Technical Research Report

Interactive Data Services in Wireless Access Networks: Capacity Planning and Protocols

by Xiaoming Zhou,
Majid Raissi-Dehkordi and John S. Baras

CSHCN TR 2005-5
(ISR TR 2005-86)

Interactive Data Services in Wireless Access Networks: Capacity Planning and Protocols

Xiaoming Zhou, Majid Raissi-Dehkordi and John S. Baras

Department of Electrical and Computer Engineering

University of Maryland at College Park, MD, 20742

Email: xmzhou@isr.umd.edu

Abstract

In this paper, we study the capacity planning in wireless access network for interactive data services such as web browsing. A closed queuing model has been developed which can capture the bottleneck effects in both the forward and the reverse channels. The model can be used to calculate the average throughput, the average response time and the number of users the system can support. We evaluate the performance of several MAC protocols such as slotted Aloha, static TDMA, Aloha/periodic stream and combined free demand assignment multiple access (CFDAMA) using realistic web traffic models. Based on the performance evaluation, we propose a new MAC protocol and a new transport layer protocol. Our new MAC protocol called combined polling free demand assignment multiple access (CPFDAMA) explores the correlation between forward channel data packets and reverse channel acknowledgement packets. Our new transport layer protocol called RWBP uses per-flow queuing, round robin scheduling and receiver window backpressure for congestion management. RWBP can eliminate congestion losses inside the wireless networks. Our protocol suite outperforms the proposed protocols in term of both channel utilization and response time. Our results can be used for service providers to dimension their networks and provide quality of service to a certain number of users.

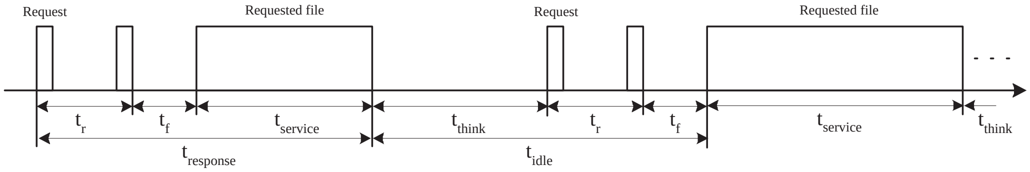


Fig. 1. The interactive user behavior model

I. Introduction

Interactive data services such as web browsing are becoming an indispensable means for people to retrieve information from the Internet. Compared with wireline networks, wireless access networks can provide tetherless services to users even when they are on the move which creates significant opportunities for the service providers. From the providers' perspective, a major concern is how much capacity is needed to provide a certain quality of service to a certain number of users.

In this paper, we assume that the forward channel from the hub to the terminals uses a different frequency from that used the reverse channel i.e. FDD. The forward channel is operated in a TDM fashion and the reverse channel is managed by a multiple access protocol. All the terminals and the hub are synchronized and MAC packets are allowed to be sent only at the beginning of a time slot. We assume there are no other errors except the collisions caused by the more than two packets sent in the same slot through the reverse channel. Whenever there is a collision, all packets involved are destroyed and the users have to retransmit the collided packets which introduces additional delay to the requests.

A typical user's behavior is as following: 1) The user starts a session by sending a request; 2) The request gets through the reverse channel; 3) The hub begins to send the requested file in the forward channel; 4) After the requested file is downloaded, the user spends some time to digest its content; 5) After the $think$ period, the user sends another request to download another file. The ON/OFF cycle keeps on going until the session is finished as shown in figure 1. The performance metric directly influencing the user perception is the file response time denoted by $t_{response}$ in figure 1, which is the time elapses between the epoch when the request is sent and the epoch when the last packet of the file is received [24].

A. Related Work

There are several models proposed in the literature to analyze the ON/OFF interactive data services. The original Engset model is a finite-population, multiple-server model with no buffer space. Heyman [14] adapts the Engset model to analyze the effect of TCP congestion control over an wired access network in which several slow links are multiplexed onto a faster link. Therefore the slow link capacity sets an upper bound of each source’s input rate to the fast link. While in our scenario, the bandwidth of the forward and reverse channel is shared among all active users. So it is possible for a user to achieve the full bandwidth if he is the only active user in the network. Schwefel [23] extends Heyman’s model [14] into packet-level with a single server and modified ON/OFF arrival process. The performance metrics such as throughput per user, aggregate throughput, queue-length distribution and average number of active users can be computed from Schwefel’s model. However this model becomes intractable with many users and power-tailed file sizes due to the complex and extensive computations it requires [14].

Berger and Kogan [4] develops a closed queuing network model for bandwidth dimensioning for elastic data traffic. The model assumes that under heavy traffic the TCP congestion control can ensure each active user to have a packet in the bottleneck queue. Therefore the utilization of the link is exponentially close to one. By letting the number of users goes to infinity, they obtain close-form dimensioning rules for a single bottleneck link.

In [24], the authors use a simple finite-population analytical model for the shared forward channel, which is applied to web browsing over third generation EDGE (Enhanced Data Rates for GSM Evolution) TDMA system. They modified the classic ”machine-repairman” model [28] by adding a delay block which is used to model the overhead delay encountered by the file while being served. Although in a real system the delay overhead could occur at various stages during the file transfer, in this model all the delays are aggregated together into a single delay block for simplicity reasons. The model further assumes that the user performance is constrained only by the forward channel. Therefore bandwidth sharing in the reverse channel has not been addressed.

