# OSU-MAC: A New, Real-Time Medium Access Control Protocol for Wireless WANs with Asymmetric Wireless Links

Chunlei Liu†, Ye Ge‡, Mike Fitz‡, Jennifer Hou‡, Wei-Peng Chen‡, and Raj Jain†
† Department of Computer and Information Science
‡ Department of Electrical Engineering
The Ohio State University
Columbus, OH 43210
*{cliu, jain}@cis.ohio-state.edu, {gey, jhou, fitz, chenwe}@ee.eng.ohio-state.edu*

## Abstract

*In this paper, we document our design of a MAC protocol, called* OSU-MAC, *subject to the physical layer characteristics and constraints of a narrow-band wireless modem testbed currently being built at the Ohio State University. The narrow-band wireless modem testbed is expected to support both real-time (bus location tracking) and non-real-time (regular) data applications. A number of techniques are proposed to support QoS imposed by the real-time applications, to deal with the asymmetry on the forward and reverse channels and the half-duplex transmission constraint imposed by the physical layer, and to enhance the error control capability of* OSU-MAC. *We also present simulation results to demonstrate the key, functional characteristics of* OSU-MAC.

*Index Terms — MAC, asymmetric wireless links, temporal QoS, half-duplex transmission constraints.*

## 1 Introduction

Wireless communication has become an important technique for supporting emerging Personal Communications Service (PCS). In PCS, both traditional telephone service and other more advanced data applications, e.g., audio/video, and periodic update of sensor information, are expected to be simultaneously supported. The latter applications, in particular, require different levels of temporal quality of service (QoS). An effective medium access control (MAC) protocol must be carefully designed for this purpose. In this paper, we document the design of a MAC protocol, called *OSU-MAC*, that supports both real-time and non-real-time applications on an OSU narrow-band wireless modem testbed, subject to its physical layer characteristics and constraints. The narrow-band wireless modem testbed is expected to support both real-time and non-real-time (regular) data applications, currently with the real-time bus location tracking via an on-board global positioning system (GPS) being the representative, real-time application and data applications such as e-mail, ftp, and telnet, being the other representative.

Our first step toward realizing the above objective is to design and implement an effective MAC protocol, *OSU-MAC*, that coordinates the transmission activities on the forward and reverse channels so as to support QoS imposed by both types of applications. In particular, we delegate to the base station the full responsibility of resource arbitration, channel access, and registration in each cell. As will be elaborated on in Section 3.1, this base- station-based scheduling approach enables provisioning of deterministic QoS for real-time applications, while maximizing the system utilization. To take the error characteristics of the physical layer into consideration, we encode both data packets and control fields in Reed-Solomon code [1, 2]. We also take into account the half-duplex transmission constraint (i.e., a mobile subscriber cannot transmit and receive at the same time) imposed by the physical layer, and propose a two-control-field structure to fully utilize the limited bandwidth available on the reverse channel. To make use of the unused bandwidth originally reserved for real-time traffic, we also propose a dynamic slot adjustment scheme. Finally, as a real-life MAC protocol currently being implemented on the OSU narrow- band wireless modem testbed, we insert, wherever advised by the wireless modem researcher, preambles, postambles, and guard times between packet slots or notification cycles for synchronization.

Several MAC protocols have been proposed in wireless networks, among which Packet Reservation Multiple Access (PRMA) [3], Dynamic TDMA (D-TDMA) [4], Dynamic Reservation Multiple Access (DRMA) [5], Resource Auction Multiple Access (RAMA) [6], Floor Acquisition Multiple Access (FAMA) [7], Remote-Queuing Multiple Access (RQMA) [8], and Multimedia Cable Network System (MCNS) [9] may have received the most attention. They have been designed either for voice/data traffic (PRMA, D-TDMA, DRMA, RAMA, and FAMA), for real-time/best-effort traffic on an abstract model (RQMA), or for cable modem users (MCNS). In comparison with the above research efforts, OSU-MAC is the first implementation work that (i) takes into account of the physical layer characteristics and constraints on a specific environment in the design phase (rather than an abstract design), (ii) provides mechanisms for providing deterministic, temporal QoS for real-time applications, and (iii) is fault tolerant with both data slots and control fields protected by the (64,48) Reed-Solomon code.

The rest of the paper is organized as follows. In Section 2, we introduce the design objectives of, the types of applications to be supported on, and the system model used for, the OSU narrow-band wireless testbed. We also outline the physical layer characteristics and constraints of the wireless testbed. These characteristics and constraints are the major factors in directing the design of the proposed MAC protocol. In Section 3, we present *OSU-MAC*. In Section 4, we give a survey of existing MAC protocols for wireless local area net-

---

The work reported in this paper was supported in part by NSF under Grant No. ANI989018. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agency.

works. In Section 5, we evaluate *OSU-MAC* in terms of throughput, packet delay, capability of providing temporal QoS, collision probability, and registration latency. We conclude the paper with Section 6.

## 2 The OSU Narrow-Band Wireless Testbed

### 2.1 Applications supported and design goals

**Applications supported:** Two types of applications will be supported: one is real-time bus location tracking via an on-board global positioning system (GPS). In the GPS system, short GPS packets of size no more than 72 bits are periodically sent from each bus being tracked to report the location of the unit. Timely delivery of these packets is important in correctly tracking the bus location. The other applications to be supported are those supported by TCP/IP, e.g., e-mail message retrieval, FTP, and Telnet.

**Design objectives:** In the current narrow-band wireless modem testbed, each base station covers an area about 10 km in radius and is expected to support

- up to 8 active GPS users with 1 minute checking delay and 4 second access delay requirements, where the checking delay is the delay incurred when a non- active terminal becomes active, and the access delay is the time interval between the arrival of a packet at an active GPS user and its transmission.

  The 4 second access delay requirement is determined as follows: since a mobile unit moves at a speed no faster than 90 kilometers per hour and the allowable location error is within 100 meters, a GPS packet must be sent and received every $100/(90 \times 10^3)$ hours, or 4 seconds. If a GPS packet is corrupted after being decoded, the corrupted packet will not be retransmitted.
- up to 64 active non-real-time users. No access delay requirement is imposed, but data packets need to be reliably transported.

By active users, we mean mobile subscribers which have registered with the base station and are actively transmitting/receiving data packets. The maximum time required to register with the base station is also a design parameter. In the current design, we require that 80% of the registration requests can be approved in two notification cycles, and 99% can be made in 10 cycles.

### 2.2 System model

The geographical area covered by a wireless network is divided into overlapping cells. Each cell is associated with a base station and can support a number of mobile subscribers. The base station is connected to one another to form a wired point-to-point backbone network, handles all the ongoing activities in the cell, and has the overall control of the system resources. Signals from the base station are broadcast to all mobile subscribers. Signals sent from mobile subscribers are, however, only received by the base station and not heard by other mobile subscribers.

