A Probing Process for Dynamic Resource Allocation in Fixed Broadband Wireless Access Networks

Qiang Wang and Anjali Agarwal, *Member, IEEE*

***Abstract*—This paper deals with the support of both real-time and non-real-time communication services in a broadband fixed wireless access network. It investigates the feasibility of dynamically allocating the bandwidth not utilized by other sectors in the staggered resource allocation method. A new medium-access control (MAC) protocol based on the proposed probing process that does not need information exchange and coordination among base stations is presented to improve data packet transmission. The probing process detects available slots unused by other sectors to provide a higher capacity in sectorized cells. A simplified C-PRMA, referred to as prioritized access with centralized polling command, is adopted to effectively implement the probing process. Simulation results show that the proposed MAC protocol with probing process provides an improvement over C-PRMA. In terms of data traffic, the proposed protocol increases the link capacity to near 100% as opposed to 78% for C-PRMA when the workload of other sectors is 69% of the link capacity. In terms of voice traffic, the probing process provides 23% more user capacity than C-PRMA when there are 20 voice users in other sectors.**

***Index Terms*—C-PRMA, frequency-division duplex (FDD), probing process, staggered resource allocation (SRA).**

I. INTRODUCTION

As THE Internet grows, broadband access has received much attention. Much of this attention has been focused on providing wired broadband access using existing copper line or coaxial cable. Wireless broadband access is emerging as an attractive alternative that can provide value-added nomadic portability and anytime, anywhere access. Customers are expecting high quality, reliability, and easy access to high-speed communications from homes and small businesses. High-speed services are needed in the very near future for: 1) accessing the World Wide Web for information and entertainment; 2) providing data rates comparable to local-area networks for telecommuters to access their computer equipment and data at the office; and 3) supporting various traffic with quality-of-service (QoS) guarantees as the need for broadband multimedia communications involving digital audio and video grows. In order to provide bandwidth on demand with QoS guarantees using scarce radio spectrum in a broadband fixed wireless network, the medium-access control (MAC) must 1) efficiently reuse a limited spectrum with interference avoidance and 2) handle dynamic and diverse traffic with high throughput.

Manuscript received March 14, 2002; revised June 18, 2002, January 23, 2003, and March 3, 2003. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under its Research Grants Program and in part by the Faculty Research Development Program of Concordia University.

The authors are with the Electrical and Computer Engineering Department, Concordia University, Montreal, PQ H3G 1MB, Canada.

Digital Object Identifier 10.1109/TVT.2003.814224

To support a user data rate of 10 Mb/s in an interference-limited wireless environment, a bandwidth of several megahertz is needed for time-division multiple access (TDMA) [1]. In wireless networks, the microwave spectrum is expensive, so efficient strategies for reusing frequencies and managing cochannel interference are critically important. The need for reuse of a common radio spectrum in all cells has also been noted by [2] and [3] for mobile broadband wireless networks. In fixed wireless networks, cell sectorization and directional antennas at fixed terminal locations are key components in reducing interference from neighboring sectors and cells. By exploiting this advantage, Fong *et al.* [4] have proposed the staggered resource allocation (SRA) method as a distributed dynamic resource allocation (DRA) algorithm for the network where the same radio spectrum is shared by each sector in every cell on a dynamic time-division basis. This new technique has been proposed for time-sharing between sectors using time reuse to avoid sources of major interference. An enhanced version has been proposed in [5] by providing different degrees of concurrent transmission in different time slots. SRA strategies permit reuse of the same bandwidth in every cell, resulting in a high degree of spectral efficiency [4] without the use of a central controller, as done in other DRA approaches [6]. For a review of the DRA mechanisms, refer to [7], which provides a survey of fixed, dynamic, and hybrid channel assignment schemes in general. It is well known that distributed dynamic channel allocation (DCA) mechanisms are simple and practical to implement because they are based on locally obtained information, as opposed to the centralized DCA schemes that assume global knowledge of information for channel assignment; this global knowledge is of course at the expense of high centralization overhead. The proposed SRA-based method belongs to the class of distributed dynamic channel allocation mechanisms.

The SRA method uses a distributed scheduling algorithm to avoid major sources of interference while allowing concurrent packet transmission. The combination of directional antennas with the SRA method is shown to be highly effective in controlling cochannel interference [4]. In the SRA method, as a conservative way, the traffic load of a sector has to be limited to less than one-third of total channel capacity to avoid interference from major sources in the neighboring cells. In practice, not all sectors have the same traffic load; specifically, some sectors own more users while others own fewer users. The main drawback in [4] and [5] is the static nature of the time allocation technique, which can cause resource wasting, particularly with bursty traffic sources of multimedia services. In this paper, the slots that were originally assigned to the sectors with lower

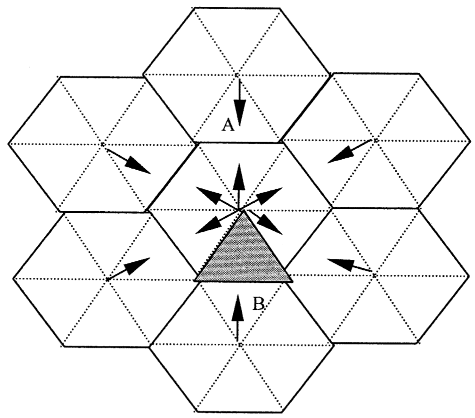


Fig. 1. Major sources of interference for downlink.

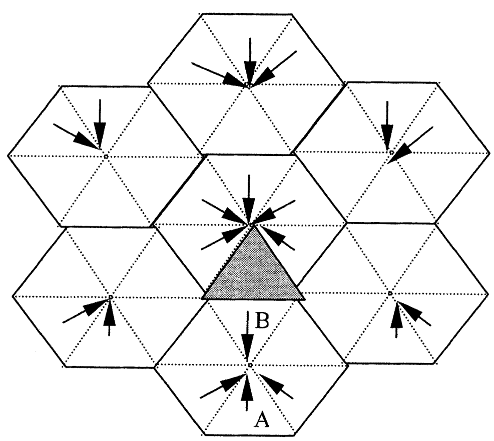


Fig. 2. Major sources of interference for uplink.

traffic load but not utilized under the condition of interference avoidance are allocated to the sectors that own more users. A probing process is proposed without the need of information exchange and coordination among base stations to solve the problem and overcome the limitation of maximum channel capacity utilization in the SRA method. The probing process is implemented in a frequency-division duplex (FDD) MAC protocol and supports both real-time and non-real-time traffic.

In Section II, the system model is presented with the interference analysis in a cellular wireless network, and the SRA method used with the proposed MAC protocol is explained. Section III presents the MAC protocol including the proposed probing process. In Section IV, the simulation results are shown and discussed in detail. Conclusive remarks are enlisted in Section V.

II. System Model

In this paper, the study focus is on a fixed broadband packet-switched network using TDMA technique with user data rates of 10 Mb/s, link lengths typically less than 10 km, and operating frequency in the range of 1–5 GHz. The whole service area in the network is divided into cells. Each cell is further divided into multiple sectors, each of which is covered by a sector antenna colocated with a base station (BS) at the center of the cell. Because of the colocation, sector antennas are also called BS antennas. Terminals (users) use directional

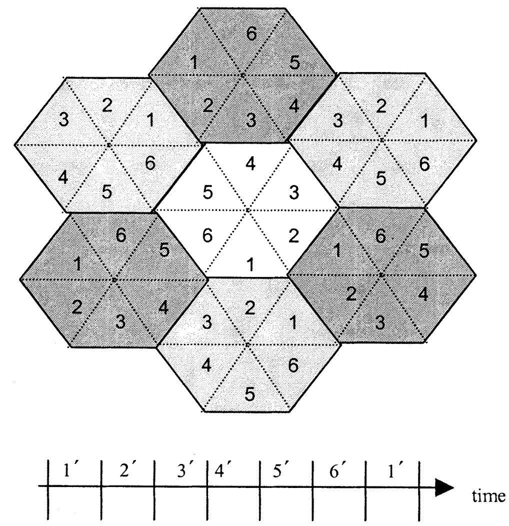


Fig. 3. Cell layout and frame structure.

