MAC Protocol Adaptation in Cognitive Radio Networks: An Experimental Study

Kuo-Chun Huang, Xiangpeng Jing and Dipankar Raychaudhuri

WINLAB, Rutgers University

671 Route 1 South

North Brunswick, NJ 08902, USA

{therom, xjing, ray}@winlab.rutgers.edu

Abstract - This paper presents an adaptive MAC (AMAC) protocol for supporting MAC layer adaptation in cognitive radio networks. MAC protocol adaptation is motivated by the flexibility of emerging software-defined radios which make it feasible to dynamically adjust radio protocols and parameters. Dynamic changes to the MAC layer may be useful in wireless networking scenarios such as tactical or vehicular communications where the radio node density and service requirements can vary widely over time. A specific control framework for the proposed AMAC is described based on the "CogNet" protocol stack which uses a "global control plane (GCP)" to distribute control information between nearby radios. A proof-of-concept AMAC prototype which switches between CSMA and TDMA is implemented using GNU radio platforms on the ORBIT radio grid testbed. Experimental results are given for both UDP and TCP with dynamic traffic variations. The results show that adaptive MAC can be implemented with reasonable control protocol overhead and latency, and that the adaptive network achieves improved performance relative to a conventional static system.

I. INTRODUCTION

The concept of cognitive radio (CR) [1] is to dynamically choose a wide variety of radio parameters and network protocol standards in order to adapt to observed radio link and network conditions. Such a cognitive radio system is capable of handling cross-layer parameter changes: at the physical layer, the agile radio can search so-called "spectrum white space" from unlicensed and even licensed bands to make good use of the spectrum opportunities with power control and modulation waveform adaptation; at the network layer, the cognitive radio system can also make runtime changes to the protocols used for packet transmission, to avoid collisions and congestion when network topology or traffic characteristics change. The ability to adapt different protocol layers with cognitive radio technology offers the prospect of improved networking performance in dynamic wireless networking scenarios such as tactical and vehicular.

There have been many research works addressing the physical layer agility of a cognitive radio system based on dynamic spectrum access (DSA) technique [2][3]. However, there still is lack of a systematic study of how the cognitive radio performs with medium access control (MAC) protocol adaptation. This paper proposes a novel adaptive MAC (AMAC) design for cognitive radio networks, with the ability to adapt between different types of MAC protocols based on the observation of network traffic changes. The candidate

medium access protocols usually have different advantages in various network situations, and the proposed AMAC protocol provides a framework to observe the runtime network changes and makes judgments for switching between those candidate protocols.

The proposed AMAC protocol is based on the "CogNet" cognitive radio protocol architecture described in [4]. The CogNet architecture includes the concept of a "global control plane (GCP)" which supports exchange of control information between networking cognitive radio devices. MAC adaptation under consideration here requires exchange of local cross-layer parameters and tokens to maintain the protocol consistency between radio nodes sharing the same physical channel. The same GCP framework has been used in earlier work for dynamic spectrum coordination in multi-radio environments [5], as well as for cross-layer ad hoc network routing [4].

In this paper, we will focus on the implementation aspects of AMAC protocol using GNU radios on the ORBIT radio grid testbed [6]. GNU radio is a type of software defined radio (SDR), which provides considerable flexibility for adaptation of both physical layer signal parameters as well as MAC protocol functions. The candidate medium access control protocols we have implemented for evaluation represent the two most commonly used classes of wireless network protocols: CSMA and TDMA. CSMA is easy to implement and is efficient for supporting short packet transmissions from bursty sources with light to medium traffic volumes. In heavier traffic scenarios, particularly with streaming sources, it is well known that TDMA provides a more efficient solution. Other MAC protocols can also be used in conjunction with AMAC, even though our GNU radio implementation is currently limited to CSMA and TDMA.