Each base station is allocated a number of frequencies (termed as channels or links) on which transmission takes places. At any time instant, only one station/subscriber can transmit on a channel; otherwise, collision occurs and all the ongoing transmissions on that channel fail. The channel used by the base station to transport data to mobile subscribers is called the *forward* channel, while that used by mobile subscribers to transport data to the base station is called the *reverse* channel. Signals on different forward/reverse channels are independent of one another. In the current OSU narrow-band wireless modem testbed, there are one forward channel and one reverse channel in each cell.

### 2.3 Physical characteristics of forward/reverse channels

The actual symbol rate on the forward channel is currently 3200 channel symbols per second. With a quaternary PSK modulation (QPSK) transmitting two coded bits per channel symbol, the forward channel can operate up to 6.4 kbps. The reverse channel supports approximately half the transmission rate of the forward channel in order to maintain a margin of path balance and allow time division multiple access. The actual symbol rate on the reverse channel is currently 2400 symbols per second. With the use of a PSK modulation transmitting two information bits per channel symbol, the reverse link can operate at a rate of 4.8 kbps.

**Structure of pilot symbol frame:** Transmission on both the forward/backward channels is broken into cycles of approximately 4 seconds in length. (The reason for choosing this cycle length will be given later.) Each notification cycle is further broken up into pilot symbol (PS) frames which have a size of 150 channel symbols. The structure of the PS frames is shown in Fig. 1 and described below: Fifteen pilot symbols are inserted periodically every 10 symbols starting at the first symbol of the PS frame. In addition, 7 additional pilots are inserted at the beginning of each PS frame for a total of 22 pilot symbols per PS frame. The pilot symbol are used by the mobile unit to estimate the fading distortion and the resulting frequency offset. Each PS frame contains 128 channel symbols for the coded information bits, and hence the transmission efficiency of each PS frame is $128/150 = 85.3\%$.

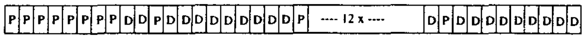


Figure 1: Pilot symbol frame, P=pilot symbol, D=data symbol.

Each regular (non-real-time) data packet is 64 bytes (512 bits) long, 48 bytes (384 bits) of which are information bits. Since a PS frame contains 128 non-PS symbols (256 bits), each data packet is transmitted in two PS frames.

**Error correction with Reed-Solomon code:** For the purpose of error correction, packets transported on forward/reverse channels are encoded in Reed-Solomon code, RS(48,64), over GF(256). Our experience with wireless errors from field tests for this RS code design indicates that two events occur with extremely high probability: (i) a small number of errors occur and are corrected; and (ii) a large number of errors occur and the RS decoder fails to provide an output. Consequently it is extremely rare that a packet is delivered with an error. It is either delivered error free or the delivery fails (the latter is considered a packet loss by the base station and mobile subscribers because of lack of acknowledgments).

**Insertion of preambles and guard times for synchronization:** On the forward channel, a preamble is inserted at the beginning of each notification cycle to allow a mobile subscriber to synchronize to the master timing of the base station. The preamble will contain a unique word to synchronize the forward and reverse channels once every notification cycle. On the reverse channel, each non-real-time data packet is proceeded by a packet preamble of 600 symbols, followed by the packet body and a packet postamble of 51 symbols. Non-real-time packets are also separated from each other by a guard time of 0.0075 second (18 symbols). On the other hand, each GPS data packet is proceeded by a packet preamble of 64 symbols and separated from each other by a guard time of 0.0075 second (18 symbols). Table 1 summarizes these time parameters.

**Packet size:** On the forward channel, since each regular data packet is transmitted in two PS frames, it takes $300/3200 = 0.094$ seconds to transmit a data packet. On the reverse channel, it takes $300/2400 = 0.125$ second to transmit a non-GPS data packet. Together with the time to transmit the preamble and postamble ($651/2400 = 0.27125$ second) and the guard time ($18/2400 = 0.0075$ second), each data slot for transmission of non-GPS packets is set to 0.40375 second. On the other hand, it takes $128/2400 = 0.05333$ second to transmit a GPS packet. Together with the time to transmit the preamble ($64/2400 = 0.02667$ second) and the guard time ($18/2400 = 0.0075$ second), each data slot for transmission of GPS packets is set to 0.0875 second. Table 1 lists the parameters that characterize the physical layer characteristic and pertain to the MAC protocol design.

## 2.4 Constraints imposed by physical layer characteristics:

**Half duplex transmission constraint:** In the current narrowband wireless testbed, the base station has a transmitter and a receiver, and can listen and transmit at the same time. However, because of the power and transmitter/receiver constraints, mobile subscribers can only transmit or receive but not both at any one time. Moreover, a 20 ms guard time has to be inserted between switch-over from the transmit function to the receive function and vice versa. This implies that (i) a mobile subscriber cannot transmit 20 ms before or after its receiving period, and (ii) slots that carry packets destined to a mobile subscriber $M$ on the forward channel must be apart from those scheduled to transport $M$'s packets on the reverse channel by at least 20 ms. This is termed as the *half duplex transmission* constraint. Several important design decisions of the proposed MAC protocol have been driven by this physical layer constraint.

**Asymmetry between the forward/reverse channels:** Another physical layer characteristic that drives the design decision is the asymmetry between the forward and reverse channels. First, the base station can transmit with stronger power than mobile subscribers, and hence the forward channel is usually more reliable than the reverse channel. As a result, data packets transmitted on the reverse channel have to be proceeded by packet preambles and separated with guard time. Second, because of the physical layer characteristics (e.g., difference in the symbol transmission rate and the modulation schemes, and the necessity of packet preamble/postamble/guard time on the reverse channel), the reverse channel has comparatively much less bandwidth than the forward channel. Third, as a result of the half duplex transmission constraint, mobile subscribers cannot be scheduled to transmit and receive at the same time, and proper guard times have to be inserted between the switch-over of transmit and receive functions. Finally, without a centralized control facility, mobile subscribers may compete for channel access on the reverse channel, sometimes on a contention basis, and hence the access delay on the reverse channel is longer and more variable.

# 3 Proposed MAC Protocol – OSU-MAC

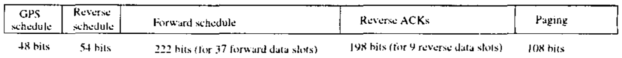
## 3.1 Base station-centric resource arbitration

A major feature of our proposed MAC protocol is that we delegate to the base station the full responsibility of resource arbitration, channel access, and registration in each cell. The reasons for this design are two-fold: first, as the base station in a cell usually has the overall control over all the system resources, it is reasonable to delegate the base station to arbitrate the assignment of data slots on both the forward and reverse channels. Second, because of the asymmetry between the forward and reserve channels, the MAC protocol should be so designed as to include as little control overhead on the reverse channel as possible. That is, the control information sent from mobile subscribers to the base station should be kept minimal. This leads to a base station-centric mechanism. The only control information sent uplink is the registration and slot reservation requests.