Sector	1′	2′	3′	4′	5′	6′
1	a	e	d	b	c	f
2	f	a	e	d	b	c
3	c	f	a	e	d	b
4	b	c	f	a	e	d
5	d	b	c	f	a	e
6	e	d	b	c	f	a

Fig. 4. Slot assignment order for the SRA method.

antennas mounted on the rooftop and pointed to their respective BS antennas. The 60° beam width of each BS antenna should be just wide enough to serve the whole sector, while a terminal antenna can have a smaller beam width of 30° to suppress interference. The ratios of front-to-back lobe gain (FTB) for BS and terminal antennas are assumed to be finite and high such that interference between users is mitigated. Cell sectorization and directional antennas at fixed locations are used in reducing the amount of interference from neighboring sectors and cells [4]. Time is slotted such that a packet can be transmitted in each slot. In addition, the downlink and uplink between terminals and BS can be provided by FDD.

A. Interference Analysis

A hexagonal cell layout is considered in this paper, where each cell is divided into six sectors. Figs. 1 and 2 show the interference sources for both downlink and uplink for a tagged sector under consideration (shaded in the figures). Using a simple path-loss model [8], it was observed that the major interference for the downlink in the tagged sector comes from other intracell sectors, sector A and the opposite sector B. A similar observation can be made in Fig. 2 for the uplink.

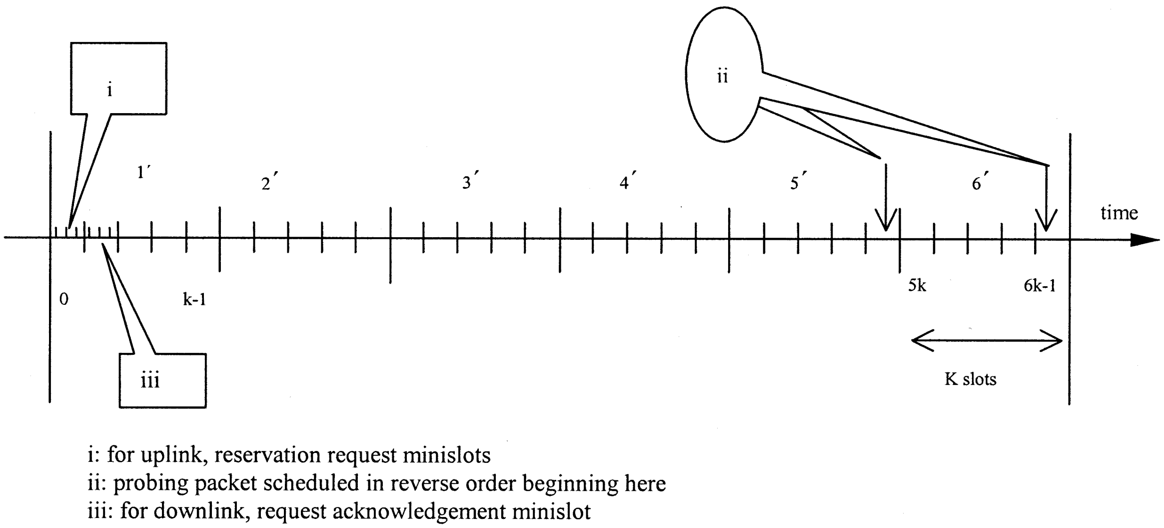


Fig. 5. Frame division and probing process for sector 1.

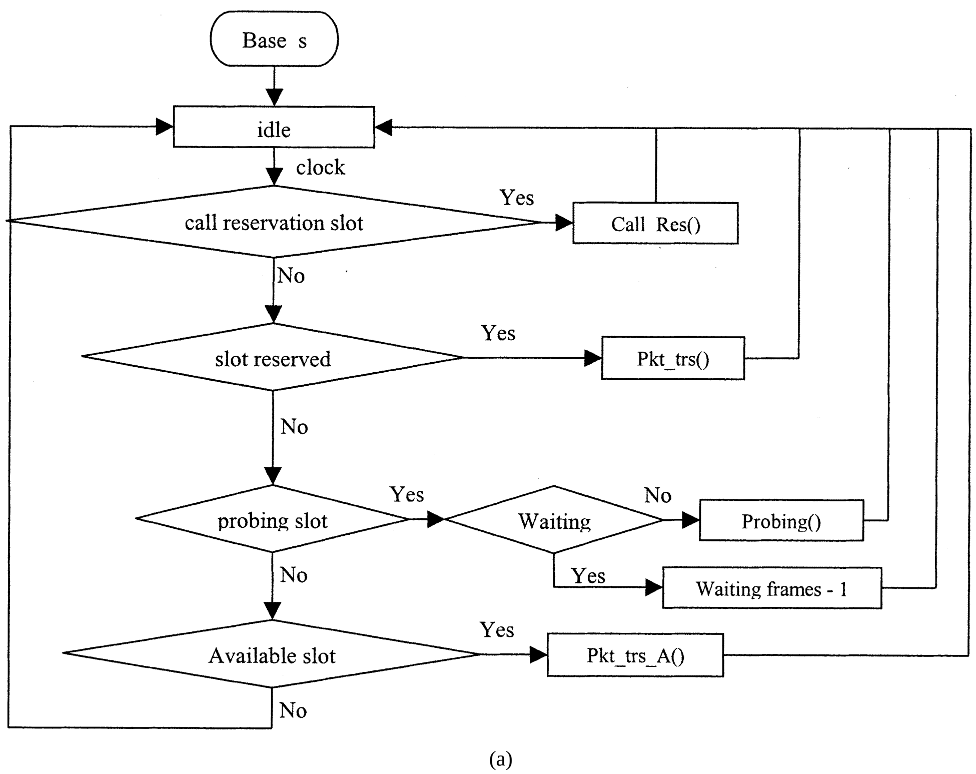


Fig. 6. (a) Main procedure for base station of sector 1.

However, interference received from other neighboring cells can be significantly attenuated because of the high FTB ratio of directional antennas and distance.