We evaluate the proposed AMAC protocol with various traffic types. The purpose of the study is to validate the control protocol by which AMAC coordinates MAC protocol switching and to measure the control overhead. System performance in terms of throughput and TCP file transfer delay is studied using both baseline static traffic and dynamic traffic types with varying packet size. Overall, these proof-of-concept GNU radio experiments within the ORBIT radio grid testbed at Rutgers University are intended to demonstrate the feasibility of building an adaptive wireless network using the CogNet GCP architecture. We believe that these experiments are among the first reported results for real-time adaptive MAC implementations using cognitive radio platforms.

In the next section, we begin by describing related work. That is followed by our network architecture in section III, the structure that we used for AMAC protocol. Then section IV provides a detailed discussion of the proposed AMAC protocol. In sec V, we provide ORBIT experiment results and we conclude with future work in Sec VI.

II. RELATED WORK

The benefits of separate control and data planes in the CogNet architecture have been previously explained in [4]. The global control plane (GCP) uses a low-rate radio PHY with wide coverage for robustness, and may include protocol modules for topology discovery, bootstrapping, and address assignment. The data plane protocol stack supports data communication via PHY, MAC and routing modules specified through an API that interacts with the control modules in the radio node. This GCP control architecture can be used for various design objectives ranging from spectrum etiquette [7], adaptive MAC under consideration here, as well as cross-layer path setup and routing [4].

In [8], the author uses the rendezvous channel (RC) to coordinate among wireless nodes in different channels. C-MAC operates over decentralized multi-channels CR networks with dynamic RC. Also, each data superframe has to sacrifice its beacon period to get control information which causes overhead to affect data transmitting. In [9] existing MAC protocols are combined into a single higher layer, using only local network feedback information to determine the optimum MAC protocol. However, this work does not consider the global network view for switching different MACs.

In [10], a TDMA implementation for software defined radios is proposed. Different software frameworks for TDMA are used for implementation: GNU radio for PHY layer development and Click for protocol development. However, the time slot for this TDMA design is ~ 1 second, too large a value to be useful for most applications. Ref [11] modifies the functions required for faster processing by the CR's FPGA instead of the host CPU, providing for better timing in a MAC layer implementation. Ref [12] also uses GNU radios for PHY implementation and Click for MAC development and reports preliminary experiments in the Hydra system.

III. NETWORK ARCHITECTURE

The proposed GCP-based cognitive radio network architecture is shown in Fig. 1. A global control plane is used to carry all the control information exchanged between nodes while the data plane is dedicated for data transmission. The concept of GCP can be implemented using various methods such as a separate control radio or a control mode the agile radio can periodically switch to. In this paper, we utilize an additional low-cost control radio operating on a dedicated control channel which is in charge of the resource reservation and allocation of radio and protocol parameters. The control radio is designed to have greater range and lower data rate as compared to the radio in data plane. The GCP assists with the initial bootstrapping as well as topology discovery when the new nodes join the network, and can subsequently support

adaptive MAC operation when network or environment condition changes. Radio parameters could be optimized for end-to-end data transmission in different scenarios with different traffic situations.
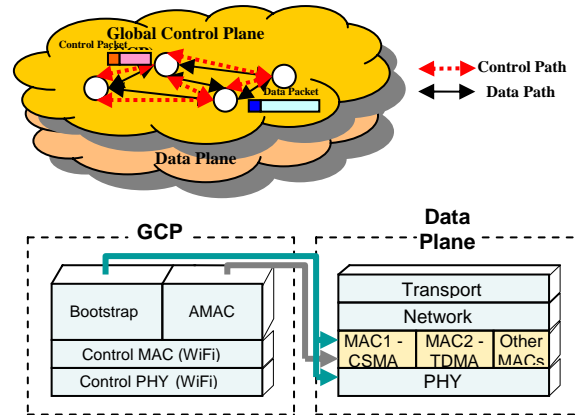

Figure 1: GCP and Data Plane structure

The data plane protocol stack on each node handles service data between the wireless nodes, while the GCP assists in establishing the operating PHY, MAC and routing parameters. Data is transmitted in the established data path negotiated using the GCP protocols. The actual data plane media access control protocols can be dynamically switched between candidate protocols in response to observed changes in network condition and/or service requirements. The AMAC protocol is implemented in the control plane and candidate MAC protocols are implemented in the data plane.