Specifically, the base station transports data packets as well as channel access information on the forward channel to mobile subscribers. Channel access information includes, among other things,

- the slot access schedule on the forward channel and on the reverse channel for the current notification cycle. In particular, the slot access schedule on the reverse channel is determined by the reservation requests received on the reverse channel in the previous notification cycle.
- acknowledgment for packets received by the base station on the reverse channel.
- information used to page inactive mobile subscribers.

**Control fields:** Each mobile subscriber has a permanent, universally unique equipment identification number (EIN) of 16 bits. In addition, a mobile subscriber is assigned a user ID of 6 bits when it registers with the base station. This 6-bit user ID is unique only within the cell, and will be used solely by the base station to specify/identify a mobile subscriber. A set of explicit control fields on the forward channel is used for the base station to convey the above channel access information to mobile subscribers. There is no explicit control field on the reverse channel. All the control information sent uplink is either carried in the header of data packets or included in regular data packets (i.e., the in-band signaling approach is used).



| | Forward channel |
|---|---|
| GPS schedule (bits) | 48 |
| Reverse schedule (bits) | 54 |
| Forward schedule (bits) | 222 |
| Reverse ACKs (bits) | 198 |
| Paging (bits) | 108 |
| Total size of control fields (bits) | 630 |
| Required RS codewords for one set of control fields | 2 |
| Required PS frames for one set of control fields | 4 |
| Required time for one set of control fields (seconds) | 0.1875 |

Figure 2: The control fields on the forward channel.

The control fields consist of following information (Fig. 2):

- GPS schedule: gives the user IDs of (up to) 8 GPS users which are scheduled to use the 8 GPS slots on the reverse channel. This field is $8 \times 6 = 48$ bits.

|  | Fwd channel | Rev channel |
|---|---|---|
| **General physical layer characteristics** |  |  |
| Chan. symbol rate (symbols per second) | 3200 | 2400 |
| Coding rate (coded bits/symbol) | 2 | 2 |
| Information symbols in a pilot frame | 128 | 128 |
| Chan. symbols in a pilot frame | 150 | 150 |
| Information bits per RS (64,48) codeword | 384 | 384 |
| Bits per RS (64,48) codeword | 512 | 512 |
| **Packet size** |  |  |
| RS codewords per packet | 1 | 1 |
| Pilot frames per regular data packet | 2 | 2 |
| Chan. symbols per regular packet | 300 | 300 |
| Time per regular packet (second) | 0.09375 | 0.125 |
| **Cycle preamble** |  |  |
| Cycle preamble length (chan. symbols) | 450 | n/a |
| Time per cycle preamble (seconds) | 0.140625 | n/a |

| **Packet parameters on reverse channel** |  |  |
|---|---|---|
|  | GPS | Regular |
| Packet size (information bits) | 72 | 384 |
| Packet size (chan. symbols) | 128 | 300 |
| Packet preamble (chan. symbols) | 64 | 600 |
| Packet preamble (seconds) | 0.02667 | 0.25 |
| Packet postamble (chan. symbols) | 0 | 51 |
| Packet postamble (seconds) | 0 | 0.02125 |
| Packet guard time (chan. symbols) | 18 | 18 |
| Packet guard time (seconds) | 0.0075 | 0.0075 |
| Total length (chan. symbols) | 210 | 969 |
| Total length (seconds) | 0.0875 | 0.40375 |

Table 1: List of parameters in the physical layer that pertain to the MAC design.

- **Reverse schedule**: is used by the base station to announce the slot schedule in response to the reservation requests received on the reverse channel, either *explicitly* or *implicitly*[1], in the previous notification cycle. It gives the user IDs of (up to) $M$ data users scheduled to use the data slots on the reverse channel. In our current design[2] $M = 9$, and hence this field is $9 \times 6 = 54$ bits.
- **Forward schedule**: is used to inform mobile subscribers to which subscriber data slots on the forward channel will be transmitted. It gives the user IDs of mobile subscribers which should receive data on the $N$ data slots on the forward channel. In our current design [3] $N = 37$, and hence this field is $37 \times 6 = 222$ bits.
- **Reverse ACKs**: are used to acknowledge receipt of data packets on the reverse channel in the previous notification cycle or to notify a mobile subscriber (whose registration request is approved) of its (EIN, user ID) pair. This field is $9 \times (16 + 6) = 198$ bits.
- **Paging**: is used to page and locate inactive mobile subscribers. To support paging of up to 18 users, this field contains $18 \times 6 = 108$ bits.

The total length of these control fields is 630 bits, which requires 2 RS codewords to carry. Note that out of the 768 informational bits available in the 2 RS codewords, 138 bits are reserved for future use. All mobile subscribers have to listen to the control fields on the forward channel in order to find out their scheduled access time to the channels.

**Slot reservation and scheduling:** For real-time, GPS applications transported uplink, we use reservation-based scheduling to ensure the QoS required. When a mobile subscriber with GPS applications registers with the base station, it will be assigned GPS slots properly spaced on the reverse channel until the mobile subscriber signs off. The GPS slots will be so assigned that at least one GPS slot in any time interval of 4 seconds is assigned to the GPS subscriber.

To allow mobile subscribers with regular, non-real-time data to gain access to the reverse channel, one or more data slots on the reverse channel are designated as *contention* slots in each notification cycle, where a contention slot is simply a data slot *not* assigned to any mobile subscribers on the reverse channel. There are three possible means to reserve data slots on the reverse channel:

1. A mobile subscriber may explicitly send a reservation request packet, specifying the number of data slots desired, on one of the contention slots.
2. When a mobile subscriber transmits its data packets in the data slots assigned to it in a notification cycle, it may set a reservation field in the header of the data packet to implicitly indicate that it would like to request more data slots in the next notification cycle.
3. A mobile subscriber may send its data packet in one of the contention slots and compete with the other mobile subscribers with data or reservation packets on a contention basis. If multiple mobile subscribers attempt to transmit their data/reservation packets in the same slot, collision occurs. (The base station has to explicitly acknowledge on the forward channel receipt of data packets on the reverse channel as a result of these potential collisions in contention slots.)

When collision occurs, mobile subscribers back off with a random period of time before their subsequent attempts. (To increase the probability of successful reservation, mobile subscribers that transmit data packets without reservation are required to back off with a longer time period.) Also, if collisions occur multiple times in a notification cycle or across multiple notification cycles, the base station may designate more than one data slots as contention slots (i.e., leave them unassigned) in the next cycle. On the other hand, if multiple contention slots have been left unused in the current cycle, the base station may decrease the number of contention slots in the next cycle.