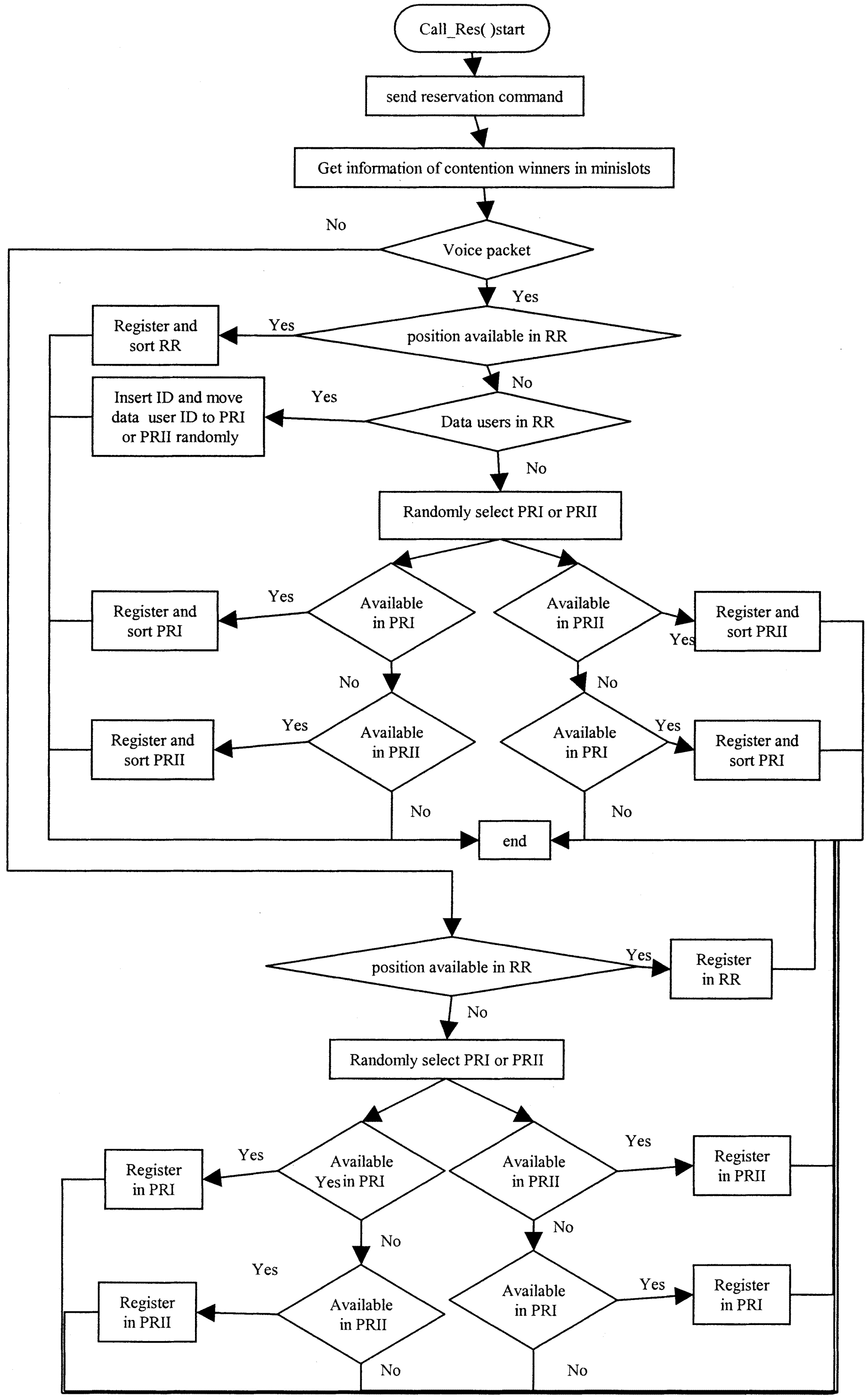
B. SRA Method

In packet-switched networks, time slots are the bandwidth resources. The time slots are allocated dynamically to various transmitters to send data packets such that a given signal-to-interference ratio can be guaranteed at the intended receiver for successful reception. This results in the concept of DRA.

Fong *et al.* [4] have proposed the SRA method as a distributed DRA algorithm for the network where the same radio spectrum is shared by each sector in every cell on a dynamic time-division basis. With the use of directional antennas to suppress interference, the SRA method is particularly effective in avoiding both intercell and intracell interference.

In the SRA method, time slots are grouped into six subframes ($1'$–$6'$) and sectors labeled by 1 to 6 anticlockwise, as shown in Fig. 3. The sector labeling patterns for adjacent cells differ by a 120° rotation, thus creating a cluster of three cells whose patterns can be repeated across the entire system. Note that the time frame shown in the figure is applicable to both downlink and uplink, which are provided by the TDD or FDD technique.

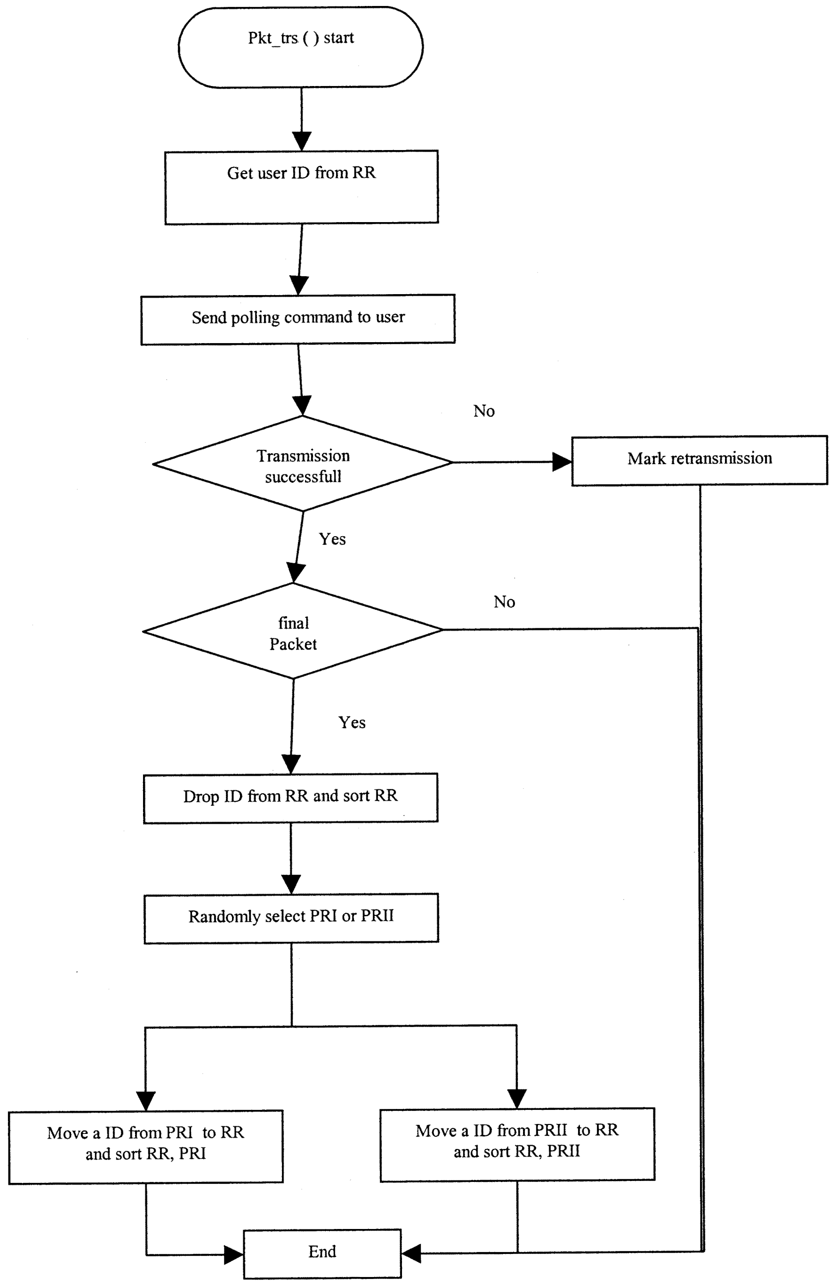
Each sector assigns time slots for transmitting packets to or from its terminals according to a special order shown in Fig. 4



(b)

Fig. 6. *(Continued.)* (b) Call reservation procedure for base station of sector 1.

(in case a terminal needs to send packets to its BS, it is assumed that the BS is made aware of the need, perhaps via a separate dedicated channel or a contention channel). For example, sector 1 first schedules packets for transmission in time slots of subframe $1'$ in the figure. If it has more traffic to send, it then uses subframe $4'$, subframe $5'$, subframe $3'$, subframe $2'$, and finally subframe $6'$. This order of scheduling is shown in Fig. 4 as a, b, c, d, e, and f. The reason for such an order is that if interference due to concurrent packet transmission in the same cell can be tolerated, then after using all slots in the first subframe, a sector should use the first subframe of the opposite sector in the same cell, in order to make the best use of BS directional antennas. Following that, time slots in the first subframes for the sectors next to the opposite sector are used. To avoid interference due to



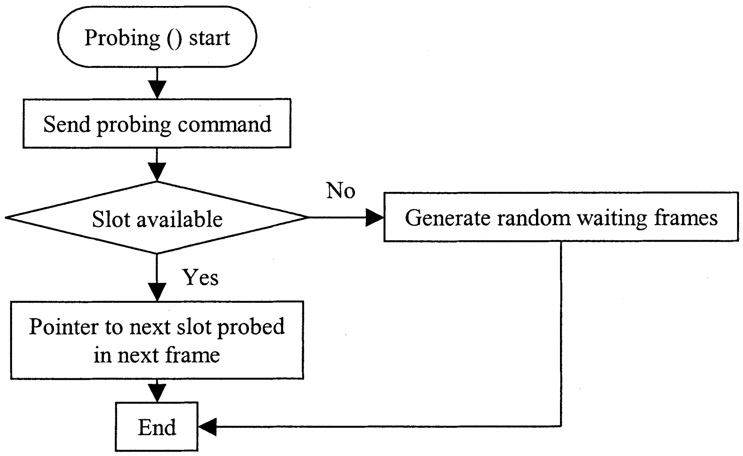
(c)

Fig. 6. *(Continued.)* (c) Packet transmission procedure for sector 1.

