Demonstrating a Dynamic Multi-Channel Access 802.11 Mesh Network Prototype for High Bandwidth Requirement

Shashi Raj Singh, Mehul Motani
Department of Electrical and Computer Engineering, National University of Singapore, Singapore
E-mail: singh_shashiraj@yahoo.com, motani@nus.edu.sg

***Abstract*—Multi-Channel Access has the potential to solve the throughput and scalability issues that mar the performance of 802.11 mesh and adhoc networks. It allows multiple simultaneous transmissions in a radio neighborhood thereby increasing network throughput and scalability. In spite of this potential, it is limited to simulation studies and its implementation on hardware is rare. In this demo, we demonstrate working of a dynamic multi-channel access 802.11 network prototype. The prototype is based on a MAC protocol called Cooperative Asynchronous Multi-channel MAC (CAMMAC), which features multi-channel use in single antenna, asynchronous mesh/ad hoc network. The protocol employs a unique control-plane cooperation approach to achieve dynamic multi-channel access. The implementation is done using COTS WiFi cards and the ath5k Linux driver. To the best of our knowledge, this is the first work that implements dynamic multi-channel access using COTS Wi-Fi cards. The demo clearly demonstrates the performance superiority of multi-channel access over traditional single channel access under heavy traffic conditions.**

I. Introduction

802.11 mesh networks still suffer from throughput and scalability issues that will worsen as the technology usage is growing exponentially. One of the reasons behind the issues can be attributed to the single channel MAC design used by the 802.11 standard. In a mesh/adhoc scenario where multiple devices try to communicate, single channel use can act as a bottleneck to throughput and scalability. Also, the 802.11 standard theoretically has 3 and 20 orthogonal channels in 2 and 5GHz bands respectively. But the 802.11 network uses only single channel even when other channels may be free, which is very inefficient. To this end, Multi-channel access can provide a solution to the woes of 802.11 mesh/adhoc networks by allowing them to use all possible channels in the mentioned bands. Multi-channel MAC enables a node to access multiple channels and allows multiple simultaneous transmissions in a given radio neighborhood. This boosts the network throughput and allows more nodes in the network. Although, sufficient theoretical research [1]–[10] related to the multi-channel access is present, its experimental prototypes are rare.

In this demo, we have demonstrated working and advantages of the multi-channel access by building a dynamic multi-channel 802.11 network prototype using COTS Wi-Fi cards and open source Linux drivers. The prototype is based on a MAC protocol called Cooperative Asynchronous Multi-channel MAC (CAMMAC) [11], which features multi-channel use in single antenna, asynchronous mesh and adhoc networks. The demo also demonstrates a unique way that CAMMAC employs to achieve multi-channel access. This new approach of node cooperation provides an easy and low cost multi-channel access solution as compared to previous proposed approaches [1]–[10] by precluding the need for synchronization and additional hardware.

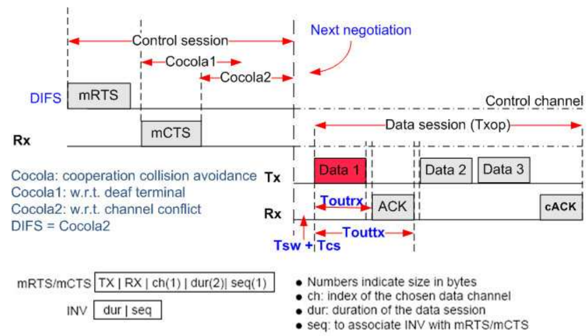

Fig. 1: Protocol features.

II. MAC Protocol Basics

CAMMAC is based on a new notion of control plane cooperation to solve the Multi-channel Coordination problem (MCC) [11]. Prior works propose solutions that either use dedicated radios or rely on time synchronization, which is cost inefficient or incurs significant overhead. By contrast, CAMMAC exploits neighboring nodes as another resource to solve the problem, providing a single-radio and asynchronous solution. For the prototype, we modified CAMMAC by reducing the number of control packets to make it more robust.