The base station notifies a mobile subscriber that makes a reservation request or transmits its data in a contention slot of whether or not the request/data has been received by indicating in the $i$th **reverse ACKs** field the user ID of the subscriber whose request/data has been received. Note that the fact that a reservation request is received does not imply the request will be honored. A mobile subscriber has to look into the **reserve schedule** field to find out whether or not it is assigned data slots.

After the base station "collects" all the requests in the current notification cycle, it uses a specific scheduling algorithm (in our current design, the round robin algorithm) to determine the slot schedule on the reverse channel, subject to the half-duplex transmission constraints. The resulting slot schedule is then announced in the **reverse schedule** control field in the next notification cycle. Mobile subscribers will then transmit their data accordingly.

## 3.2 Registration of mobile subscribers

A mobile subscriber registers itself with the base station (also) through the use of contention slots. A mobile subscriber that newly enters the cell first listens to the forward channel to synchronize itself on the reverse channel and to find out the positions of contention slots. Then it transmits its registration request in one of the contention

---

[1] To be discussed below.

[2] The reason why $M = 9$ will be given in Section 3.3.

[3] The reason why $N = 37$ will be given below.

slots. The registration request packet may compete with other registration/reservation/data packets. If a collision results, the mobile subscriber with the registration request *persists* in the next notification cycle until it succeeds in one notification cycle or fails after a pre-determined number of attempts. Note that we give the priority of using contention slots to mobile subscribers which intend to register themselves with the base station, as other mobile subscribers with reservation/data packets will back off in the case of collision.

If a registration request made in the $i$th contention slot is successfully received by the base station, it will be passed to the registration handling module for approval. If the registration request is approved, the base station will notify the requesting subscriber by returning the (EIN, user ID) pair in the $i$th reverse ACKs field.

### 3.3 Structure of notification cycle on the reverse channel

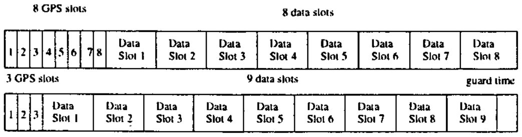


Figure 3: Two formats of the notification cycle on the reverse channel.

The structure of the notification cycle on the reverse channel is shown in Fig. 3. Depending on the number of active GPS subscribers, the system can choose one of the two possible formats. If there are more than three active GPS subscribers, the system uses the first format. In this format, 8 GPS slots are scheduled first, followed by 8 data slots. The second format is used when the number of active GPS subscribers is less than or equal to 3. In this case, five unused GPS slots are combined to form a data slot (to be used by data users). The notification cycle begins with 3 GPS slots, followed by 9 data slots and a guard time of 0.03375 second.

| | **Format 1** | **Format 2** |
|---|---|---|
| GPS slot 1 | 0.30125 | 0.30125 |
| GPS slot 2 | 0.38875 | 0.38875 |
| GPS slot 3 | 0.47625 | 0.47625 |
| GPS slot 4 | 0.56375 | — |
| GPS slot 5 | 0.65125 | — |
| GPS slot 6 | 0.73875 | — |
| GPS slot 7 | 0.82625 | — |
| GPS slot 8 | 0.91375 | — |
| Data slot 1 | 1.00125 | 0.56375 |
| Data slot 2 | 1.40500 | 0.96750 |
| Data slot 3 | 1.80875 | 1.37125 |
| Data slot 4 | 2.21250 | 1.77500 |
| Data slot 5 | 2.61625 | 2.17875 |
| Data slot 6 | 3.02000 | 2.58250 |
| Data slot 7 | 3.42375 | 2.98625 |
| Data slot 8 | 3.8275 | 2.98625 |
| Data slot 9 | — | 3.39000 |

Table 2: Reverse channel access time of the two formats.

Since the base station knows how many active GPS subscribers are in the system, it is the base station's responsibility to choose and announce which format to use. The announcement is made implicitly through the number of GPS subscribers in the control fields. If the number is greater than 3, then the base station and all mobile subscribers will use format 1, otherwise, they use format 2. By using a format, we mean the mobile subscribers will synchronize themselves to the beginning of each notification cycle and access the reverse channel according to the time given in Table 2.

**Dynamic GPS slot adjustment on the reverse channel:** As mentioned in Section 2.1, GPS units report the bus location once every 4 seconds. A naive approach to fulfill this real-time requirement is to statically allocate the same GPS slot in each notification cycle to a registered GPS user. This approach, although simple, may result in bandwidth waste. This is because as GPS users register and later sign-off, some of the GPS slots may be allocated and then released, creating holes between allocated GPS slots. For example, if GPS users 1 to 8 registered and assigned GPS slots in order. Later, users 2, 3, 5, 6, 7 left the system, creating two holes slots 2–3 and slots 57. These holes cannot be used by data users, even if the number of GPS users is less than 3.

A more sophisticate approach is to dynamically adjust GPS slots to consolidate allocated GPS slots and then combine unused GPS slots into data slots. If there are more than three GPS users, the system uses format 1. When GPS users leave, GPS slots are re-assigned to existing GPS users and unused GPS slots converted into a data slot, *subject to the real-time requirements of existing GPS users.* If more GPS users register later, this data slot can be split into five GPS slots again. The real-time requirements of existing GPS users are ensured through the following rules of slot re-assignment:

**(R1)** The GPS slots in a cycle are allocated in order.

**(R2)** When a GPS user is admitted into the system, it is allocated the first unused GPS slot.

**(R3)** When a GPS user assigned GPS slot $i$ leaves the system, the GPS user that uses GPS slot $j > i$ (if any) is re-assigned slot $i$.

Note that with **(R3)**, when a GPS user is re-assigned a slot, it is ensured to has an slot access interval that is less than 4 seconds in the current notification cycle and hence the real-time requirement is fulfilled. Also, with these rules, allocated GPS slots are consolidated at the beginning of each notification cycle, and unused GPS slots can be converted into a data slot.

**The number of regular data slots on the reverse channel:** Given that (1) each notification cycle is approximately 4 seconds long, (2) there are at most 8 GPS slots, each of length 0.0875 second, and (3) each non-real-time data slot is 0.40375 second in length (Table 1), the total number of regular data slots is

$$(4 - 0.0875 \times 8)/0.40375 \approx 8.$$

**Guard time:** The exact cycle length on the reverse channel under the above configuration is 3.93 seconds. As will be discussed in Section 3.4, the notification cycle length on the forward channel is 3.9844 seconds. To make the notification cycles on both channels the same, We add a guard time of 0.0544 second on the reverse channel.

### 3.4 Structure of notification cycle on the forward channel

The notification cycle on the forward channel begins with a preamble of 300 symbols. The preamble is then followed by control fields and data slots.