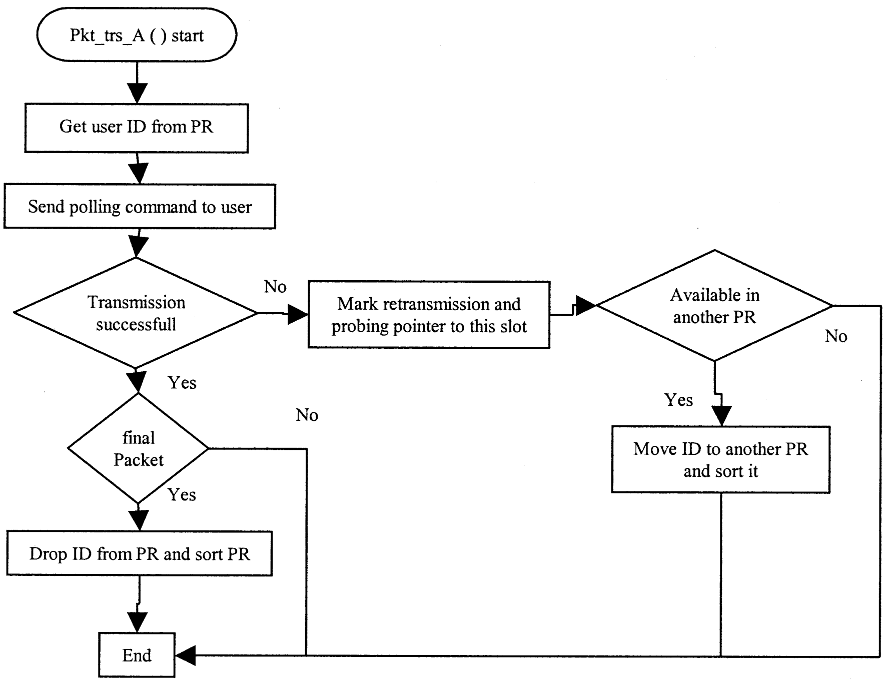
overlapping antenna patterns of neighboring sectors, their first subframes are used as a last resort. For simplicity (while causing very minor throughput degradation), the assignment from the left- to right-hand side of the subframes is only considered. As depicted in Fig. 4, the assignment order for the next sector is "staggered" by a right rotation by one subframe based on the order of the previous sector. The assignment order, regardless of the associated sector, is generally referred to as the staggered order in the following.

It is easy to see from Fig. 4 that if all sectors have traffic loads of less than one-sixth of total channel capacity, all packets are transmitted in different time subframes, thus causing no interference within the same cell. Of course, as the traffic load increases, packets are transmitted simultaneously, thus increasing the level of interference. Nevertheless, the staggered order exploits the characteristics of directional antennas to allow multiple concurrent packet transmissions while reducing the intracell interference.

Besides managing intracell interference, the SRA method also avoids interference from major sources in the neighboring cells. This is particularly so when the traffic load is low to moderate. For example, consider the downlink for sector 1 in the middle cell of Fig. 3. Sector 2 in the bottom cell and sector 3 in the upper cell are the major sources of interference. By examining the staggered order for sector 1, 2, and 3, it was observed that they do not transmit simultaneously, so they will not interfere with each other provided that all of them have a traffic load of less than one-third of total channel capacity. The



(d)



(e)

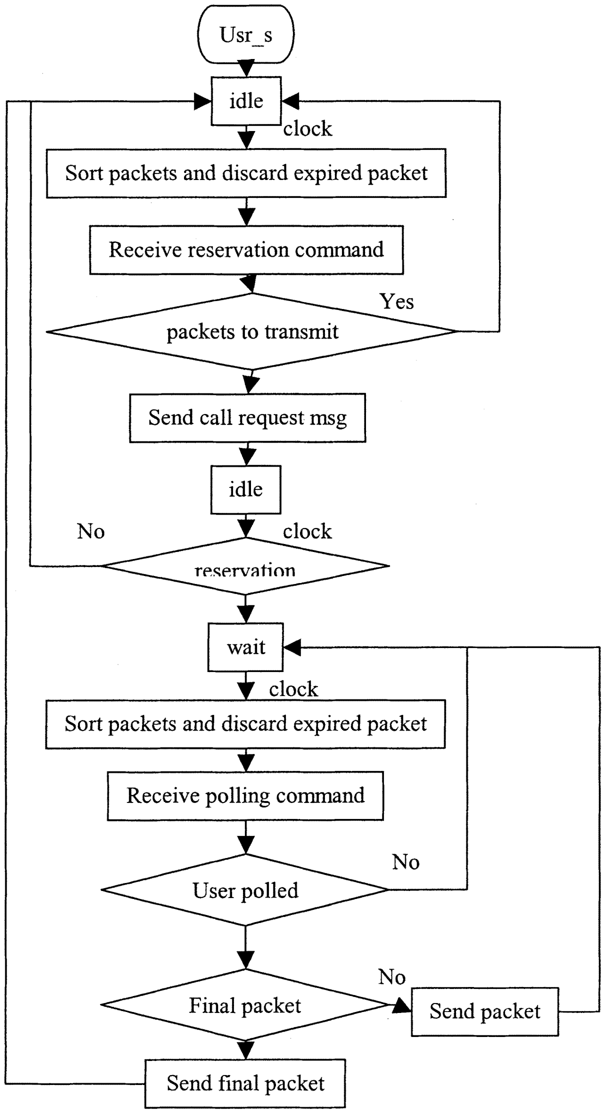
Fig. 6. *(Continued.)* (d) Probing procedure for base station of sector 1. (e) Sector 1 procedure for packet transmission in sector 2 or 3.

same argument applies to the uplink where sectors 2 and 5 of the bottom cell in the figure now become the major sources of interference. The same applies to each sector in every cell due to the symmetry of the staggered order and cell layout. This feature of SRA is the basis of this work to design a new MAC protocol.

III. MAC Protocol With Probing Process

This section describes the proposed MAC protocol with probing process. According to the SRA method described in Section II, a frame is divided into six subframes and is shown in Fig. 5. For the central cell in Fig. 3, subframes $1'$ and $4'$ are reserved for sectors 1 and 4, subframes $2'$, $5'$ are reserved for sectors 2 and 5, and subframes $3'$, $6'$ are reserved for sectors 3 and 6. To effectively implement the probing process, a centralized packet reservation multiple access (C-PRMA) [11] is adopted to enhance the control of BS to users. In this work, only the data and voice traffic are studied. BS schedules the transmission of downlink data packets or the polling of uplink data packet transmission permission with lower priority than the scheduled voice traffic.

In the proposed protocol, the S-ALOHA algorithm is utilized for the reservation of requests with a random retransmission of collided minipackets and as used in C-PRMA [11]. However, a simplified C-PRMA is considered to study the effect of the proposed probing process. The first slot of subframes $1'$, $2'$, $3'$ in the uplink channel is dedicated to the transmission of the reservation requests. The reservation request can be structured as a minipacket, as it carries a limited amount of information. Consequently, the available slots can be split into $m > 1$ minislots, each one of a length equal to the minipacket transmission time. When the BS issues a transmission-request command, each user with pending requests decides whether or not to make a request



(f)

Fig. 6. *(Continued.)* (f) Procedure for user station of sector 1.

with probability p_r (i.e., in a pr-persistent manner). If it has decided to send the request, it does so in one of the request minislots. After each first slot of subframes $1'$, $2'$, $3'$ in the uplink channel, a slot in the downlink channel is dedicated to acknowledge the reservation requests. In the acknowledgment minislots, BS transmits the identifier of all the new users that have successfully reserved.

The request includes the following service parameters [11]:

- L_i maximum tolerable packet delay, in slots;
- T_i packet interarrival time when the source is transmitting, in slots;
- P_i voice or data packet;
- W_i waiting time of the first packet of the burst in the transmission buffer before its successful reservation;
- C_i lifetime, in slots, of the head-of-the-line (HOL) packet in the transmission buffer of user i.

The parameter C_i is equal to $L_i - W_i$ when a new reservation is accepted, and it is decreased after each packet transmission at each time slot.