B. Our Contributions

In this paper, the transmission time and the propagation delay in both channels are explicitly modeled. Therefore bottlenecks in both channels can be captured. The request delay depends on the reverse channel bandwidth and the propagation delay. For example, one retransmission could increase the request delay by one more round trip time. The file service time in the forward channel depends on the file size and the number of requests currently being served. We adopt a closed queuing model which is very general and can be applied to different wireless access works such as cellular networks [22], wireless metropolitan area networks (WMAN) [11][1][30] [26][18][12] and satellite networks [27]. It can provide insight into the dynamics of the network and into the design of the MAC protocols in the reverse channel. From this model, we can calculate the important performance metrics such as the utilization in the forward channel, the file response time and average user throughput etc.

II. System Model

A single user behavior model is shown in figure 1 and the system model is shown in figure 2. There are N identical ON/OFF users in this closed queuing network. The reverse channel is a multiple access channel and the request delay includes the request transmission time, the reverse channel propagation delay and possibly additional delay caused by retransmissions. Once the request is received, the requested file will be sent over the forward channel. We assume per user queuing is maintained at the hub and a processor sharing discipline such as round robin is used in the forward channel. Therefore the service time $t_{service}$ of a requested file depends on its file size, number of files currently being served and the forward channel bandwidth. File sizes are independent and identically distributed with an average length

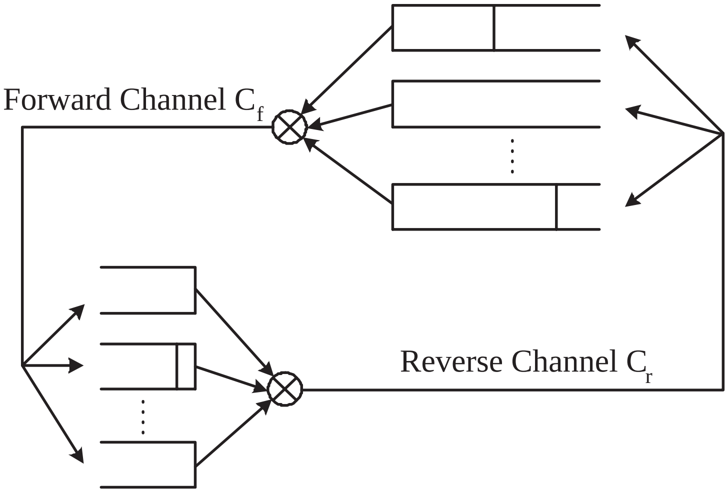


Fig. 2. Closed queuing network model for interactive data services in wireless access networks

of $E[L_f]$ bits. The propagation delay in the forward channel is t_f. The digest time of a user t_{think} is independent of the file service time $t_{service}$. The bandwidth in the forward and reverse channel is C_f bps and C_r bps respectively.

In our analytical model, we lump the think time, the request delay in the reverse channel and the propagation delay in the forward channel into a single delay called t_{idle} i.e. $t_{idle} = t_{think} + t_r + t_f$ as shown in figure 1. It has been shown in [24][14] that the system performance measures are insensitive to distributions of the file sizes and idle times except through their means. Therefore the results are the same as when the distributions are exponential. In the following we will develop a Markov chain by assuming the idle time and service time are exponentially distributed.

Let us denote by $K(t)$ the number of files currently being received in the forward channel at time t. $K(t)$ is a stochastic process which takes values in the state space $S = \{0, 1, 2, ..., N-1, N\}$. When $K(t)$ equals k where $1 \leq k \leq N$, each of the k files is served at a rate $\frac{C_f}{k}$ bps. Since the files are exponentially distributed with mean file size $E[L_f]$ bits, each file is finished at the rate of $\frac{C_f}{k*E[L_f]}$ [21]. Therefore the rate anyone of them finished is given by $\frac{C_f}{E[L_f]}$.

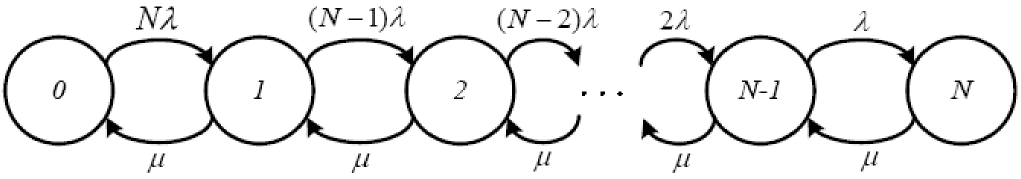


Fig. 3. The state diagram of the finite-population queuing system

The Markov chain is shown in figure 3 in which:

$$\mu = \frac{C_f}{E[L_f]} \tag{1}$$

$$\lambda = \frac{1}{E[t_{idle}]} \tag{2}$$

The steady state probabilities of $K(t)$ are given by the following equation:

$$P_k = P_0 \cdot \rho^k \cdot \frac{N!}{(N-k)!} \tag{3}$$

where

$$\rho = \frac{\lambda}{\mu} \tag{4}$$

$$P_0 = \frac{1}{\sum_{k=0}^{N} \rho^k \cdot \frac{N!}{(N-k)!}} \tag{5}$$

The utilization of the forward channel is given by:

$$U = 1 - P_0 \tag{6}$$

In the closed queuing network, the total number of users in the system is N, the average delay in the system is $E[t_{response} + t_{think}]$ as shown in figure 1 and the average throughput i.e. average number of users entering/leaving the system is $\mu \cdot U$ [24]. By applying Little's law, we have

$$E[t_{response} + t_{think}] = \frac{N}{\mu \cdot U} \tag{7}$$

Therefore the average file response time is given by

$$E[t_{response}] = \frac{N}{\mu \cdot U} - E[t_{think}] \tag{8}$$

and the average user throughput is defined as follows:

$$E[r] = \frac{E[L_f]}{E[t_{response}]} \tag{9}$$

From figure 1, we can see that $t_{response} = t_r + t_f + t_{service}$. Therefore $E[r]$ takes into account the reverse channel access delay t_r and the forward channel propagation delay t_f in addition the file service time $t_{service}$. Another important metric is the average service rate denoted by $E[r^{'}]$

$$E[r^{'}] = \frac{E[L_f]}{E[t_{service}]} \tag{10}$$

$E[r^{'}]$ gives the average service rate of a file since the user receives its first packet. In the following, we will show that we can calculate the distribution of the service rate which can be used to provide the probabilistic bounds.