**Two control fields to deal with the half-duplex transmission constraint:** As mentioned above, the base station uses the control fields on the forward channel to announce channel access schedules on both channels, to acknowledge packets received on the reverse channel, and to page inactive mobile subscribers. All mobile subscribers must listen to the control fields to obtain the above information. However, due to the half-duplex transmission constraint imposed by the physical layer, mobile subscribers scheduled to transmit on the reverse channel at the same time of the control fields being transmitted on the forward channel will not be able to listen to the control fields. One straightforward solution is not to schedule any mobile subscriber for transmission on the reverse channel during the period when the control fields are being transmitted on the forward channel. We did not adopt this solution, because it results in a waste of bandwidth on the reverse channel.

As an alternative to deal with the half-duplex transmission constraint, we insert two sets of control fields, with the second one exclusively used by mobile subscribers scheduled to transmit during the time interval when the first control fields are transmitted. In other words, the bandwidth utilization on the reverse channel is improved at the expense of introducing a second set of control fields on the forward channel (which has comparatively more abundant bandwidth).

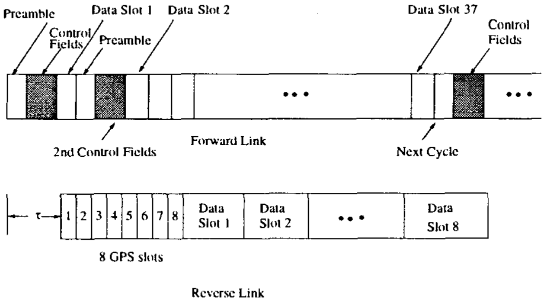


Figure 4: The structure of notification cycles on the forward and reverse channels.

There are three issues that must be resolved before we can fully realize the two control field design:

**The location of the second set of control fields:** One intuitive approach would be to evenly divide data slots into two groups and arrange the preamble, control fields, and data slots in each notification cycle as follows:

| | | | | | |
|---|---|---|---|---|---|
| Preamble | Control fields | Half of the data slots | Preamble | Control fields | Half of the data slots |

The problem with the above arrangement is that the base station cannot use the first half of data slots on the forward channel to send data to the mobile subscribers that listen to the second set of control fields. This is because these subscribers will not know their schedule until they listen to the second set of control fields. Similarly, the reverse data slots that are ahead (in time) of the second set of control fields cannot be assigned to these mobile subscribers either. To deal with this problem, we propose to place the second set of control fields as closely as to the beginning of each notification cycle. As shown in Fig. 4, the notification cycle is structured as the preamble (of size 300 symbols) followed by the first set of control fields (2 RS codewords), one data slot (1 RS codeword), another preamble (of size 150 symbols), the second set of control fields (2 RS codewords), and the rest of data slots. With this configuration, in the worst case (which occurs when the base station only has packets destined for the data user which is scheduled to transmit in the data slot), only one data slot on the forward channel cannot be used.

The reason for not concatenating the two sets of control fields back to back is to ensure that the data user scheduled to transmit in the last reverse data slot completes its uplink transmission and has sufficient times to switch from the transmit function to the receive function.

**The set of control fields a mobile subscriber should listen to:** All mobile hosts need to listen to the control fields in order to synchronize with the base station and to receive channel access schedule. With the two control field design, some users will transmit at the same time of the control fields. How does they know which control field they should listen to? In order to solve this problem, we shift the cycle on the reverse channel $\tau$ seconds later than that on the forward channel (Fig. 4), where

$$\begin{aligned}\tau &= 0.09375(\text{preamble}) + 0.1875(\text{control fields}) + 0.02) \\ &= 0.30125 \text{ seconds.}\end{aligned}$$

The extra 0.02 seconds makes it possible for the GPS users to transmit right after they learn their schedules on the forward channel.

After the shift, the only slot on the reverse channel that overlaps the first control fields in the next notification cycle is the last data slot. The user which is scheduled to transmit in this slot should listen to the second set of control fields. All the other users should listen to the first set of control fields. In summary, mobile subscribers use the following rules to decide which set of control fields it should listen to: (i) When a mobile subscriber first enters the system, it listens to the first control fields; and (ii) If a mobile subscriber is assigned to transmit in the last reverse data slot, it listens to the second set of control fields; otherwise, it listens to the first set of control fields.

Note that the base station must not assign the first slot on the forward channel to the user which listens to the second set of control fields.

**Difference between the first and second sets of control fields?** The only difference between the two sets of control fields is that the second set of control fields has to acknowledge the activity that occurs on the reverse channel when the first set of control fields is transmitted. Specifically,

- If the last data slot on the reverse channel was used to send a data packet, the second set of control fields acknowledges its reception (in the reverse ACKs field).
- If the last data slot on the reverse channel was used by a new mobile subscriber for registration, the second set of control fields announces whether or not the reservation succeeds.

Also, the base station can schedule forward data slots that were announced idle in the first set of control fields to the user assigned the last data slot on the reverse channel, based on whether or not the user requests for more data slots in the packet header of its packet (transmitted in the last slot). However, the base station *cannot* make any change in the reverse schedule.

**The number of data slots per cycle on the forward channel:** The number of data slots that can be transmitted per notification cycle is contingent, among other things, upon the sizes of data slots and notification cycle. Since each data packet is 2 RS codewords long, we decide that each data slot is of size 2 RS codewords (300 symbols with both pilot symbols and Reed-Solomon error check bits considered) as well. Given that (1) the forward channel can transmit 12800 symbols in a 4-second period, (2) the two preambles are totally 450 symbols, (3) the two sets of control fields are 600 symbols (2 RS codewords) each, and (4) each data slot is of size 300 symbols (Table 1 and Fig. 2), the number of data slots available is thus $(3200 \times 4 - 450 - 600 \times 2)/300 \approx 37$. This implies that the exact length of a notification cycle is 3.9844 seconds.

### 3.5 Scheduling constraints and algorithm

As mentioned in Section 3.1, we delegate to the base station the full responsibility of (i) generating slot schedules on both the forward and reverse channels and (ii) handling registration requests in each cell. After introducing the notification cycle structure on both the forward and reverse channels, we are now in a position to delve into the details of how data slots are scheduled on both channels. Since the reverse channel has limited bandwidth than the forward channel, the base station schedules slots on the reverse channel and then assign slots on the forward channel, the latter subject to the half-duplex-transmission and two-control-fields constraints.

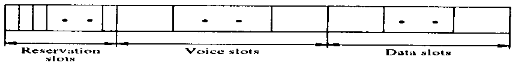
**Generation of slot schedules for data/GPS applications:** Mobile subscribers make their reservation requests for the reverse channel through explicit/implicit reservation or contention. The slot schedule for regular data slots on the reverse channel is then generated using the round robin scheduling algorithm. After the schedule is determined by the scheduling algorithm, the schedule is then re-adjusted to lump slots allocated to a mobile subscriber together so that the subscriber does not have to repeatedly switch between transmitting and sending in a cycle.