In the scheduling process depicted in Fig. 5, it is assumed that subframes $1'$ and $4'$ are reserved for sector 1 of the central cell in Fig. 1. Sector 1 schedules its packet transmission in time slots of subframes $1'$ and $4'$ beginning from slots S_0 to S_{k-1} and slots S_{3k} to S_{4k-1}, where slot S_0 is used as reservation request minislots and slot S_1 is used as request acknowledgment minislots. Sector 1 can also schedule its packet transmission in time slots of subframes $5'$ and $2'$, or in time slots of subframes $6'$ and $3'$. Scheduling is performed in the reverse order starting from S_{6k-1} to S_{5k} in the case of subframe $6'$ or from S_{5k-1} to S_{4k} in the case of subframe $5'$. A similar order is followed for subframe $2'$ or $3'$. Sector 1 should first execute the probing process to detect available slots before it schedules its packet transmission in subframes $5'$ and $2'$ or $6'$ and $3'$.

Fig. 6 shows the flow charts for the procedures used to evaluate the performance of the proposed MAC protocol. The flow charts for the BS of sector 1 are presented in Fig.6a–e. The BS after receiving the clock signal judges the kind of present slot and calls the respective procedure. Fig. 6(f) shows the procedure for the user stations. If a user receives the reservation command and there are packets to transmit, it will send a call request to the BS. If the reservation is successful, it will transmit packets in certain slots according to the command from the BS. If not, it will try to send a request again in the next frame.

In the following, the main procedures defined in the flow charts are given in detail with their C-like code.

A. Probing Procedure

The proposed probing procedure as described in Fig. 7 begins by the sector polling the user nodes to determine unassigned or available slots. NumProbingI and NumProbingII define the total number of available slots in subframes $6'$ and $3'$, and subframes $5'$ and $2'$, respectively, for sector 1. The polling is done in the reverse order starting from slot $6k-1$ and slot $5k-1$ for subframes $6'$ and $5'$, respectively. If the user node responds with a probing packet, the sector assumes that the slot can be used and increases the corresponding number of slots by one. The process continues to determine more available slots. If a slot is not available, the sector gives up the slot and tries again after random frames.

The probing process is forbidden to proceed in Slot k, Slot k+1, Slot 2k, or Slot 2k+1 because Slot k and Slot 2k are divided into the reservation request minislots for other sectors in uplink, and Slot k + 1 and Slot 2k+1 are used for the reservation acknowledgment minislots by other sectors in downlink. Therefore, the maximum of NumProbingI and NumProbingII is $2k-2$.

An essential parameter to be confirmed in the proposed simulation is the waiting time, in frames, of probing command retransmission after an unsuccessful probing procedure. This parameter reflects how often the probing process starts. Too intensive probing packet transmissions with lower waiting times cause serious effect to the communications in sector 2 and sector 3, as shown in the simulation results in Section IV. To diminish the interference to the communications of sector 2 and sector 3 as much as possible, it is important to find the proper value for the waiting frames of probing command retransmission.

```
void Probing(){

  /* initializing */
  int NumProbingI=0;
  int PntProbingI=5k-1;        /* initially point to the probing slot in subframe 5′*/
  int NumProbingII=0;
  int PntProbingII=6k-1;       /* initially point to the probing slot in subframe 6′*/

  /* Generate the command to ask for the probing packet response from a certain user in subframes 2′ and 5′*/
  Probing_I:
  if ( k+1<PntProbingI<2k || 4k-1<PntProbingI<5k ){    /* Execute probing process in subframes 2′ and 5′ */
      ProbingCommand(i);                               /* send the probing command to User i */
      if (slot available){
        NumProbingI++;
        PntProbingI--;                                 /* probing slot for the next frame */
        if (PntProbingI == 4k-1) PntProbingI = 2k-1;   /* point to the probing slot in subframe 2′ */
        goto Probing_I;                                /* execute probing process again in the next frame */
      }
      else {
        j=rand();
        Wait(j);                                       /* wait for random frames j if slot not available*/
      }
  }

  /* Generate the command to ask for the probing packet response from a certain user in subframes 3′ and 6′*/
  Probing_II:
  if ( 2k+1<PntProbingII<3k || 5k-1<PntProbingII<6k ){/* Execute probing process in subframes 3′ and 6′ */
      ProbingCommand(i);                               /* send the probing command to User i */
      if (slot available){
        NumProbingII++;
        PntProbingII--;                                /* probing slot for the next frame */
        if (PntProbingII == 5k-1) PntProbingII = 3k-1; /* point to the probing slot in subframe 2′ */
        goto Probing_II;                               /* execute probing process again in the next frame */
      }
      else {
        j=rand();
        Wait(j);                                       /* wait for random frames j if slot not available*/
      }
  }
}
```

Fig. 7. C-like code of probing procedure.

B. Call Request Reservation Procedure

Fig. 8 describes the procedure for call request reservation. Three registers named Reserved Register (RR) of size $2k-1$, Probing Register I (PRI) of size $2k-2$, and Probing Register II (PRII) of size $2k-2$ are defined. The positions of RR correspond to the slots $S_1 - S_{k-1}$ and $S_{3k} - S_{4k-1}$ (S_0 reserved for the reservation request minislots). The positions of PRI correspond to slots $S_{k+2} - S_{2k-1}$ and $S_{4k} - S_{5k-1}$ (S_k and S_{k+1} reserved for the reservation request minislots and the reservation acknowledgment minislots by other sectors). The positions of PRII correspond to slots $S_{2k+2} - S_{3k-1}$ and $S_{5k} - S_{6k-1}$ (S_{2k} and S_{2k+1} reserved for the reservation request minislots and the reservation acknowledgment minislots by other sectors). Each position of RR, PRI, and PRII is either empty or filled with User ID. NumRev, NumPI, and NumPII express the user numbers registered in RR, PRI, and PRII.

When the BS receives a real-time reservation request from a voice user, it first checks if there is a position in RR, PRI, and PRII to accommodate the user. If $\mathrm{NumRev} < 2k - 1$, it marks the User ID in the empty position of RR and NumRev is increased by one. If $\mathrm{NumRev} >= 2k - 1$, it checks if there are non-real-time users in RR. If so and ($\mathrm{NumPI} < \mathrm{NumProbingI}$ or $\mathrm{NumPII} < \mathrm{NumProbingII}$), it moves the leftmost (Fig. 5) non-real-time user in RR to PRI or PRII and marks the User ID in the position. If there is no non-real-time user in RR and $\mathrm{NumPI} < \mathrm{NumProbingI}$ or $\mathrm{NumPII} < \mathrm{NumProbingII}$, the sector also tries to mark the User ID in the leftmost position of PRI or PRII and move the non-real-time users to higher positions. Generally, the sector randomly selects PRI or PRII to schedule users. The users may also be scheduled first in PRI and then in PRII. If $\mathrm{NumPI} >= \mathrm{NumProbingI}$ and $\mathrm{NumPII} >= \mathrm{NumProbingII}$, all available slots are full and the request will not be registered and acknowledged.