In the protocol, nodes maintain a spectrum usage table that stores channels that are currently in use in the network. There is a control channel shared by all the nodes to allow them to 1) negotiate data channels 2) alert a pair of nodes of MCC problem. On the control channel, nodes negotiate for a data channel by following a handshake called control session as shown in Fig. 1. In the control session, a sender and a receiver negotiate a data channel using an mRTS/mCTS exchange, which is followed by a cooperation phase (Cocola) where neighbors can alert the pair of any MCC problem by sending INV packet. Then in the data session, both nodes switch to their chosen data channel and the sender senses the channel for duration Tcs, and if free, sends packets to the receiver based on a transmission opportunity (TxOp) limit. The receiver replies with a combined ACK (cACK) at the end of the TxOp.

III. Implementation

The implementation of the prototype is done using a development platform that consists of Linux OS, ath5k driver and commercial WiFi cards. We introduced the CAMMAC protocol design as a *software MAC* as shown in Fig 3 by modifying the mac80211 stack, ath5k driver (2.6.30) and

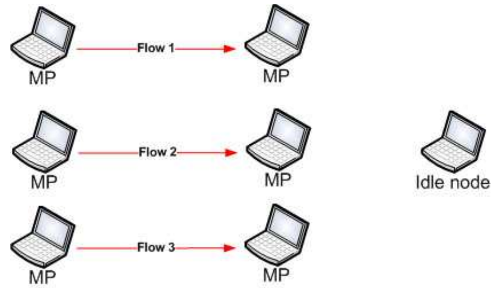

Fig. 2: Demo setup.

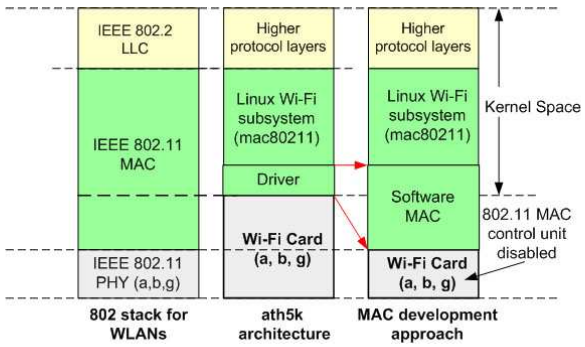

Fig. 3: Software MAC.

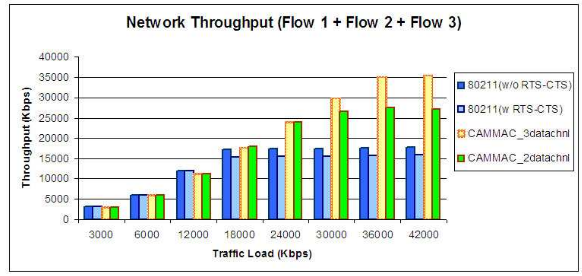

Fig. 4: Throughput comparision: CAMMAC vs 802.11 MAC.

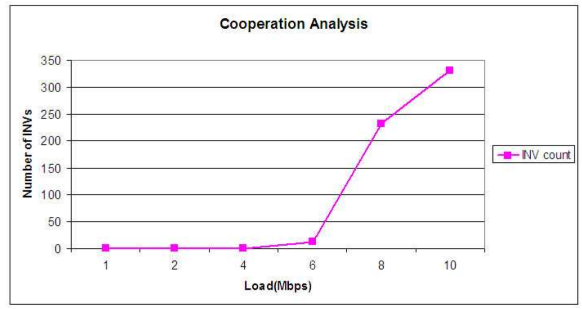

Fig. 5: INV sent in phase 2.

register values of the Wi-Fi card. We also disabled most part of the 802.11 control logic (distributed coordination function) in the card excluding carrier sensing. The *software MAC* control unit consists of 19 control logic states called internal control states, 9 timers and associated handlers. The unit uses current and previous state flags, *high resolution timers*, tasklets and hard interrupts from the hardware for running the control state machine in the driver.

IV. Demonstration

A. Description

The demonstration scenario shown in Fig. 2 consists of 7 laptops. The laptops form a mesh network, with 7 mesh points (MP). The 3 pairs in the center will be used to show throughput performance. In the topmost pair, one MP will stream a high definition video to the other one that will show qualitative performance. The other 2 pairs will be used for UDP traffic transfer using Iperf to show quantitative performance. The idle MP will act as a cooperative node by alerting the pairs of any channel conflict.