After the reverse slots are scheduled, the data slots on the forward channel are allocated in a similar way, but subject to the following constraints: (i) a mobile subscriber cannot be scheduled to transmit on the reverse channel and to receive on the forward channel at the same time; (ii) a mobile subscriber cannot be scheduled to transmit on the reverse channel 20 ms before and after it is scheduled to receive on the forward channel; and (iii) a mobile subscriber cannot be scheduled to receive on the forward channel 20 ms before and after it is scheduled to transmit on the reverse channel.

**Registration and reservation:** In order to allow new users to register and registered users to send reservation requests, the first few data slots in a notification cycle are always left unassigned and used as contention slots. Users can use contention slots to register themselves with the base station or make slot reservation.

To reduce the registration latency (defined as the time interval between the time when a registration is first made and the time it is finally received by the base station), the base station monitors the collision rate in the contention slots. If the rate exceeds some predetermined threshold, the base station dynamically allocates a few more contention slots in the subsequent notification cycles, and vice versa.

(1) Frame structure in PRMA



(2) Frame structure in D-TDMA

Figure 5: Frame structures for PRMA and D-TDMA.

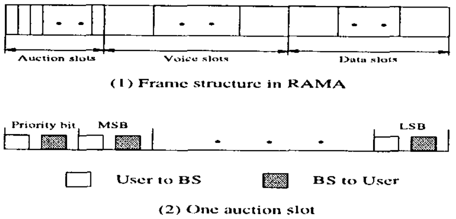
## 4 Related Work

In this section, we summarize, and compare our proposed MAC protocol against, existing MAC protocols for wireless local area networks.

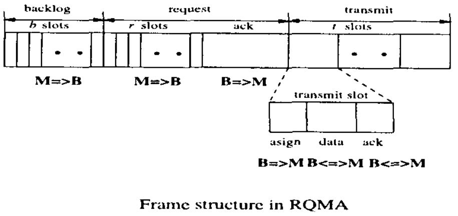
**PRMA:** The first MAC protocol proposed to support wireless communication is the PRMA protocol [3]. As shown in Fig. 5 (1), in PRMA, time is divided into basic transmission units, called *slots* and several slots form a scheduling unit, called a *frame*. Users are classified into two types: voice users and data users. PRMA does not have dedicated reservation bandwidth. Each user with voice/data to transmit simply randomly chooses a slot available inside a frame for packet transmission. Whether or not contention occurs in a slot will be known to all the senders by the end of the slot. If a voice user successfully transmits its voice packet in a slot, the slots will be labeled as reserved in subsequent frames until released by the voice user upon completion of its transmission. This rule, however, does not apply to data users, who are required to contend for every slot it would like to use for data transmission. Due to its CSMA nature, PRMA suffers from low utilization in medium to heavy traffic loads.

**D-TDMA:** As shown in Fig. 5 (2), in D-TDMA, time is divided into frames, and each frame is composed of reservation slots, voice slots and data slots. A slotted ALOHA approach is used in reservation slots for reservation requests. That is, to reserve an information (voice/data) slot, a user sends in a randomly chosen reservation slot a reservation packet. The reservation packet contains information needed to establish a connection, e.g., the source/destination addresses. At the end of a reservation period, successful reservation will be identified and the final slot schedule will be broadcast to all the users by the base station. Once allocated a voice slot, a user can use the same slots in subsequent frames until it completes its transmission, while a data user is granted one data slot (in the same frame as it makes the reservation) at a time. Unsuccessful users will retry in the next frame according to a *reservation retransmission* probability.

**RAMA:** RAMA is very similar to D-TDMA except that it uses a different reservation approach. As shown in Fig. 6 (a), reservation slots in a frame are replaced by auction slots in RAMA. In each auction slot, the available resources (i.e., information slots) will be auctioned to requesting users and will be assigned to the winner of the auction. The auction procedure works as follows (Fig. 6 (a)): each requesting user is assigned a user ID which is randomly generated when the user decides to attend the auction. The number of digits used in the random number depends on the number of users currently in the network. Requesting users start to transmit their IDs in the auction slots, one at a time, from the most significant bit to the least significant bit. After each bit transmission, the base station broadcasts the largest bit value it receives just now, and those contending mobile hosts with unmatched bit value will drop off. There will be a final winner by the end of each auction slot. This winner will not attend any future auction in the same frame. The users dropping off in the



(a) RAMA



(b) RQMA

Figure 6: Frame structures of RAMA and RQMA.

current auction slot can select another random number and reenter the auction in the next slot. One attractive property of RAMA is that it is guaranteed that one mobile host will finally win out in each auction and be successful in sending its reservation request to the base station.

**DRMA and FAMA:** DRMA is a variation of the above protocols, and differs in the degree of design complexity and the level of bandwidth efficiency thus achieved. DRMA eliminates the reservation/auction slots in D-TDMA/RAMA, and uses (if necessary) an available slot as a set of reservation slots. Efficiency is achieved by dynamically assigning reservation slots, rather than using fixed reservation slots. FAMA, on the other hand, basically applies the carrier sense multiple access with collision detection mechanism to the control and jamming packets sent from mobile hosts to the base station, and can be regarded as a CSMA/CD scheme in a wireless LAN.

All the above wireless MAC protocols are tailored to meet the specific requirement of supporting only voice and data users, and do not address the need for supporting other aspects of QoS. In particular, they provide no temporal QoS required by bus location tracking applications. Recently, several other MAC protocols have been proposed that address the QoS issues in wireless LANs, which we summarize below.

**RQMA:** RQMA supports three types of traffic: constant bit rate (CBR), real-time, and best-effort. A frame in RQMA is divided into three fields: $b$ backlog slots, $r$ request slots (and their corresponding ack subfields), and $t$ transmission slots (Fig. 6 (b)). A mobile host sends a request in a request slot (in a slotted ALOHA fashion) to the base station to either establish a RT/CBR session or send best-effort packets. If the base station successfully receives a request in a request slot, it sends an ack in the corresponding ack subfield. In case that a real-time session is established, a mobile host then uses one backlog slot (assigned by the base station) to inform the base station of any newly-arrived packets of the real-time session and their deadlines. The base station then determines when a mobile host can send/receive data packets of a session, by specifying in the *assign* subfield of each transmit slot the mobile host id and the session id. The most desirable feature of RQMA [8] is that it takes into consideration of the error characteristic of the wireless channel, and establishes *a priori* a *real-time retransmission* session to retransmit time-critical data packets upon error detection in the normal transmission phase.