From an outside observers' point of view, the probability of k users currently served in the forward channel is give by equation 3. However from a user inside the system we need to consider the batch effects [29], the probability of a user being a member of k currently begin served users is given as following

$$P_k^{'} = \frac{k \cdot P_k}{\sum_{k=1}^{N} k \cdot P_k} \tag{11}$$

Also note that

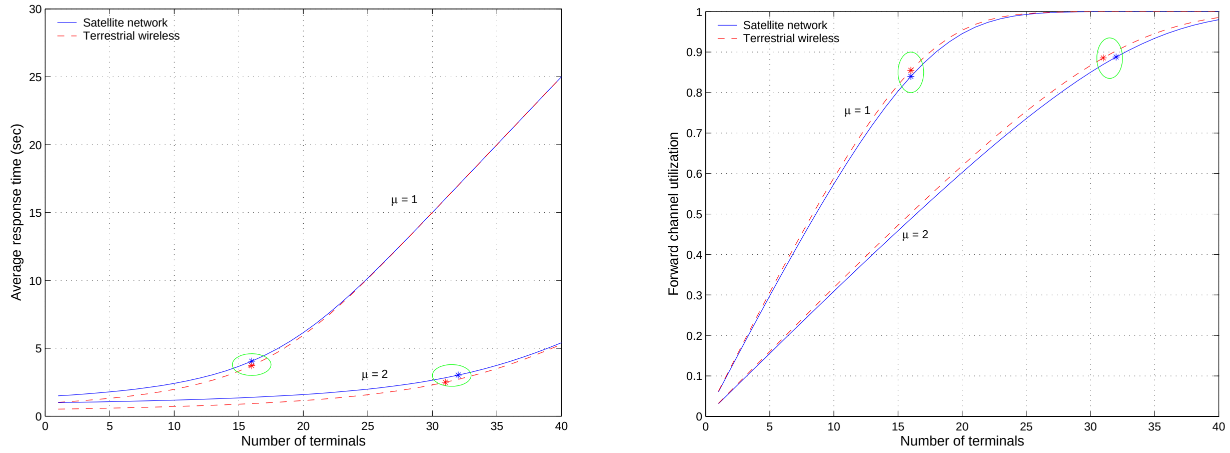
$$P\{r^{'} = \frac{C_f}{k}\} = P_k^{'} \tag{12}$$

$$E[r^{'}] = \sum_{k=1}^{N} \frac{C_f}{k} \cdot P\{r^{'} = \frac{C_f}{k}\} \tag{13}$$

From the above two equations, the average service rate each user receives and probabilistic estimates about it.

III. Analytical and Simulation Results

In the section, we will show how the system performance depends on the forward and reverse channel parameters such as bandwidth, delay and terminal population. We evaluate the performance of several proposed MAC protocols in the reverse channel. After thorough analysis of the forward and reverse



(a) Average file response time

(b) Forward channel utilization

Fig. 4. Average file response time and forward channel utilization for different terminal populations

channel traffic, a new MAC protocol called CPFDAMA is designed which outperforms existing MAC protocols in term of both channel utilization and file response time.

A. Bottleneck in the Forward Channel Only

In this section, we release the reverse channel bandwidth constraint so that it only introduces a constant propagation delay. Figure 4(a) shows the file response time in terrestrial wireless networks and satellite networks with one way propagation delay of 0.01 ms and 250 ms respectively. The average file size is 64 KB and the forward channel bandwidth is 512 kbps or 1024 kbps correspondingly $\mu = 1$ or $\mu = 2$. The average think time is 15 seconds. These parameters are chosen based on the web traffic models developed in [20][6][2][25][10].

When there is only one active user in the system, there is no queuing and the file response time equals the average service time in the forward channel plus the two way propagation delay. As the number of users N keeps on increasing, the forward channel utilization approaches utility as shown in figure 4(b). Let's define N^* as following:

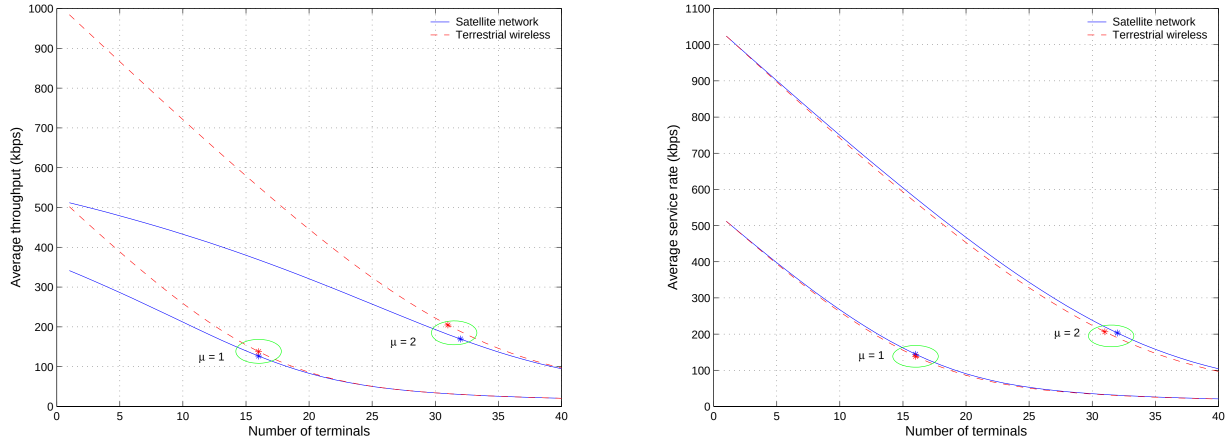
$$N^* = 1 + \frac{\mu}{\lambda} \tag{14}$$