When the BS receives a reservation request from a non-real-time user and there is an empty position ($\mathrm{NumRev} < 2k - 1$) in RR, it marks the new User ID in the empty position of RR. If RR is full, it randomly selects PRI or PRII to find if there is an empty position in them, in which case it marks the User ID. BS repeats the process until

```
void CallRes() {

  send reservation command to users;
  /* schedule all contention winners in minislots */
  for (j=winners) {          /* for all contention winners */
    c[j]=l[j]-w[j];          /* reset c[j] */

  /* If it is the voice traffic */
    if(Voice_packet&&c[j]>0){
      if(NumRev<2k-1){ /* there is a position in RR */
        rr[NumRev]=j;   /* register the user j in RR */
        NumRev++;
        sort_array(rr); /* sort RR according to c[ ] ascendingly */
      }
      else {             /* no position in RR */
        search_array(rr, data_packet);/* search data packets in RR */
        if (Data_in_array){    /* there is a data user x in RR */
          insert_array(rr,j);/* register j in RR replacing x */
          sort_array(rr);     /* sort RR */
          h=rand_sel_array(pri,prii); /* randomly select PRI or PRII */
          insert_array(h,x);/* insert x in PRI or PRII */
        }
        else {             /* total voice users in RR */
          h=rand()%2;       /* randomly select PRI or PRII */
          if(h==0 and NumPI<NumProbingI){/* select PRI and it's not full */
            pri[NumPI]=j; /* register j*/
            NumPI++;
            sort_array(pri);/* sort PRI[ ] */
          }
          else if(h==0 and NumPII<NumProbingII){   /* if PRI is full */
            prii[NumPII]=j;  /* register j in PRII[ ] */
            NumPII++;
            sort_array(prii);/* sort PRII[ ] */
          }
          if(h==1 and NumPII<NumProbingII){/* select PRII and it's not full */
            prii[NumPII]=j;  /* register j in PRII[ ] */
            NumPII++;
            sort_array(prii);
          }
          else if(h==1 and NumPI<NumProbingI){/* if PRII is full */
            pri[NumPI]=j;
            NumPI++;
            sort_array(pri);
          }
        }
      }
    }

  /* if it is the data traffic */
    if (Data_packet){
      if(NumRev<2k-1){        /* RR[ ] is not full */
        rr[NumRev]=j;          /* register j in RR[ ] */
      }
      else {               /* RR[ ] is full */
        h=rand()%2;          /* randomly select PRI or PRII */
        if(h==0 and NumPI<NumProbingI){/* select PRI and it's not full */
          pri[NumPI]=j;     /* register j in PRI */
          NumPI++;
```

Fig. 8. C-like code of call request reservation procedure.

an empty position is not found, in which case the request is refused. The BS can also first select PRI to schedule packet transmission and then select PRII.

C. Packet Transmission Procedure

Fig. 9(a) and (b) describes the packet transmission procedure in reserved and available slots, respectively. The BS decides the

```
                }
                else if(h==0 and NumPII<NumProbingII){   /* if PRI is full */
                   prii[NumPII]=j;       /* register j in PRII */
                   NumPII++;
                }
                if(h==1 and NumPII<NumProbingII){/* select PRII and it's not full */
                   prii[NumPII]=j;
                   NumPII++;
                }
                else if(h==1 and NumPI<NumProbingI){ /* if PRII is full */
                   pri[NumPI]=j;
                   NumPI++;
                }
            }
        }
    }
}
```

Fig. 8. (*Continued.*) C-like code of call request reservation procedure.

```
void pkt_trs() {

  int CommandPointer=0;              /* initializing */

  /* in subframe 1′ and 4′ */
  i=rr(CommandPointer);          /* get user ID from RR */
  if (i!=0){
     send_polling_command(i);     /* send polling command to user i*/
     CommandPointer++;       /* point to next user */
  }
  if(transmission_successful){
     c[i]=c[i]+t[i];
     schedul(i);          /* schedule next packet for user i */
     if(Final_Packet) {          /* user i reports final packet */
          ShiftArray(rr, i);          /* drop user i by shifting RR from RR[i] */
          if (NumPI!=0 and NumPII!=0) {  /* PRI and PRII is not empty */
            z=rand_select(PRI,PRII);   /* randomly select PRI and PRII */
            rr[2k-2]=z[0];      /* move used ID from z to rr */
            ShiftArray(z,0);       /* shift z */
          }
     }
  }
  else{
        c[i]=c[i]+Long_Frame;       /* re-transmission in next frame */
        MarkRetr[i]=1;          /* mark re-transmission */
  }
}
```

(a)

Fig. 9. (a) C-like code of packet transmission procedure in reserved slots.

scheduling for the slots in the beginning of a frame. In its reserved slots, it sends polling commands according to the RR's User IDs in the order from low position to high position. In the unreserved slots, the BS sends commands according to the User IDs of PRI or PRII in the reverse order from high position to low position. A polling command is not sent by BS if the position of RR, PRI, or PRII is empty.

In the process of message transmission, any interference caused by other sectors that originally reserved subframes $2'$, $3'$, $5'$, and $6'$ will result in sector 1 or its users stop the packet transmission.

When a user finishes the transmission, it will send the final packet with stop information in its header to the BS. After receiving the packet, the BS removes the User ID from RR or PR.

IV. Performance Evaluation

Simulation was done using OPNET to study the sector capacity performance of the protocol for voice and data traffic.

```
void pkt_trs_A() {

  /* in subframe 3′ and 6′ */
  int CommandPointerI=NumPI;        /* get user ID from high positions of PR */
  i=pri(CommandPointer);      /* get user ID from PRI */
  send_polling_command(i);       /* send command user i*/
  CommandPointerI--;        /* point to next user in reverse order */
  if(transmission_successful){
     c[i]=c[i]+t[i];
     schedul(i);        /* schedule next packet for user i */
     if(Final_Packet) {        /* user i reports final packet */
        ShiftArray(pri, i);         /* drop user i by shifting PRI from PRI[i] */
     }
  }
  else{
     MarkRetr[i]=1;         /* mark re-transmission */
      if(NumPII<NumProbingII){    /*PRII is not full*/
        prii[NumPII]=i;    /*move user ID to PRII */
        pri[NumPI]=0;
     }
  }

  /* in subframe 2′ and 5′ */
  int CommandPointerII=NumPII;
  i=prii(CommandPointer);
  send_polling_command(i);
  CommandPointerII--;
  if(transmission_successful){
     c[i]=c[i]+t[i];

        schedul(i);
        if(Final_Packet) {
           ShiftArray(prii, i);
        }
     }
     else{
        MarkRetr[i]=1;
         if(NumPI<NumProbingI){    /*PRI is not full*/
           pri[NumPI]=i;    /*move user ID to PRI */
           pri[NumPII]=0;
        }
     }
  }
```

(b)

Fig. 9. (*Continued.*) (b) C-like code of packet transmission procedure in available slots.

The voice traffic model used here is based on the work done by Brady [12] with silence suppression. Voice traffic is characterized by a two-state Markov chain model. The two states, ON and OFF, correspond to the talk spurts and idle periods of speech. In the ON state, voice packets are generated at a constant rate. No packet is generated in the OFF state. Time spent in each state is exponentially distributed with mean of 1.35 s for the OFF state and 1.0 s for the ON state [13]. Since voice packets must be delivered in real time, there is a maximum transmission delay allowed; any voice packet that has not been transmitted within 40 ms of its generation time will be dropped at the source. For this study, the packet-dropping rate in this manner must not exceed 1%.

The data users are modeled to generate packets according to a Poisson process. Since data traffic is generally non-real-time traffic, no packets are discarded due to excessive delay. However, too many data users cause the infinite increase of mean delay. There is a breakdown value existing for the system capacity. The breakdown value is detected in the simulations.

All the settings for the simulations are listed in Table I. In this paper, interference between the user and base station is not considered and is therefore not modeled.

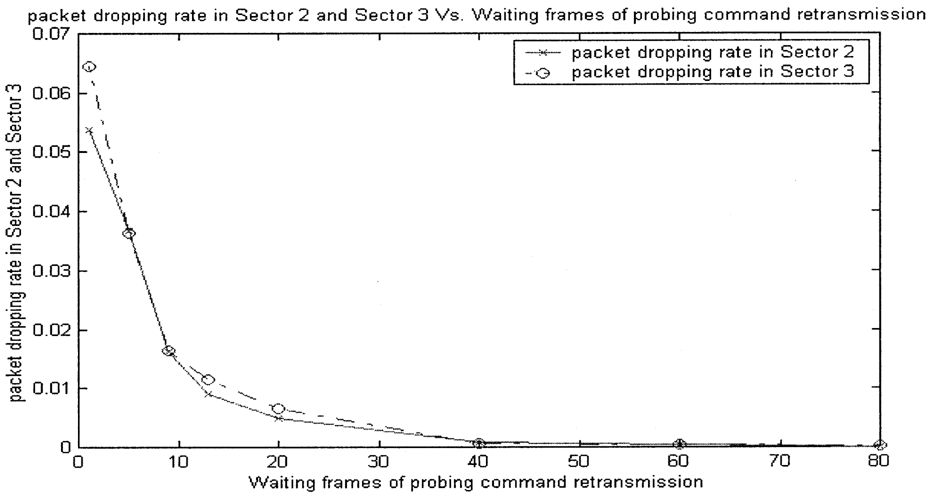


Fig. 10. Packet dropping rate in sectors 2 and 3.