**MCNS MAC:** It has also come to our attention that the Cable Modem standard – Data Over Cable Service Interface Specification (DOCSIS) has recently been developed by the industrial association Multimedia Cable Network System (MCNS) Partners to specify the internal and external network interfaces for a system that allows bi-directional transfer of Internet Protocol traffic, between the cable system and the customer ends, over a cable television system. In particular, the Radio Frequency (RF) Interface Specification of DOCSIS [9] specifies the MCNS MAC protocol to be used in the Cable Modem system.

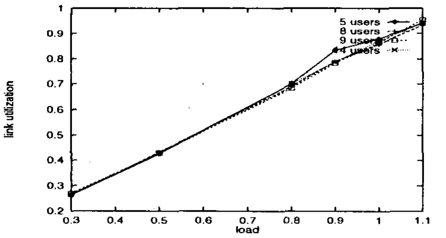
There are many features in common between the MCNS MAC protocol and the proposed MAC protocol. For example, as we use user ID to identify mobile subscribers in a cell, MCNS uses the Service ID to provide both device identification and class-of-service management to the cable modems (which are equivalent to mobile subscribers in wireless networks). Similar to our proposed MAC protocol, cable modems in MCNS request bandwidth for data transmission and the cable modem termination system (CMTS) broadcast to every cable modem the slot allocation schedule.

In spite of all the above research efforts, we believe our proposed MAC protocol is the first that (i) takes into account of the physical layer characteristics (e.g., Reed Solomon code encoding for error control, preambles/postambles for synchronization, dynamic slot adjustment to alleviate asymmetry on the forward/reverse channels) and constraints (e.g., the two-control-field structure to deal with the half-duplex transmission constraint), (ii) provides a mechanism for deterministically ensuring temporal QoS (e.g., the 4-second access delay requirement for up to 8 GPS mobile subscribers), and (iii) is being implemented on a narrow-band wireless modem testbed.

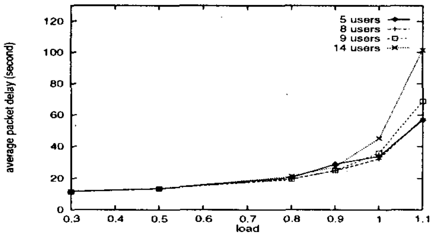
## 5 Performance Evaluation

We have implemented OSU-MAC in a Java-based simulation environment, called *JavaSim* [10], and conducted a simulation study to validate the proposed design. We do not include a simulation comparison to the other existing protocols (summarized in Section 4) because all the protocols have been designed with different objectives under different environments: OSU-MAC has been designed for both bus tracking applications and regular data applications on a specific testbed, while other protocols were designed either for voice/data users (PRMA, D-TDMA, RAMA, DRMA, FAMA), for real-time/best-effort traffic on an abstract model (RQMA), or for cable modem users (MCNS). A comparison among them would not be fair.

The simulation scenario is as follows: there are up to 8 buses within the cell covered by a base station. Each bus carries a GPS unit that transmits GPS packets periodically to report its location. Also, mobile subscribers in the cell may send/receive short e-mails on the reverse/forward channel. For the purpose of evaluating the MAC performance, we assume the e-mail messages are generated at a mobile subscriber according to a Poisson process with mean interarrival time $T$. Two types of packets are used in the simulation: packets of fixed length $L = 120$ bytes and variable-length packets whose length is drawn from a uniform distribution between 40 and 500 bytes. When the number of GPS users is less than or equal to (greater than) 3, each



(a) Link utilization



(b) Packet delay

Figure 7: Performance evaluation with respect to link utilization and packet delay.

notification cycle on the reverse channel has $d = 9$ ($d = 8$) data slots, among which the first is a contention slot for registration and reservation.

Given that there are $m$ mobile data subscribers in the cell, the *load* index $\ell$ of the reverse channel is defined as

$$\ell = \frac{\frac{m \times 3.9844}{T} \times L}{40 \times d},$$

where $\frac{m \times 3.9844}{T}$ is the average total number of messages generated in a notification cycle, $\frac{m \times 3.9844}{T} \times L$ is the total number of bytes generated, and $40 \times d$ is the total number of data bytes that can be transported in the $d$ data slots on the reverse channel.

The simulation runs are designed to evaluate the system performance under light, medium and heavy loads, with the value of $\ell$ varying from 0.3, 0.5, 0.8, 0.9, 1.0 to 1.1, the number of GPS users varying from 1 to 8, and the number of data users varying from 5 to 14. Given different combinations of traffic conditions, the interarrival time $T$ is calculated as

$$T = \frac{m \times L \times 3.9844}{40 \times d \times l}.$$

In spite of several system parameters involved, the results are found to be quite robust in the sense that the conclusion drawn from the performance curves (reported below for the variable-length packet case) is valid over a wide range of parameter values.

**Utilization on the reverse channel:** The link utilization, defined as the percentage of the available bandwidth used to carry data on the reverse channel, versus the load index $\ell$ is shown in Fig. 7 (a). When the load index $\ell \leq 0.8$, most packets get through, and the link utilization is close to the traffic load. When the load index is close to 1, some packets are dropped because of buffer overflow, and the link utilization is smaller than the traffic load.

**Packet delay:** Packet delay versus the load index is depicted in Fig. 7 (b). When $\ell \leq 0.5$, packets can be delivered in three to five cycles, even under the case of variable-length packets (with an average packet size of 280 bytes). This, coupled with the fact that the bandwidth available on the reverse channel is limited due to the physical layer characteristics, demonstrates the ability of OSU-MAC to accommodate a large number of mobile subscribers, while maintaining high utilization and small packet delay under small to medium loads. When the load increases beyond 0.9, the packet delay increases dramatically, due to the fact that the traffic load grows beyond the system capacity and packets start to queue up.

**Control overhead:** We use the ratio of the number of reservation packets (transmitted in contention slots) to the total number of data packets (transmitted in data slots) as an index of control overhead. As depicted in Fig. 9, counter-intuitively the control overhead decreases as the load increases. This is because as the load increases, reservation requests are usually piggybacked in the reservation bit of the packets sent uplink, leading to the less number of reservation packets. Due to the same reason, as the load increases, the probability that collision occurs in contention slots decreases (Fig. 8 (a)), and the average reservation latency also decreases (Fig. 8 (b)).

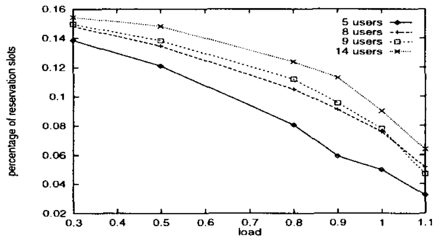
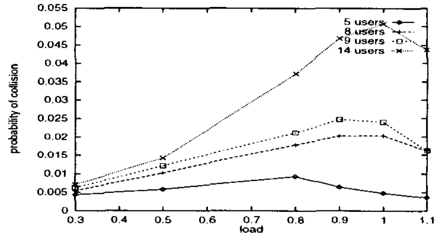
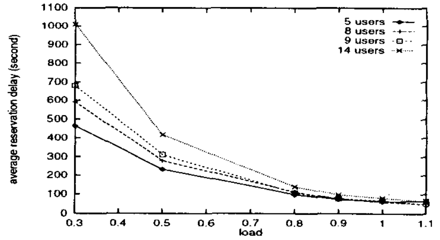


Figure 8: Control overhead as a function of load.

