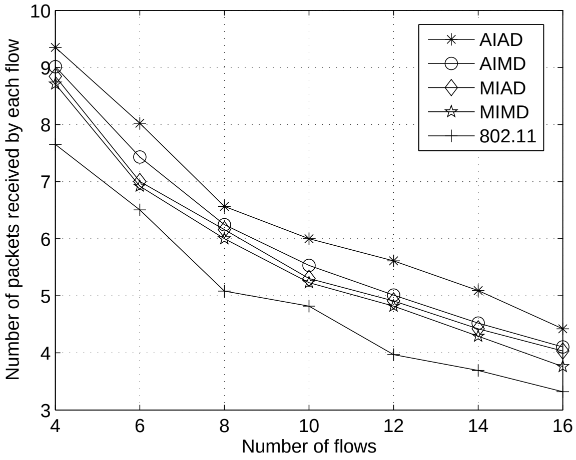

Fig. 10. Per flow end-to-end throughput measurements for UDP traffic in a grid network.

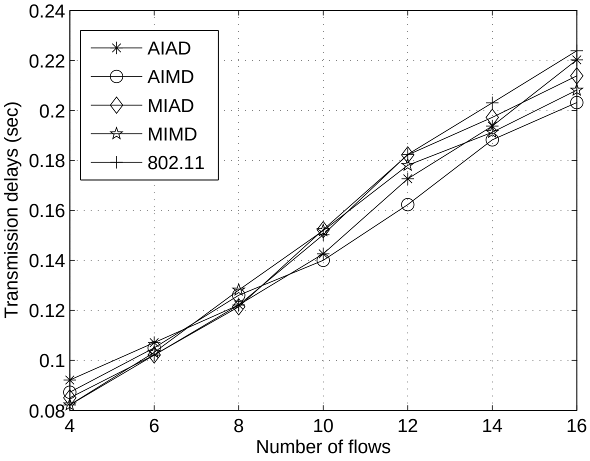

Fig. 11. Per flow end-to-end delay measurements for UDP traffic in a grid network.

B. Grid Topologies

In Figures 8 and 9 we show the end-to-end throughput and delays when TCP is used at the transport layer and the nodes are arranged according to the grid topology. Again, we observe that the proposed scheme leads to better throughput than the original 802.11 and the improvement is the greatest for the AIAD scheme. We also observe that while the delays are comparable when the number of flows in the network is small, the proposed scheme leads to lower delays when the network load increases.

For the grid topology with UDP as the transport layer, Figures 10 and 11 show the end-to-end throughput and delays, respectively. As in the other cases, we again observe that the proposed scheme leads to better throughput than the original 802.11 and the improvement is the greatest for the AIAD scheme. We also observe that while the delay with the original 802.11 may be a little smaller than those with the proposed scheme when the number of flows in the network is small, its delay increases as the number of flows increases and exceeds that of the proposed scheme.

We note from the results that the degree of improvement in the performance is a function of both the transport protocol as well as the topology. This is because while UDP generates packets to send independently of the MAC layer behavior, the adaptive nature of TCP controls the rate at which packets are generated in response to congestion (i.e. indirectly from the level of channel contention). This dynamic nature of TCP does impact the level of channel contention at the MAC layer.

From the results for the end to end delay, we note that the rate of increase of the delay for the case of TCP in grid networks is lower that the delay for TCP in random networks as well as UDP in grid and random networks. To explain this, we need to observe the four figures for TCP and UDP throughput in grid and random networks. For both TCP and UDP, the throughputs are higher in grid topologies than in random topologies since the level of contention in grid topologies is uniform at each node. Now consider AIAD as an example. As the number of flows increases from 4 to 12, the percentage reduction in the throughput is similar for both TCP and UDP: 34% and 53% for TCP in grid and random topologies and 37% and 48% for UDP. Now, the throughput of UDP is inversely proporational to the end to end delay. Hoever, for TCP the throughput is invesely proportional to the square root of the round trip time. This shows up as a relatively smaller variation in the end to end delays for the case of TCP in random topologies.