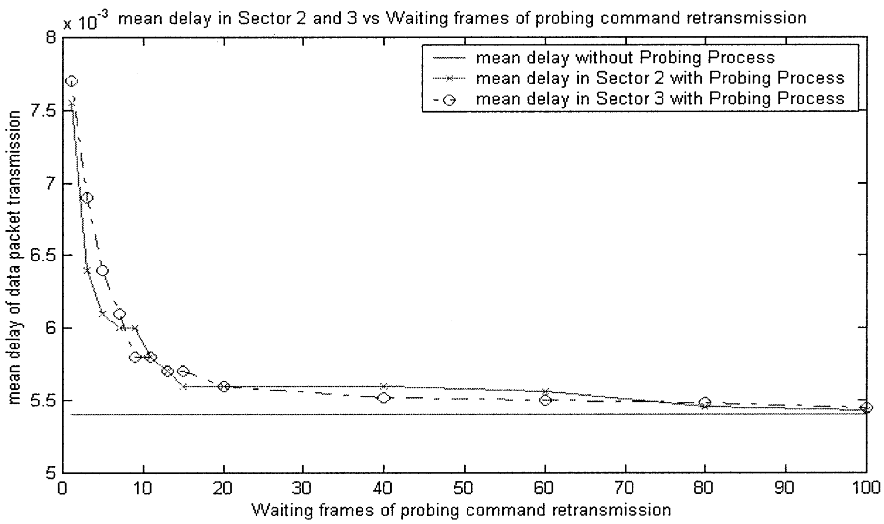


Fig. 11. Mean delay in sectors 2 and 3.

TABLE I
SIMULATION PARAMETERS

Channel Data Rate (Mbps)	4.1472
Frame length (ms)	5
Slots per frame	54
Mini slots per first slot	24
Slots used per sector	17
Slot size (bytes)	48
Command length (bytes)	4
Packet size (bytes)	44
Reservation Request length (bytes)	4
Voice users in other sectors	20
Acknowledgement length (bytes)	4
Average time spent in speaking for voice	1.0
Average time spent in silence for voice	1.35
Allowed transmission delay (ms)	40
Allowed packet dropping probability	1%

Fig. 10 was simulated with 48 voice users in sector 1, 26 voice users in sector 2, and 26 voice users in sector 3. It shows that the packet-dropping rate in sector 2 and 3 decreases with the increase of waiting frames of probing command retransmission. When the value of waiting frames is over 20, the interference becomes rather smaller and the packet-dropping rate is lower than 1%. Fig. 11 also provides a similar result for data traffic. The mean delay of data packet transmission in sectors 2 and 3 is tested when there are 22 data users in sector 1, 15 in sector 2, and 15 in sector 3. In sector 1, the mean generating rate of data packet of each user is one packet per 0.006 s. In sector 2 and 3, the mean generating rate is one packet per 0.008 s. The results show that for the data traffic, the mean delay is very close to the value without probing process when the value of waiting frames is over 20.

Figs. 12 and 13 present the results when the packet transmission is first scheduled in PRI (the available slots in sector 2), then in PRII (the available slots in sector 3) for voice and data traffic with the proposed probing process. This means there is more traffic scheduled in subframes $2'$ and $5'$ than in subframes $3'$ and $6'$. When the number of waiting frames to retransmit probing command is lower, the probing process is executed more frequently, resulting in higher packet dropping in both sectors 2 and 3. However, it helps in accurately determining the avail-

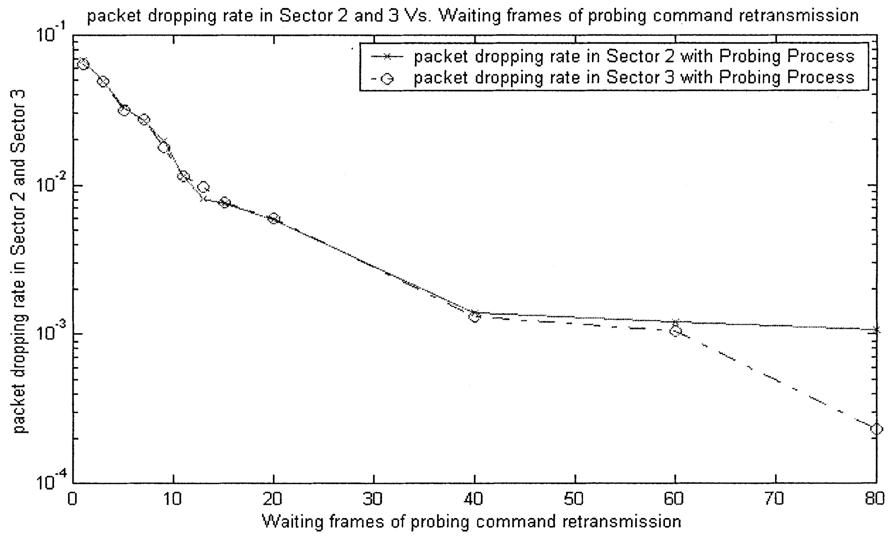


Fig. 12. Packet dropping rate with sector 2 probed before sector 3.

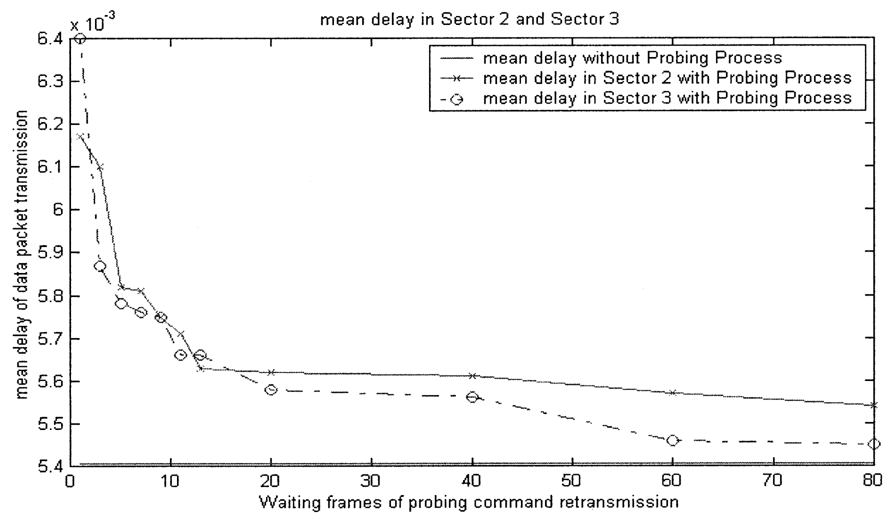


Fig. 13. Mean delay with sector 2 probed before sector 3.

able slots in both sectors 2 and 3. With an increase of waiting frames to retransmit the probing command, the probing process is not executed as frequently, and the packet-dropping rate is reduced. However, available slots are not determined accurately, resulting in a higher packet dropping rate for voice traffic and longer mean delay for data traffic in sector 2 than in sector 3.

From Figs. 10–13, it can be seen that a lower value of waiting frames of probing command retransmission results in the frequent probing packet transmission. This interferes with the communications of sector 2 and 3. To limit this interference, a value of 25 is used as the waiting frames in the following simulations.

To take maximum advantage of the probing process, the traffic in other sectors should be low. The number of users in sector 2 and 3 is therefore taken as 20 for detecting the maximum capacity in sector 1. Fig. 14 shows the packet-dropping rate with respect to the number of voice users in sector 1 with 20 voice users in sectors 2 and 3 separately. If a 1% packet dropping rate is allowed, C-PRMA supplies 34 voice user capacity; however, C-PRMA with probing process provides 44 user capacity.

The aggregate traffic generated by all the data users in a sector is defined as the workload to the sector. The total channel capacity is assumed to be 4.1472 Mbps, so the link capacity of a sector is 1.3824 Mbps as 1/3 of the total channel capacity. In Fig. 15, sector 1 provides 1.08-Mbps workload as 78% of the link capacity that provides the breakdown point of the system with C-PRMA. C-PRMA with probing process can support 1.38-Mbps workload near 100% of the link capacity when the workload of sector 2 and 3 is 0.96 Mbps separately as 69% of the link capacity.