IV. AMAC PROTOCOL

The flexibility offered by cognitive radios makes it possible to design adaptive wireless networks that adjust the protocol stack used based on observed external conditions or service requirements. For example, in a vehicular scenario, there can be large variations in radio node density as cars move from an urban intersection to a highway. CSMA/CA protocol might work well in certain situations when the number of nodes is small, but it may be beneficial to switch to alternative protocols such as TDMA to avoid instability and hidden-node problems when the number of neighbor nodes increases dramatically. Additionally a separate control protocol can help to provide seamless roaming across networks supporting different network conditions. Thus, we develop a protocol framework for dynamic adaptation of MAC layer protocols, considering the examples of switching between CSMA/CA and TDMA.

(1) AMAC architecture

This "Adaptive MAC Protocol (AMAC)" mainly focuses on switching among different MAC protocols to achieve better overall network performance. The proposed AMAC protocol is able to incorporate channel coordination when available vacant channel is detected and to adjust MAC/PHY configurations according to different service requirements and network node density. The GCP outlined earlier provides a control framework to set up network adaptation functions.

It is important for cognitive radio nodes to start with bootstrapping and discovery process in this GCP-based framework. The bootstrapping function can help configure PHY/MAC capabilities and current status when the nodes power up. After initialization, a discovery protocol is executed to provide end-to-end reachability and determine optimum path information across multiple hops. Based on these exchanged control messages, nodes are able to initiate data transmission.

The proposed AMAC protocol is used to apply different operating conditions in which wireless nodes can achieve efficient data communication. It dynamically changes MAC behavior on-the-fly and configures per-hop PHY parameters of the data plane at each intermediate node. AMAC includes three phases as following:

- Baseline MAC selection: Each node starts with the MAC protocol which handles the nominal light network traffic condition (CSMA is used in this paper).
- PHY adaption: Nodes first try to discover if they can adapt PHY parameters (such as operating channel, power or modulation type) when the performance of data transmission drops significantly.
- MAC adaptation: When the PHY adaptation is not able to meet performance goals, nodes may initiate a switch of the MAC protocol. AMAC provides a mechanism to let each node negotiate with others and find the suitable MAC protocols commonly used for different network environments.

Node mobility in vehicular environments can result in major changes to the propagation environment, radio density and network topology. Cognitive radio nodes are capable of adjusting PHY parameters (e.g. frequency, power, rate, modulation) on the fly when the performance degrades. Each transmitting node will monitor performance (i.e. throughput or delay) of data transmission periodically. When the performance requirement goes below specified objectives , the nodes will initiate PHY parameters adaptation via the GCP to negotiate suitable configurations, such as switching to empty channels or power adjustment.

(2) MAC Switching Schemes for AMAC

When PHY adaptation alone fails to meet performance objectives, the next level of adaptation involves change of the MAC protocol. AMAC grants cognitive radio nodes a further way to solve degraded performance caused by contention. When there are nodes initiating video streaming in a high network density area, it may be hard to satisfy the required QoS with PHY adjustments. Instead, if the nodes can adjust the MAC protocol with time or frequency schedule management (i.e. TDMA or FDMA), the desired QoS can potentially be achieved. As another example, a mobile cognitive radio moving from a sparse rural area to an urban can switch between two MAC layer protocol (such as CSMA and TDMA) to deal with such major changes in operating environment.

When the cognitive radio node powers on, it first listens and collects control messages from nearby nodes by using GCP. If the node decides to join the network, it starts by flooding control messages in the GCP to form the initial network connections. Beacon message includes required information of node's states and ID. In Table I, we show the format of periodic control messages (operating MAC, frequency and power) which updates neighboring radios with a node's latest parameters.