N^* is called the *saturation number* [28]. As long as the terminal population N is less than N^*, the response time only increases gradually with N due to the statistical multiplexing gain. However when N is greater than N^*, the system soon becomes overloaded and the response time increases almost linearly with N. With the same file service rate μ in the forward channel, the response time and utilization are very similar to each other in terrestrial wireless and satellite networks. When the service rate is doubled, the saturation number N^* is also doubled and the response time becomes smaller when the terminal population is less than N^*.

Figure 5(a) shows the average throughput in the forward channel for different terminal populations. It shows that the average throughput is much smaller in satellite networks than in terrestrial wireless networks when the number of terminals is small. The is because the two way propagation delay becomes a significant component in the total response time when the service time is small. After the terminal population is increased above N^*, the service time becomes much larger than the propagation delay therefore both of them have similar throughput.

The average service rate shown in 5(b) is the packet arrival rate since the user receives the first packet in a file. This figure shows that the average service rates in both networks are very close to each other. This figure can be used to dimension the networks. For example, in order to provide average service rate of no less than 400 kbps, forward channel bandwidth of 512 kbps i.e. $\mu = 1$ can serve up to 5 terminals. When the forward channel bandwidth is increased to 1024 kbps i.e. $\mu = 2$, it can provide the same average service rate to 22 terminals. On the other hand, if the terminal population and the average service rate requirement are known, it is also easy to find out how much bandwidth is needed. For example, if there are 15 terminals and the mean service rate of each should be greater than 600 kps, from figure 5(b) we can find out that the forward channel bandwidth should be at least 1024 kbps.

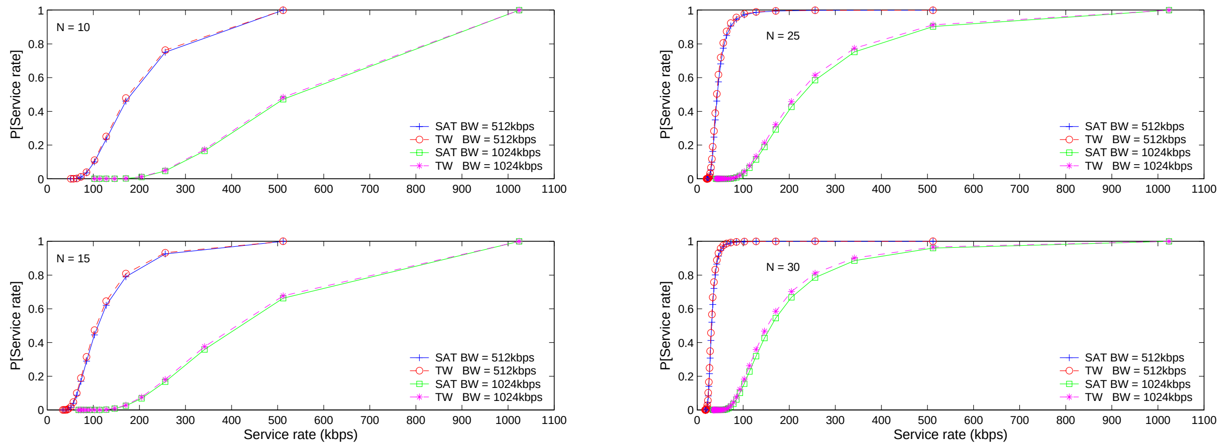
In addition to the average service rate, we can get the rate distributions for different terminal populations through equation 12. Figure 6(a) and figure 6(b) show the service rate distributions for different terminal



(a) Forward channel average throughput

(b) Forward channel average service rate

Fig. 5. Average throughput and service rate in the forward channel for different terminal populations



(a) N = 10 and 15

(b) N = 25 and 30

Fig. 6. Forward channel service rate distributions for different terminal populations

populations which can be used to provide probability based service. For example, figure 5(b) shows that the average service rate is around 600 kbps when the number of terminals is 15 and the forward channel bandwidth is 1024 kbps. The lower plot in 6(a) shows that with probability more than 30% each terminal will receive a service rate of 1024 kbps while with probability of less than 10% the service rate will be less than 256 kbps.

B. MAC Protocols in the Reverse Channel

In this section, we will describe four MAC protocols in the reverse channel which include static TDMA, slotted Aloha and two reservation based protocols Aloha/periodic stream and CFDAMA. Advantages and disadvantages are also pointed out for each protocol in term of efficiency and delay for bursty traffic.

1) Static TDMA: In static TDMA when there are N terminals in the network, each terminal is assigned a time slot every N slots for the life time of the terminal [15]. Therefore the slots assigned to a terminal will be wasted if it has no packets to send.

2) Slotted Aloha: In slotted Aloha, all the terminals are synchronized with each other and MAC packets are sent only at the beginning of a slot. If a MAC packet is received successfully by the hub, an acknowledgment will be sent back to the terminal and the packet is purged from the retransmission buffer. However if a collision happens, the terminal will timeout after one round trip time and retransmit the packet after a random backoff time. An exponential backoff strategy is used here.