C. Parameter Selection

Next we investigate the performance of the proposed scheme as the increase and decrease parameter settings are varied. For the grid network with 6 TCP flows in the network, the highest throughput is obtained for the AIAD case for increase and decrease parameter values of 0.003 and 0.005, parameter values of 0.003 and 1.06 for AIMD, parameter values of 1.04 and 0.005 for MIAD and parameter values of 1.06 and 1.04 for MIMD (see [12] for more details). For other network sizes, topologies and transport protocols, the optimal parameter settings are very close to the numbers above. The results showin in this paper were generated by using the optimal parameters for each setting. A general observation is that with multiplicative decrease, there is a large variation in the performance of the protocol and there is no general trend in the results.

IV. CONCLUSION

In this paper, we investigated MAC coordination in ad hoc networks in the presence of persistent multi-hop flows from the perspective of spatial reuse. We proposed an adaptive pacing mechanism using a token bucket filter to balance the transmissions on adjacent nodes for better spatial reuse. Simulation results demonstrated the performance improvements of our scheme over the original 802.11 MAC.

REFERENCES

[1] ANSI/IEEE Standards 802.11, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 1999.

[2] J. Li, C. Blake, D. De Couto, H. Lee and R. Morris, "Capacity of ad hoc wireless networks," *Proc. ACM MOBICOM*, pp. 61-69, Rome, Italy, July 2001.

[3] S. Xu and T. Saadawi, "Does the IEEE 802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Networks?" *IEEE Communications Magazine*, vol. 39, no. 6, pp. 130-137, June 2001.

[4] H.-J. Ju, I. Rubin and Y.-C. Kuan, "An adaptive RTS/CTS control mechanism for IEEE 802.11 MAC protocol," *Proc. IEEE VTC*, pp. 1469-1473, April 2003.

[5] G. Holland, N. Vaidya and P. Bahl, "A rate-adaptive MAC protocol for multi-Hop wireless networks," *Proc. ACM MOBICOM*, pp. 236-251, Rome, Italy, 2001.

[6] J. Deng, B. Liang and P. Varshney, "Tuning the Carrier Sensing Range of IEEE 802.11 MAC," *Proc. IEEE GLOBECOM*, pp. 2987-2991, Dallas, TX, November 2004.

[7] F. Ye, S. Yi, and B. Sikdar, "Improving Spatial Reuse of IEEE 802.11 Based Ad Hoc Networks," *Proc. IEEE GLOBECOM 2003*, pp. 1013-1017, San Francisco, CA, December 2003.

[8] J. Zhu, X. Guo, L. Yang and W. Conner, "Leveraging Spatial Reuse in 802.11 Mesh Networks with Enhanced Physical Carrier Sensing," *Proc. IEEE ICC*, pp. 4004-4011, Paris, France, June 2004.

[9] X. Yang and N. Vaidya, "Explicit and Implicit Pipelining for Wireless Medium Access Control," *Proc. IEEE VTC*, pp. 1427-1431, Orlando, FL, October 2003.

[10] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla, "The Impact of Multihop Wireless Channel on TCP Throughput and Loss," *Proc. IEEE INFOCOM*, pp. 1744-1753, San Francisco, CA, March 2003.

[11] S. El Rakabawy, A. Klemm, and C. Lindemann, "TCP with Adaptive Pacing for Multihop Wireless Networks," *Proc. ACM MOBIHOC*, pp. 288-299, Urbana-Champaign, IL, May 2005.

[12] F. Ye, H. Yang, H. Yang and B. Sikdar, "MAC Coordination to Boost Spatial Reuse in IEEE 802.11 Ad Hoc Networks," Technical Report, Department of ECSE, RPI, Troy, NY ,2008. `http://networks.ecse.rpi.edu/∼bsikdar/tech_rep.pdf`