Type	Flags	Sender ID
Beacon Transmit Power		Max PHY Rate
Operating Frequency		Operating MAC

TABLE I: GCP control message format

The key requirement in choosing a baseline operating MAC is to ensure flexible date communication. For instance, CSMA-based MAC is simple, flexible and able to handle most networks with low traffic load; TDMA-based MAC has a fixed structure and may avoid repeated channel contentions for scenarios with stream traffic. As a result, we choose CSMA as the baseline MAC and consider switching to TDMA when performance is degraded due to excessive contention.

The strategy for switching the MAC protocol is based on (1) performance degradation threshold; or (2) predicted future traffic patterns. We propose two approaches to help cognitive-radio nodes decide whether to request MAC switching. In the first approach, each node computes its performance in an observation interval while data traffic flows continue. If the performance is degraded below a threshold, the node will request a MAC switch. In this approach, we focus on reserving the node's performance in order to satisfy application requirements. In the second approach, each node checks its future traffic patterns in the MAC transmit buffer periodically. If the average packet size is large, the node will request a TDMA MAC switch in order to reduce contention. On the other hand, for smaller average packet size, CSMA MAC is preferable to prevent unnecessary wastage of slot time. Thus, we are able to adapt different MAC protocols according to their characteristics.

However, if all the cognitive radio nodes change their MAC protocol based on a local decision, the protocols used by all nodes may not be compatible. Thus, we introduce a voting scheme for nodes to agree on the MAC protocol to be used across the whole network. While a sender requests MAC switching based on previously outlined algorithms, it first broadcasts in the GCP about which MAC protocol it would like to apply. The nodes receiving this "MAC switch request" control message will respond with a GCP message indicating their local decisions. In this way, nodes vote for the MAC protocol to be used and the common protocol for the network is determined by the majority nodes. The control message format for the voting scheme is shown in Table II.

Type	Flags	Sender ID
MAC type		

TABLE II: Control message format for MAC switching request

An example network of data traffic starting from the sender to the receiver with AMAC capability handling both CSMA and TDMA modes is shown in Fig. 2. During the bootstrap process, all the nodes are able to obtain a global

view of the network by exchanging beacons. After data traffic is initiated, the sender node uses CSMA as its baseline MAC protocol. In the mean time, the sender computes the average throughput of its packets periodically (e.g., an average of ten packets). When the performance in the observation interval drops more than 20% compared to the previous observation interval, the node will initiate a request to switch to TDMA using the GCP. When all its neighbor nodes receive the request, they vote on whether to switch to the proposed MAC protocol based on their local throughput performance. As shown in Figure 2, node A, C and Receiver all agree to switch to TDMA and node B votes to keep using CSMA. In the end, the TDMA MAC will be switched on for all the nodes based on the majority's voting decision.
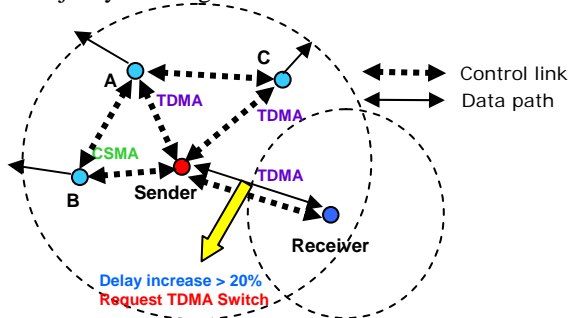

Figure 2: A-MAC scheme in Cognitive Radio Network

V. EXPERIMENTAL EVALUATION ON GNU RADIO/ORBIT

In view of host processing latency and unpredictable USB delay in the GNU radio/USRP platform, we choose basic CSMA and TDMA modes for validation of the AMAC protocol. In the current experiment stage, we use only four nodes with dual radios (GNUradio + WiFi) for proof-of-concept tests. Additionally, to focus on the MAC switching evaluation, we only demonstrate results with MAC switching without PHY adaption.