In slotted Aloha, the average number of attempts for a successful transmission is e^G, where G is the average load in reverse channel including both new arrivals and retransmissions. The aggregate arrivals is assumed to be a Poisson process. The average access delay of the slotted Aloha denoted by D_{SA} [17] is

$$D_{SA} \ = 1 + t_{prop} + (e^G - 1)(1 + 2 * t_{prop} + t_{bkoff}) \tag{15}$$

Where t_{prop} is the normalized propagation delay and t_{bkoff} is the normalized backoff time. Both are normalized to the packet transmission delay. If the offered load G is heavy, collisions will increase the packet delay dramatically as shown in equation 15. Therefore to deliver the packets with short delay, the slotted Aloha channel bandwidth should be sized such that it operates in the low load region most of the time.

3) Aloha/periodic Stream: Aloha/periodic stream [32] is a reservation based protocol. After new packets arrive at the empty queue of a terminal, the terminal becomes active and an Aloha request will be sent to the hub. After the Aloha request is received, the terminal is assigned periodic bandwidth. If the persistent backlog packets exceed some threshold during the active period, additional bandwidth is requested by piggybacking the request in the data packets. Additional bandwidth is provided until the maximum is attained or the backlog is decreasing. If the terminal hasn't transmitted traffic for a period of time, the terminal will be inactive. The bandwidth is given an inactivity timeout value. If no packet arrives from the terminal during the timeout period, the bandwidth assigned to it will be released.

The Aloha/periodic stream scheme tries to explore the regeneration cycles of the reverse channel traffic. However due to the bursty nature of the traffic, the assigned channel bandwidth to the terminal is wasted if there are no packets arriving at the terminal during the timeout period. The timeout parameter plays an important role in this protocol in term of efficiency and delay. If the timeout is set relatively long, the wasted bandwidth could increase, especially for bursty Internet traffic as in the case we are interested in. On the other hand, if the timeout value is set too small, it will increase the frequency of the request and release cycles. This will increase the packet delay due to the request phase overhead. At the same time more frequent requests will increase the Aloha request channel traffic load and lead to more collisions which eventually will increase access delay dramatically especially in networks with large propagation delay e.g. satellite networks. Besides the timeout setting problem, the period during which the channel release message propagates to a specific terminal is still hold by the terminal and cannot be used by other terminals. This problem becomes more significant with the increase of the propagation delay.

4) CFDAMA: Combined free demand assignment multiple access (CFDAMA) [19] introduces the new concept of free bandwidth assignment. CFDAMA first allocates reverse channel bandwidth to the terminals on a demand basis. However when there is no demand, the scheduler allocates the remaining free bandwidth to the terminals according to some scheduling schemes such as round robin. There are three possible request schemes. Preassigned scheme requires each terminal has a dedicated request channel. Random request scheme allows all the terminals to access the same request channel with the possibility of collisions. The authors argue that piggybacking the request in the data packet header is the most efficient request strategy. The reverse channel bandwidth is managed by a centralized scheduler located at the hub.

When the channel is lightly loaded, the probability of a terminal obtaining free assignment is high therefore small packet delay can be achieved. However the probability of receiving free bandwidth depends on the population of the terminals and the delay performance degrades with the increase of the population size. When the reverse channel is heavily loaded, CFDAMA behaves like a reservation scheme and high channel efficiency can be achieved.

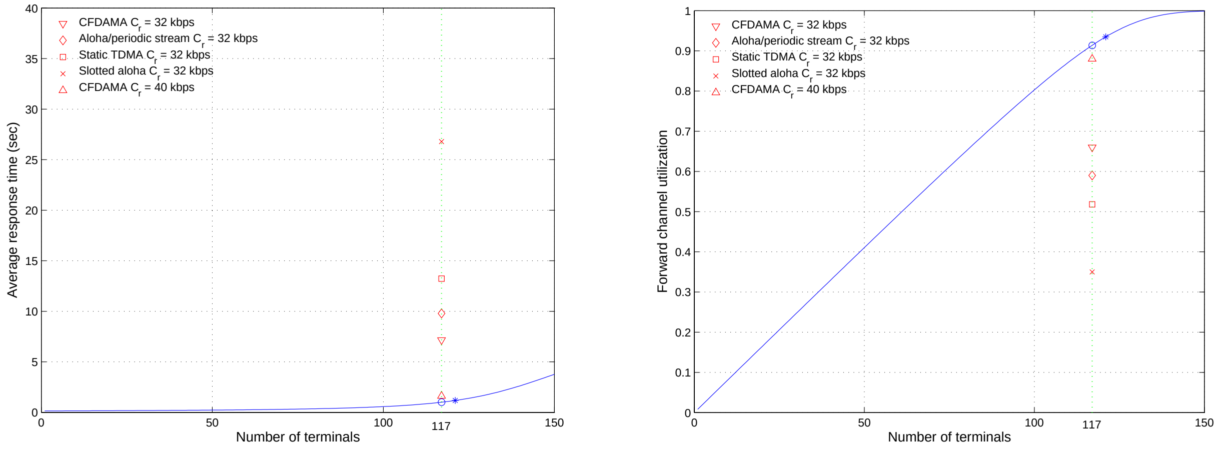
C. Bandwidth Requirement for Request Traffic in the Reverse Channel

In the stable state the file service rate in the forward channel equals the request service rate in the reverse channel. Assume perfect scheduling in the reverse channel i.e. the hub knows the distributed queue status at the terminals without delay, we have the following equation

$$\frac{C_f}{E[L_f]} = \frac{C_r}{E[L_r]} \tag{16}$$

In which C_f and C_r are the forward and reverse channel bandwidth; $E[L_f]$ and $E[L_r]$ are the average file and request sizes. We assume the requests have a fixed size of 432 bytes [20]. In section III-A, we evaluate the system performance of a 4 Mbps forward channel without reverse channel bandwidth constraint. From equation 16, we can get the reverse channel bandwidth should be at least 27 kbps in order not to become the system bottleneck.