V. Conclusion

This paper has proposed a probing process implemented in C-PRMA with an SRA radio resource allocation algorithm for both downlink and uplink in fixed broadband wireless access networks where a given frequency band is reused in every sector of every cell. It has been shown that this approach provides a way to improve the bandwidth utilization when the slots that were originally assigned to the sectors with lower

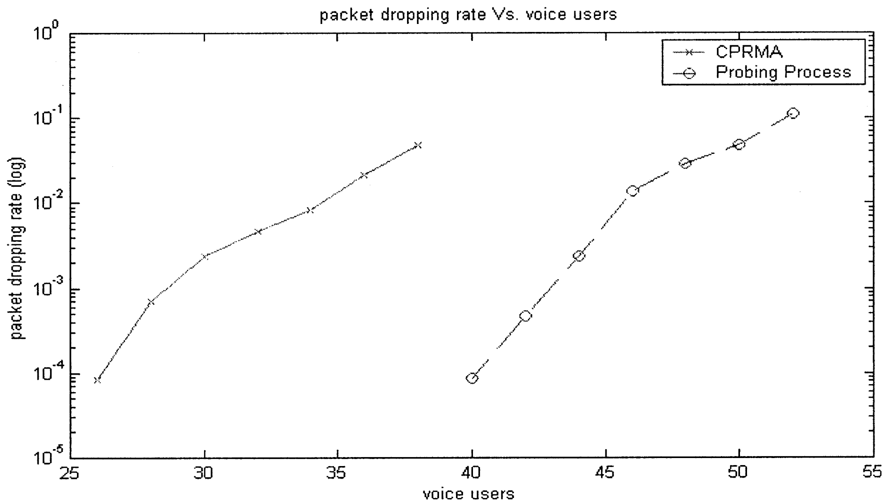


Fig. 14. User capacity for voice traffic in sector 1.

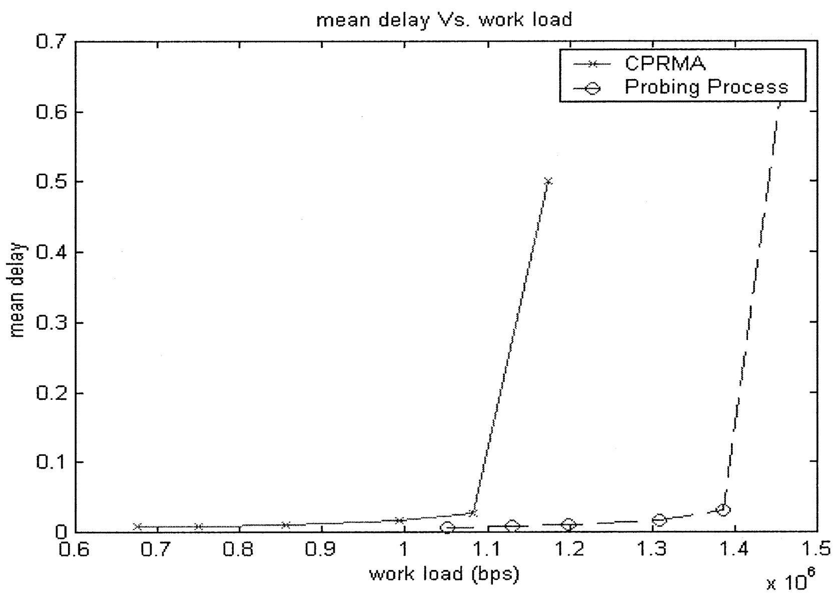


Fig. 15. User capacity for data traffic in sector 1.

traffic load but not utilized under the condition of interference avoidance are allocated to the sectors that own more users. The results demonstrate that C-PRMA with the probing process performs significantly better than C-PRMA when it is used with the SRA method. The performance improvement that the probing process offers can be seen in the simulation. For voice traffic, the probing process provides 23.5% more user capacity when there are 20 voice users in sectors 2 and 3 separately. For data traffic, the system breakdown point increases 22% when the workload of sector 2 and 3 is 0.96 Mbps separately as 69% of the link capacity. It is believed that the proposed probing process will be a significant step forward in improving capacity in fixed wireless access networks.

ACKNOWLEDGMENT

The authors would also like to thank the anonymous reviewers and A. Fapojuwo for their suggestions to improve the quality and presentation of this paper.

REFERENCES

[1] T. S. Rappaport, *Wireless Communications Principles and Practice*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[2] F. Borgonovo, L. Fratta, M. Zorzi, and A. Acampora, “Capture division packet access: a new cellular access architecture for future PCNs,” *IEEE Commun. Mag.*, pp. 154–162, Sept. 1996.

[3] F. Borgonovo, M. Zorzi, L. Fratta, V. Trecordi, and G. Bianchi, “Capture-division packet access for wireless personal communications,” *IEEE J. Select. Areas Commun.*, vol. 14, pp. 609–622, May 1996.

[4] T. K. Fong, P. S. Henry, K. K. Leung, X. Qiu, and N. K. Shankaranarayanan, “Radio resource allocation in fixed broadband wireless networks,” *IEEE Trans. Commun.*, vol. 46, pp. 808–818, June 1998.

[5] K. K. Leung and A. Srivastava, “Dynamic allocation of downlink and uplink resource for broadband services in fixed wireless networks,” *IEEE J. Select. Areas Commun.*, vol. 17, pp. 990–1006, May 1999.

[6] N. Kahale and P. E. Wright, “Dynamic global packet routing in wireless networks,” in *Proc. IEEE INFOCOM’97*, Kobe, Japan, Apr. 1997, pp. 1416–1423.

[7] I. Katzela and M. Naghshineh, “Channel assignment schemes for cellular mobile telecommunication systems: a comprehensive survey,” *IEEE Personal Commun.*, pp. 10–31, June 1996.

[8] X. Qiu and K. Chawla, “Resource assignment in a fixed wireless system,” *IEEE Commun. Lett.*, vol. 1, pp. 108–110, July 1997.

[9] A. Srivastava, N. K. Shankaranarayanan, and K. K. Leung, "Sector-based resource allocation for broadband fixed wireless networks," in *IEEE VTC'98*, pp. 1680–1684.
[10] W. C. Y. Lee, *Mobile Cellular Telecommunications Systems*. New York: McGraw-Hill, 1989.
[11] G. Bianchi, F. Borgonovo, L. Fratta, L. Musumeci, and M. Zorzi, "C-PRMA: a centralized packet reservation multiple access for local wireless communications," *IEEE Trans. Veh. Technol.*, vol. 46, May 1997.
[12] P. T. Brady, "A model for on-off speech patterns in two-way conversation," *Bell Syst. Tech. J.*, vol. 48, no. 7, pp. 2445–2472, Sept. 1969.
[13] W. C. Wong and D. J. Goodman, "A packet reservation multiple access protocol for integrated speech and data transmission," *Proc. Inst. Elect. Eng. I*, vol. 139, no. 6, pp. 607–612, Dec. 1992.

Qiang Wang received the B.S. degree from the Wireless Engineering Department, Southeast University, Nanjing, China, in 1991. He is currently pursuing the Master degree in electrical and computer engineering at Concordia University, Montreal, QC, Canada.

From 1991 to 1998, he was an Engineer with Siemens Business Communication Systems Ltd., China, where he worked with software test and upgrade for switching systems and signaling systems. His research area is in MAC protocols for broadband wireless access networks.

Anjali Agarwal (M'99) received the B.E. degree from Delhi College of Engineering, India, in 1983, the M.Sc. degree from the University of Calgary, Calgary, AB, Canada, in 1986, and the Ph.D. degree from Concordia University, Montreal, QC, Canada, in 1996.

She is an Assistant Professor in the Department of Electrical and Computer Engineering, Concordia University. She was a full-time Lecturer at the University of Roorkee, India, and a Protocol Design Engineer and a Software Engineer in industry, where she was involved in providing specifications and design issues for voice-over-IP support and in the software development lifecycle of real-time embedded software. Her current research interests are in various aspects of real-time and multimedia communication over the Internet and access networks. Most recently, she has been working on MAC and IP telephony for broadband wireless access networks, and on MPLS networks.