A. Experiment Setup

We conducted experiments on AMAC with CSMA to TDMA switching on the GNU radio nodes in ORBIT testbed. The GCP and data plane are implemented with a dual-radio structure by using a separate control and data radio in each node. The GCP is implemented by 802.11b radios operating at 2.4GHz (which are available in addition to the GNU radio on every ORBIT testbed node) and the data plane is a GNUradio operating at 400MHz. Radio parameters for GNUradio are specified in Table III. Fig. 3 depicts the 802.11b - GNUradio node structure and the network topology. The network scenario includes four dual-radio wireless nodes. Each node has same inter-node distances (an average of 60 feet) and same radio configurations. In order to represent the worst-case interference scenario, we let node pair (1, 4) and (2, 3) represent as two different data transmission flows.

In the case of AMAC CSMA/TDMA mode, we implement basic CSMA and coarse-grained TDMA protocol because of limitations to GNU radio timing control. In the CSMA protocol, when the sending node senses carrier, it delays 1ms as baseline and then implements exponential back-off while continuing to sense the carrier. In the TDMA

protocol, we let each node synchronize to a central node and design the time frame based on the packets round-trip time. For AMAC, all the nodes first use CSMA protocol as baseline MAC and then decides the suitable MAC protocol for the network based on their local decisions after data transmission initiated. Using the voting procedure, the node requesting MAC change will collect all the votes and announce the final decision.
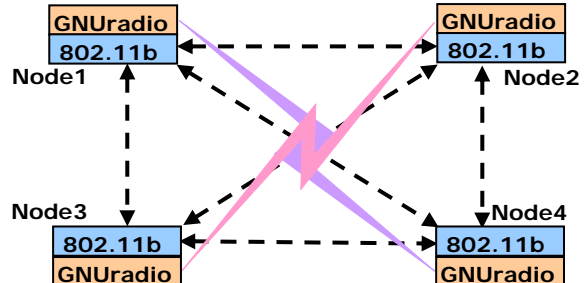

Figure 3: Experimental network topology for A-MAC evaluation

	DATA PLANE (GNURADIO)
PHY TYPE	GMSK
OPERATING FREQ	400MHz
PHY RATE	50kbps
MAC TYPE	CSMA/TDMA
TRANSPORT PROTOCOL	(1). UDP with CBR (25, 50, 75Kbps and packet size of 500B and 1500B) (2). TCP (3) UDP with CBR 25Kbps and mix packet size of 100B and 1500B

TABLE III: Data plane radio parameters (GNU radio)

B. Experimental Results

(1) Baseline evaluation with static traffic

The AMAC algorithm outlined before is evaluated using different traffic types (i.e. UDP and TCP) and different switching thresholds. Since the physical bit rate of the GNU radio is 50kbps, we offer three different UDP traffic loads (25kbps, 50kbps and 75kbps) to illustrate CSMA, TDMA and AMAC performance. For TDMA, we choose two possible time slots for two senders and each slot is 300ms and 100ms for 1500B packet size and 500B packet sizes respectively. For this experiment, we use throughput as the switching threshold. If the performance drops more than 20% in observation five packet intervals, the sending node will request MAC switching. In our experiment, we let nodes 1 and 2 transmits UDP packets to node 4 and 3 respectively at the same time in order to examine the performance of AMAC in the worst-case contention scenario.

Average AMAC throughput measurements for the given 4-node topology are shown in Fig 4. We obtain the average throughput of UDP transmission for both small packet size (500B) and large packet size (1500B). The average throughput is computed from the number of packets successfully transmitted over a window of 100 packets. Three different

PHY bit-rates, 25, 50 and 75 Mbps are shown for comparison. The results show that in this experiment, both TDMA and CSMA have similar performance with CSMA giving slightly higher throughput for node pair 1 and TDMA for node pair 2. Use of AMAC appears to bring up the performance to the higher value for both node pairs. Even though the performance differences are small, the experiment confirms the fact that adaptive MAC switching can provide the upper bound of performance across both MAC protocols. For the results in Fig 4, it is observed that the percentage throughput goes down with the increase of PHY bit-rate because the relative proportion of MAC overhead goes up with the PHY rate for both TDMA and CSMA as implemented.