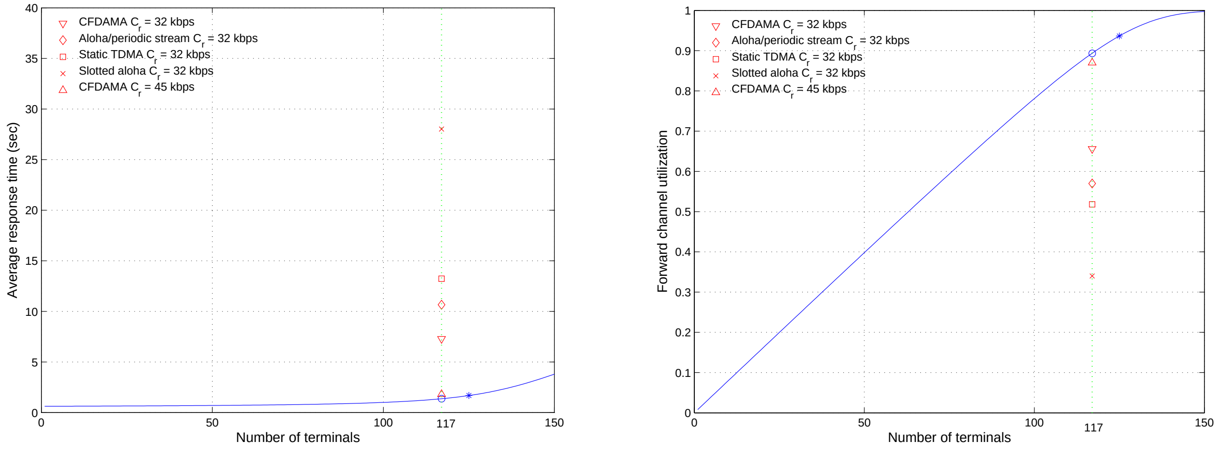
The file response time and the forward channel utilization are shown in figure 7(a) and figure 7(b) for four different MAC protocols. With the reverse channel bandwidth of 32 kbps in terrestrial wireless networks, CFDAMA performs the best which can achieve a forward channel utilization of 66% and file response time of 7.2 seconds. As expected Aloha/periodic stream performs better than static TDMA due to the dynamic release of slots hold by idle users after the timeout. Because the bandwidth is so limited, a lot of collisions occur in slotted Aloha which causes the response time increased dramatically. With 32 kbps bandwidth, the reverse channel is the system bottleneck for the above four MAC protocols. We increase



(a) File response time

(b) Forward channel utilization

Fig. 7. The effects of bottleneck in the reverse channel (terrestrial wireless networks)



(a) File response time

(b) Forward channel utilization

Fig. 8. The effects of bottleneck in the reverse channel (satellite networks)

the reverse channel bandwidth to 40 kbps in terrestrial wireless networks and to 45 kbps in satellite networks so that CFDAMA can achieve a utilization close to 90%, which corresponds to the utilization without reverse channel constraint. Figure 8(a) and figure 8(b) show similar results in satellite networks. The performance of Aloha/periodic stream and CFDAMA is worse in satellite networks compared with terrestrial wireless networks because the queue status in each terminal has been delayed longer.

D. Effects of Acknowledgement Traffic in the Reverse Channel

In the previous section, we only consider the request traffic in the reverse channel. For web browsing, the traffic in the reverse channel also includes the transport layer acknowledgements as shown in figure 10. Assuming perfect scheduling and following the same argument in the previous section, the ACK rate should equal the data rate to achieve high utilization in the forward channel

$$\frac{C_f}{2 \cdot L_{data}} = \frac{C_r}{L_{ack}} \tag{17}$$

in which L_{data} is the forward channel data packet size and L_{ack} is the reverse channel acknowledgement size. The terminal sends an ACK every two data packets received as in TCP. From equation 17, we can calculate the bandwidth required in the reverse channel for a given C_f. For example, if L_{ack} = 40 bytes and L_{data} = 552 bytes which include 40 bytes TCP/IP header and 512 bytes payload. For forward channel bandwidth of 4 Mbps, the bandwidth required for ACKs in the reverse channel is at least 148.4 kbps.

It should be pointed out that there exists a close correlation between forward channel data packets and the reverse channel acknowledgement packets, i.e. the acknowledgement packets are triggered by the data packets. In TCP, an acknowledgement is sent every two data packets are received. Regardless how bursty and unpredictable of the forward channel traffic is, this behavior will always hold and be predictable [7]. This motivates us to design a new MAC protocol CPFDAMA which explores this characteristics as described in the following.

E. CPFDAMA

We classify the reverse channel packets into two categories as in table I by the transport layer port numbers and the packet sizes, and each type of packets is served by a different MAC protocol.

The system model of our protocol called combined polling, free demand assignment multiple access protocol (CPFDAMA) is shown in figure 9. A message coming from the upper layer is first segmented and then buffered in one of the two MAC layer queues. The reverse channel bandwidth controller is located

TABLE I

REVERSE CHANNEL PACKET TYPES AND PROPOSED MAC SCHEMES FOR WEB BROWSING

Packet Type	Proposed MAC scheme
Transport Layer ACKs	Polling
HTTP Requests	CFDAMA