**Fairness:** As described in Section 3.5, we use the round robin algorithm to assign data slots on the reverse channel to mobile subscribers with data packets. As shown in Fig. 10, OSU-MAC ensures



(a) The probability of collision



(b) Reservation latency

Figure 9: Performance evaluation with respect to the probability of collision in contention slots and the reservation latency.

fairness among mobile subscribers (i.e., the fairness index under all traffic loads are over 0.99), where the fairness index is defined as [11] $\frac{n \cdot \sum_{i=1}^{n} u_i^2}{(\sum_{i=1}^{n} u_i)^2}$ and $u_i$ is the bandwidth the $i$ mobile subscriber acquires.

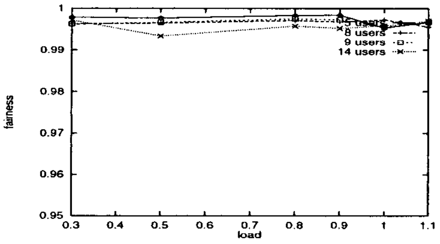
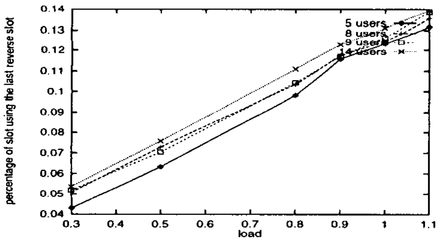


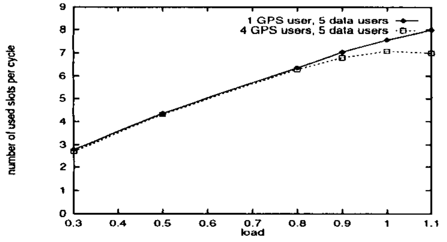
Figure 10: Fairness

**Performance improvement due to the two-control-fields design and dynamic slot adjustment:** Fig. 11 (a) gives the percentage of bandwidth gain by using the second set of control fields. This is obtained by calculating the ratio of the number of data packets sent in the last data slot on the reverse channel to the total number of data packets sent (as the last data slot on the reverse channel overlaps with the first set of control fields). As little as 5% and as much as 14% of the bandwidth is saved by use of the second set of control fields.

Fig. 11 (b) depicts the average number of data slots that have been used in the case that the number of GPS users is 1 and 4, respectively. Recall that when there are 3 or less GPS users, 5 GPS slots will be converted to an additional data slot. Hence, Fig. 11 (b) shows how effective the dynamic slot re-adjustment approach helps in utilizing bandwidth originally allocated to unused GPS slots. The effect of dynamic slot re-adjustment is not significant when the load is light, but as much as 15% more bandwidth can be utilized with slot re-adjustment.



(a) Improvement using two control fields.



(b) Improvement using dynamic slot adjustment.

Figure 11: Performance improvement resulted from using two control fields and dynamic slot adjustment.

## 6 Conclusion

In this paper, we have designed and implemented a MAC protocol, *OSU-MAC*, that coordinates the transmission activities on the forward and reverse channels so as to support both the real-time bus location tracking application and the regular data applications, on an OSU narrow-band wireless modem testbed. There are several unique features of OSU-MAC: first, we delegate to the base station the full responsibility of resource arbitration, channel access, and registration in each cell. This base- station-based scheduling approach enables provisioning of deterministic QoS for real-time applications, effective supports of slot reservation and mobile registration, while maximizing the system utilization. Second, we take into account of the error characteristics of the physical layer. In particular, we consider the half-duplex transmission constraint and propose a two-control-field structure to fully utilize the limited bandwidth available on the reverse channel. Third, to make use of the unused bandwidth originally reserved for real-time traffic, we also propose a dynamic slot adjustment scheme. Finally, as a real MAC protocol currently being implemented on the OSU narrow- band wireless modem testbed, we insert, wherever needed as advised by the wireless modem researcher, preambles, postambles, and guard times between packet slots or notification cycles for synchronization. We are currently implementing OSU-MAC on the MS-Windows operating system and will develop a real-time bus tracking application and an email delivery system to demonstrate the use of *OSU-MAC* for distributed applications.

## References

[1] R. E. Blahut, *Theory and practice of error control codes*. Addison-Wesley, Reading, MA, 1983.

[2] A. J. McAuley, "Reliable broadband communication using a burst erasure correcting code," in *Proc. of ACM SIGCOMM*, pp. 297–306, September 1990.

[3] S. Nanda, D. Goodman, and U. Timor, "Performance of PRMA: A Packet Voice Protocol for Cellular Systems," *IEEE Trans. Veh. Techn.*, vol. 40, pp. 584–598, 1991.

[4] N. Wilson, R. Ganesh, K. Joseph, and D. Raychaudhuri, "Packet CDMA Versus Dynamic TDMA for Multiple Access in An Integrated Voice/Data PCN," *IEEE JSAC*, vol. 11, pp. 870–884, 1993.

[5] X. Qiu and V. O. Li, "Dynamic Reservation Multiple Access (DRMA): A New Multiple Access Scheme for Personal Communication System (PCS)," *Wireless Networks*, vol. 2, pp. 117–128, 1996.

[6] N. Amitay, "Distributed Switching and Control with Fast Resource Assignment/Handoff for Personal Communications Systems," *IEEE JSAC*, vol. 11, pp. 842–849, 1993.

[7] C.L.Fullmer and J.Garcia-Luna-Aceves, "Floor Acquisition Multiple Access (FAMA) for Packet-Radio Networks," in *Proceedings of ACM SIGCOMM '95*, 1995.

[8] N. R. Figueira and J. Pasquale, "Remote-Queueing Multiple Access (RQMA): Providing Quality of Service for Wireless Communications," in *Proc. IEEE INFOCOM'98*, IEEE Computer Society, April 1998.

[9] CableLabs, "Data Over Cable Service Interface Specification." http://www.cablemodem.com/public/pubtech.html, April 1998.

[10] H.-Y. Tyan, B. Wang, Y. Ye, L. Su, W. Lin, and C.-J. Hou, "NetSim$^Q$: a java-integrated network simulation tool for qos control in high speed networks." http://eewww.eng.ohio-state.edu/drcl/grants/middleware97/netsimQ.html., 1999.

[11] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling,*. Wiley-Interscience, New York, NY, May 1991. Winner of *1991 Best Advanced How-To Book, Systems award* from the Computer Press Association.