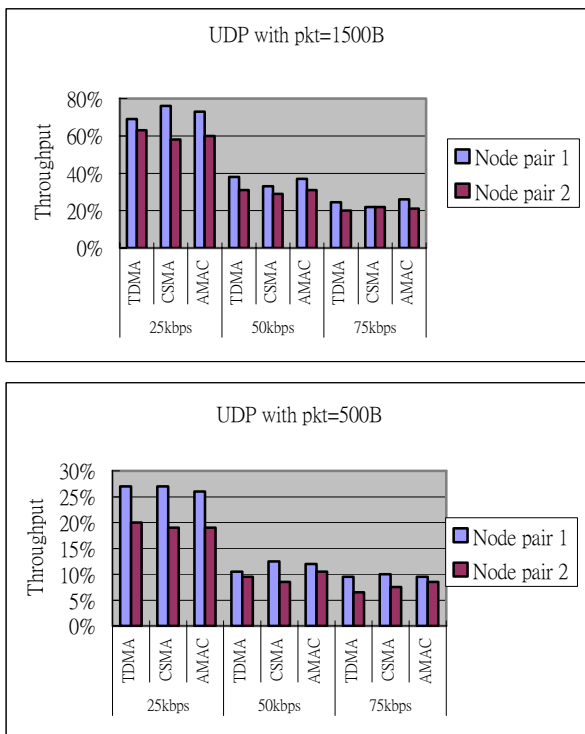UDP with pkt=1500B

UDP with pkt=500B

Figure 4: Compare average throughput with smaller and larger packet size

In Fig. 5, we show how AMAC responds to changes in UDP traffic with 25kbps CBR traffic load. The results for network throughput vs. time (proportional to number of transmitted packets) are compared for CSMA, TDMA and AMAC. When both pairs of nodes initiate data transmission, CSMA starts to perform worse and the average throughput goes down to about 10~15 kbps. On the other hand, TDMA can always achieve 17-22 kbps because there is no contention between the node pairs. In the AMAC experiment, the network switches total five times over the observation interval. At first, nodes discover throughput drops due to contention so they switch to TDMA. After the 65th packet, in Fig. 5, TDMA performance drops because of poor radio activity, it then switches to CSMA until the 95th sent packet. Around the 125th sent packet, the nodes switch to CSMA and to TDMA around the 140th sent packet. AMAC shows a good adaptation between CSMA and TDMA in order to maintain performance.
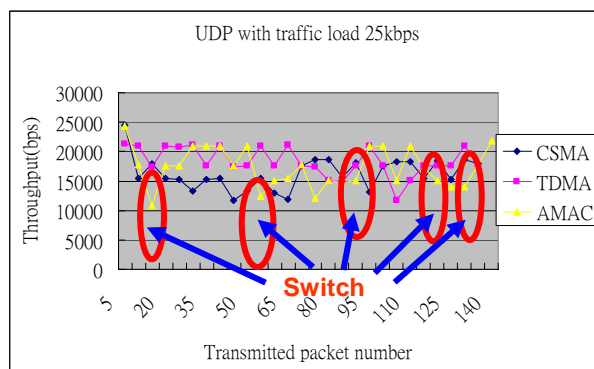

UDP with traffic load 25kbps

Figure 5: Compare average throughput with successfully transmitted packets

(2) Experiments with realistic dynamic traffic

In order to quantify the effect of AMAC on practical applications better, we present a file exchange scenario over TCP protocol. The TDMA time slot for these four nodes is set up as 300ms because of GNUradio processing latency (~50ms) and packet round trip time (~250ms). Each node still starts with CSMA protocol and detects its application throughput during 10seconds observation interval. If it discovers the throughput drops 20% compared to the throughput of the last interval, a MAC switching request will be initiated and exchanged on the GCP between all the senders and receivers.
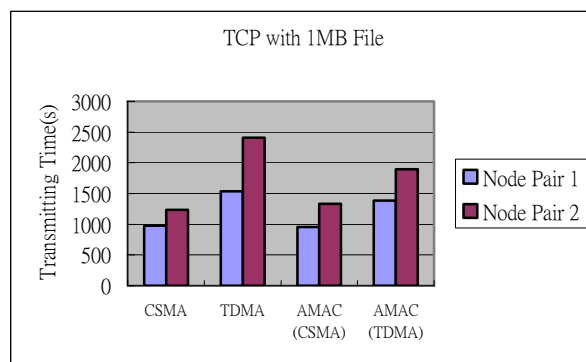