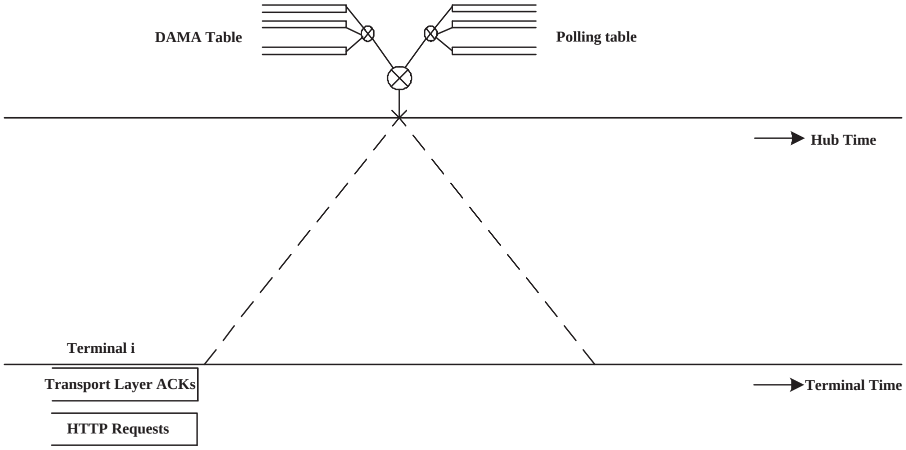


Fig. 9. System model of combined polling, free and demand assignment multiple access (CPFDAMA) protocol for web browsing

at the hub and it assigns reverse channel bandwidth to the terminals based on the requests they've made.

In CPFDAMA, the reverse channel bandwidth is divided into two parts. One part is used to transfer HTTP requests and CFDAMA is used as the MAC protocols. Another part of the bandwidth is used for the transport layer acknowledgements and polling is used for them. The boundary between them is fixed. For example in every five slots, the first to the fourth slots are assigned to ACK traffic and the last slot is assigned to HTTP request traffic.

In the transport layer, the sender polls the receiver to send an acknowledgement by setting a bit in the packet header. This one bit information serves as a reservation request for the ACK in the reverse channel. The MAC layer at the hub will check the packet header of every packet. If the polling bit is set, an entry will be added in the polling table at hub controller. It will try to assign enough slots for transmitting the transport layer acknowledgements. Therefore after a terminal receives a polling data packet, it will receive enough slot assignments to enable it to transmit the acknowledgements. The idea for this scheme is to try to change MAC problem into classical centralized scheduling problem by prediction. The advantages of this approach compared with CFDAMA are as following. The ACK traffic does not need to send bandwidth request in the reverse channel which decreases the bandwidth request delay and at the same time reduces the control channel traffic. Therefore less control channel bandwidth is required. Because there is no bandwidth request overhead, the delay performance of ACK traffic is also improved.

For the HTTP request traffic, it will behave as in CFDAMA. A terminal sends bandwidth requests to the hub and the hub assigns bandwidth to each terminal according to some scheduling scheme. The hub bandwidth controller has two tables. One is for the CFDAMA and the other is for the polling. The CFDAMA table contains the requesting terminal ID numbers and their corresponding requested slots. The polling table contains terminal ID numbers and the number of slots needed to send the ACKs. Whenever a table becomes empty, the hub can assign free bandwidth to terminals in a round robin manner. Following the example in the previous paragraph, let's assume one frame has five slots. The first four assigned to ACK and the last one assigned to HTTP requests. If the CFDAMA tables becomes empty, the controller can assign the last slot in the current frame as a free slot to a terminal and the same slot in the next frame to another terminal and so on. This process continues until there is a new entry added to the CFDAMA table. If a terminal receives a free slot while the HTTP request queue is empty, it can send ACK in that slot to improve the utilization and delay performance. The same approach applies to the polling table.

In section III-C and section III-D, 40 kbps is need for HTTP requests in CFDAMA to achieve a utilization close to 90% in terrestrial wireless networks. The bandwidth needed for ACK traffic is at least 148.4 kbps. Since the slot size is 48 bytes and the size of an acknowledgement is 40 bytes, 8 bytes in a slot has been used for padding. Therefore the bandwidth requirement has to be increased when taking the padding into account. The bandwidth required by the ACK traffic now becomes 48/40*148.4 kbps i.e. 178.08 kbps. In the following simulation, we choose 40 kbps for the HTTP request bandwidth and 200

TABLE II

THE EFFECT OF ACKNOWLEDGEMENTS IN THE REVERSE CHANNEL (TERRESTRIAL WIRELESS NETWORKS)

MAC protocols	Forward channel utilization (percentage)	File response time (sec)	ACK throughput (kbps)	ACK delay (sec)
Slotted Aloha	33%	27.6	76	12.3
Static TDMA	70.1%	4.98	125	5.36
Aloha Periodic	78.0%	4.2	146	4.0
CFDAMA	82.2%	2.18	150	1.6
CPFDAMA	90.0%	1.40	166	0.15

kbps for ACK bandwidth. Therefore the total bandwidth in the reverse channel is 240 kbps. Each reverse channel frame contains 6 slots. One slot is for HTTP request traffic and the other five used for ACK traffic. As in the terrestrial wireless networks, we can get the bandwidth requirement for satellite networks. The satellite networks use a 45 kbps for HTTP request bandwidth and 180 kbps for ACK bandwidth. Each reverse channel frame contains 5 slots. One slot is for HTTP request traffic and the other four used for ACK traffic.

Table II shows the file response time and forward channel utilization in terrestrial wireless networks considering both the request and acknowledgement traffic. Because the acknowledgement traffic increases the queuing delay of the requests, the file response time increases correspondingly compared with figure 7(a). At the same time, the forward channel utilization has decreased due to the smaller request throughput in the reverse channel. From this table, we can see that CPFDAMA can achieve the best performance in both channel utilization and file response time. It should be noted that it also outperforms all the other protocols in term of ACK throughput and ACK delay. Similar results are shown in table III for satellite networks. Therefore CPFDAMA is an efficient MAC protocol for web browsing in access networks.