The demo consists of three phases; 1) original 802.11, 2) CAMMAC with 2 data channel and 3) CAMMAC with 3 data channel. In the first phase, we use original 802.11 MAC protocol and show throughput performance of the pairs. In the second phase, we use CAMMAC with 2 data channels. In this phase, there will be three pairs but only 2 data channels, so idle MP will cooperate and alert nodes of any possible channel conflict. In the third phase, 3 data channels are used by three pairs and collision is avoided by a proper channel selection strategy used by CAMMAC(choose the previously used channel first). By varying the Iperf traffic (Flow 2+3), we will show the effect on the video streaming in each phase. In case of the first phase where we use the original 802.11, the video hangs on applying heavy Iperf load (6Mbps). For the second and third phase, video plays smoothly for high loads (6Mbps) as the multi-channel access MAC allows nodes to use all the available data channels. This is not possible in case of the original single channel 802.11 MAC even when the channels are available. All the channels used in this demo will be in the 5 GHz band.

B. Results

Fig. 4 shows aggregated throughput (Flow 1+2+3) results for the Fig. 2 setup where all the traffic was (Flow 1,2,3) Iperf traffic. The results are for the three demo phases as described before. The first phase was divided into two phases based on the original 802.11 MAC (w RTS-CTS and w/o RTS-CTS) that used a single channel. For the CAMMAC, one unused channel was used as the control channel and, three (CAMMAC_3datachnl) and two (CAMMAC_2datachnl) unused channels were used as data channels. TxOp limit of 11 msec was used for the CAMMAC. As shown, the performance of CAMMAC slightly trails behind the 802.11 when traffic load (sum of applied iperf loads at the 3 Tx) is low. This is because a single channel is sufficient for mild channel contention, so the control session in CAMMAC acts as an overhead and degrades the performance. As shown in Fig. 5, we found that there was no INV sent when the load was less than 4 Mbps that also shows less contention. After that, as the traffic load increases, channel contention becomes more prominent and the single channel is no longer sufficient. As such, the throughput of 802.11 (w/o RTS-CTS) saturates below 18 Mbps and 80211(RTS-CTS) around 15.5 Mbps. For the CAMMAC, performance is far better than 802.11 due to its multi-channel use enabled by the multi-channel access MAC as compared to the fixed channel MAC use. Although the contention for the control channel also increases, it is much lower than data channel contention because of the small control-session duration.

C. Procedural Details

The demo requires a table with sufficient space to place 7 laptops. The laptops will require 7 power points for power supply. The setup time for the demo is 10 minutes.

V. Acknowledgement

This work was partially supported by project grant NRF2008NRF-POC001-078 from the National Research Foundation, Singapore, and project grant NRF2007 IDM-IDM002-069 from the Interactive and Digital Media Project Office, Media Development Authority, Singapore.

References

[1] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, "A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks," in *I-SPAN*, 2000.

[2] A. Nasipuri, J. Zhuang, and S. R. Das, "A multichannel CSMA MAC protocol for multihop wireless networks," in *WCNC*, 1999.

[3] N. Jain, S. R. Das, and A. Nasipuri, "A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks," in *IEEE ICCCN*, 2001.

[4] A. Adya, P. Bahl, J. Padhye, and A. Wolman, "A multi-radio unification protocol for IEEE 802.11 wireless networks," in *IEEE Broadnets*, 2004.

[5] R. Maheshwari, H. Gupta, and S. R. Das, "Multichannel MAC protocols for wireless networks," in *IEEE SECON*, 2006.

[6] J. Chen, S. Sheu, and C. Yang, "A new multichannel access protocol for IEEE 802.11 ad hoc wireless LANs," in *PIMRC*, 2003.

[7] J. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *ACM MobiHoc*, 2004.

[8] J. Zhang, G. Zhou, C. Huang, S. H. Son, and J. A. Stankovic, "TMMAC: an energy efficient multi-channel MAC protocol for ad hoc networks," in *IEEE ICC*, 2007.

[9] A. Tzamaloukas and J. Garcia-Luna-Aceves, "Channel-hopping multiple access with packet trains for ad hoc networks," in *IEEE Device Multimedia Communications*, 2000.

[10] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks," in *ACM MobiCom*, 2004.

[11] T. Luo, M. Motani, and V. Srinivasan, "Cooperative asynchronous multichannel MAC: Design, analysis, and implementation," *IEEE Transactions on Mobile Computing*, vol. 8, pp. 338–52, March 2009.