TCP with 1MB File

Figure 6: Compare transmission time by using CSMA, TDMA and AMAC

In Fig. 6, we show the results of using TCP to transmit a 1MB file between these two pairs of nodes. Since TDMA has to have a large time slot and only two pairs of nodes initiate data transmission, the performance of TDMA is worse than CSMA due to the inefficient use of a whole slot taken by TCP ACK packets. Based on the voting strategy, we define AMAC (CSMA) as CSMA favorable mode which means the whole network switches to CSMA as long as at least two nodes decide to use CSMA as their protocol. Similarly, AMAC (TDMA) means the network will apply TDMA when at least two nodes choosing TDMA protocol. In Fig. 6, the file transfer times are compared with different algorithm. It is obvious that AMAC algorithm reduces collision by switching to different MAC so no matter which MAC protocol AMAC favors, it is still capable of finding the better MAC to apply.

(3) *Experiments with dynamic applications*

AMAC is evaluated with dynamic applications using a mix of short messages and streaming service. In Fig. 7, we

present the results while long packets (1500B) are followed by short packets (100B). Each sending node starts with CSMA as baseline MAC and switches to TDMA if it detects the average future packets buffer size is more than 800B, which is the approximate value determined by the half of sum of different packet sizes. AMAC performs closely to and even slightly better than TDMA, because it uses CSMA first for short packets and then switches to TDMA for long packets.

The overhead involved in AMAC from the previous three sets of experiments is calculated in Table IV. The payload overhead ratio means the control overhead is divided by the total payload transmitted. Based on the overhead ratio, we can estimate how much GCP bandwidth will be used by the control packets. We compare the payload overhead with different traffics and network conditions. It is observed from the results that the overhead introduced by the MAC switching is relatively small, typically <<1%. Of course, this number will increase as the number of nodes in the network increases, but as shown in [4] it is possible to aggregate GCP control packets to prevent exponential increase in network overhead.
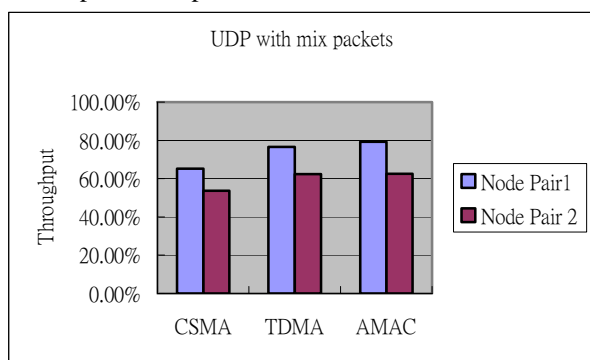

Figure 7: Compare average throughput when traffic with mixed short and long packets is used

Experiment Scenario	Node Pair 1 % control overhead	Node Pair 2 % control overhead
TCP AMAC (CSMA)	0.1634% (1636B)	0.184% (1840B)
TCP AMAC (TDMA)	0.2068% (2070B)	0.246% (2462B)
UDP AMAC (25kbps load) (1500B pkt size)	0.408% (1240B)	0.1768% (521B)
UDP AMAC (25kbps load) (500B pkt size)	0.255% (240B)	0.111% (104B)
UDP AMAC (25kbps load) (Mix of 100B and 1500B pkt size)	0.029% (116B)	0.02% (80B)

TABLE IV: AMAC overhead from GCP control traffic

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have experimentally studied the adaptive MAC (AMAC) protocol using the GCP-based control framework for cognitive radio networks. Our proof-of-concept implementation with GNU radios on ORBIT shows that it is possible to implement dynamic MAC switching in networks of cognitive radios using the capabilities of the control plane protocol. Experimental results with small networks show that

MAC switching can provide performance improvements in dynamic traffic environments, and that switching latency and control overhead are not excessive.

For future work, we plan to conduct more realistic experiments with larger networks of ~10-20 cognitive radio nodes instead of the ~2-5 GNU radios used in the current set of experiments. We also plan to model dynamic vehicular and tactical network scenarios more accurately to assess system level performance gains from adaptive MAC. A more advanced protocol switching algorithm that takes into account multiple factors such as application performance, network congestion and cross-layer protocol interactions is also being investigated. also In addition, we are working on an improved TDMA implementation (jointly with Prof. Steenkiste of CMU) on the GNU/USRP2 platform with improved timing and FPGA support for MAC operations [11].

REFERENCE

[1] J. Mitola III, "Cognitive Radio: An Integrated Agent Architecture for Software Radio," PhD thesis, Royal Institute of Technology (KTH), Sweden, May 2000.

[2] D. Maldonado, B. Le, A. Hugine, T. W. Rondeau, C. W. Bostian, "Cognitive radio applications to dynamic spectrum allocation: a discussion and an illustrative example," *IEEE Proc. DySPAN*, 8-11 Nov. 2005, pp. 597 - 600.

[3] Jun Zhao, Haitao Zheng and Guang-Hua Yang, "Distributed Coordination in Dynamic Spectrum Allocation Networks" in *IEEE DySPAN*, Nov. 2005.

[4] Xiangpeng Jing and Dipankar Raychaudhuri, "Global Control Plane Architecture for Cognitive Radio Networks," *in Proceedings of IEEE CogNets 2007 Workshop - Towards Cognition in Wireless Networks (in conjunction with IEEE ICC)*, June 24-28, 2007.

[5] Xiangpeng Jing, Shanmuga S Anandaraman, Mesut Ali Ergin, Ivan Seskar and Dipankar Raychaudhuri, "Distributed Coordination Schemes for Multi-Radio Co-existence in Dense Spectrum Environments: An Experimental Study on the ORBIT Testbed," *in Proceedings of IEEE DySPAN 2008*, Chicago, IL, Oct. 14-17, 2008.

[6] D. Raychaudhuri, "ORBIT: Open-Access Research Testbed for Next-Generation Wireless Networks," NSF Network Research Testbeds Program, NSF award #ANI-0335244, 2003-07, May 2003.

[7] Xiangpeng Jing and D. Raychaudhuri, "A Spectrum Etiquette Protocol for Efficient Coordination of Radio Devices in Unlicensed Bands," *in Proceedings of PIMRC 2003*, Vol. 1, pp. 172 - 176, Beijing, China, Sept. 7-10, 2003.

[8] Cordeiro, Carlos; Challapali, Kiran, "C-MAC: A Cognitive MAC Protocol for Multi-Channel Wireless Networks" in *IEEE DySPAN*, Apr. 2007

[9] Doerr, C., Neufeld, M., Fifield, J., Weingart, T., Sicker, D.C., Grunwald, D, "MultiMAC - an adaptive MAC framework for dynamic radio networking" in *IEEE DySPAN*, Nov. 2005

[10] Rahul Dhar, Gesly George, Amit Malani, and Peter Steenkiste, "Supporting Integrated MAC and PHY Software Development for the USRP SDR", in *IEEE* Workshop on Networking Technologies for Software Defined Radio (SDR) Networks, Sept. 2006.

[11] Enabling MAC Protocol Implementations on Software-defined Radios, George Nychis, Thibaud Hottelier, Zhuochen Yang, Srinivasan Seshan, Peter Steenkiste, Networked Systems Design and Implementation, 2009.

[12] K. Mandke, Soon-Hyeok Choi, Gibeom Kim, R. Grant, R. C. Daniels, Wonsoo Kim, Robert W. Heath Jr, and S. Nettles, "Early Results on Hydra: A Flexible MAC/PHY Multihop Testbed", in the *Proc. of the IEEE Vehic. Tech. Conference*, Dublin, Ireland, April 23 - 25, 2007.