TABLE III

THE EFFECT OF ACKNOWLEDGEMENTS IN THE REVERSE CHANNEL (SATELLITE NETWORKS)

MAC protocols	Forward channel utilization (percentage)	File response time (sec)	ACK throughput (kbps)	ACK delay (sec)
Slotted Aloha	32.1%	29.8	70	13.7
Static TDMA	70.2%	5.44	125	5.6
Aloha Periodic	76.8%	4.4	141	4.3
CFDAMA	81.3%	3.2	150	2.3
CPFDAMA	88.0%	1.60	155	0.45

IV. WEB BROWSING IN WIRELESS ACCESS NETWORKS: A SYSTEM APPROACH

In the above sections, we implicitly assume that the files are located at the hub which is true if the file servers are located in the same LAN with the hub in which the file transfer delay between the file servers and the hub is negligible. While for web browsing, the hub are connected to the web servers through a wide area network and the delay between them has to be taken into account to get accurate performance results. In this section, we will consider the web browsing in wireless networks from an end-to-end perspective.

To improve web browsing performance in wireless networks, three layers need to be considered. First, at the application layer, a web traffic model is needed to generate realistic source traffic; Second, a transport layer protocol is needed which can solve the problems of TCP over wireless links; Third, a MAC protocol is needed to work efficiently with the given source traffic characteristics and the transport layer. The contribution of our approach is that we are the first to address this problem in a systematic manner rather than focus on a specific layer. Our goal is to analyze web browsing performance in wireless access networks and to improve its performance by designing appropriate protocols.

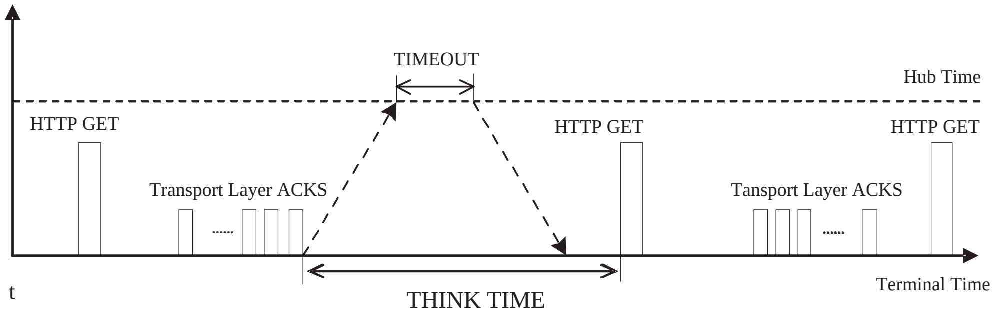


Fig. 10. Reverse channel traffic characteristics for web browsing

A. Web Traffic Modeling

For web browsing, the main traffic loads in the wireless networks are flowing from the Internet to the end users in the forward channel. The traffic flowing in the reverse channel mainly composes some control packets and small data packets [7][20] such as TCP acknowledgement packets and HTTP request packets as shown in figure 10.

A good web traffic model is essential for simulations and experiments to investigate end-to-end performance such as page response time. Recent studies [9][8] show that web traffic shows self similarity, which means that the traffic is bursty over wide range of time scales and therefore has long range dependence in contrast to traditional Poisson traffic models which are short range dependent. Self similar traffic can be modeled as superposition of many independent and identically distributed ON/OFF sources whose ON and OFF times have a heavy tailed distributions. Crovella [9] shows evidence that a number of file distributions on the web exhibit heavy tail distributions, including files requested by users, files transmitted through the network and files stored on the servers.

A random variable X follows a heavy tailed distribution if

$$P[X > x] \sim x^{-\alpha} \;\; as \; x \rightarrow \infty, \; 0 < \alpha < 2,$$

That is, regardless of the behavior of the distribution for small values of the random variable, if the asymptotic shape of the distribution is hyperbolic, it is heavy tailed. The simplest heavy tailed distribution is the Pareto distribution, with probability density function

$$p(x) = \alpha k^{\alpha} x^{-\alpha-1} \;\; where \; \alpha, k > 0, x \geq k$$

and cumulative distribution function

$$F(x) = P[X \leq x] = 1 - (k/x)^{\alpha}$$

The parameter k is the location parameter and it represents the smallest possible value of random variable X. For Pareto distribution, if $\alpha \leq 2$, it has infinite variance; if $\alpha \leq 1$, it has infinite mean. While $1 < \alpha < 2$, the mean of Pareto distribution is $\alpha/(\alpha - 1) * k$ which is of interest to us.

HTTP is a request-response based protocol. There are several empirical web traffic models proposed in the literature [20][6][2][25][10], these models are based on the traffic traces collected either in a local area network or in a wide area network. The elements of an HTTP model are: 1) HTTP request length; 2) HTTP reply length; 3) user think time between retrievals of two successive pages. Mah and Smith [20][25] argue that it is not sufficient to simply transmit data into the network according to these traffic models. This is because these application-dependent but network-independent web traffic models should be layered over TCP so that the sizes and timing of the packets can be modeled accurately. In our web traffic model, a user begins his web browsing session by sending an HTTP request to the web server. The request length will be fixed of 432 bytes. After the server receives the request, it replies with a page which size will be sampled from a Pareto distribution with with $\alpha = 1.01587$ and $k = 4$ KB. The user think time will be modeled also by Pareto distribution [2] with $k = 5$ sec and $\alpha = 1.5$.

B. Improve Web Performance by designing a New Transport Layer Protocol

Considering the interoperability issue, we adopts the connection splitting based scheme [27][3][13] which is currently used in the industry, and design a new transport layer protocol for reliable data transfer over the wireless link.

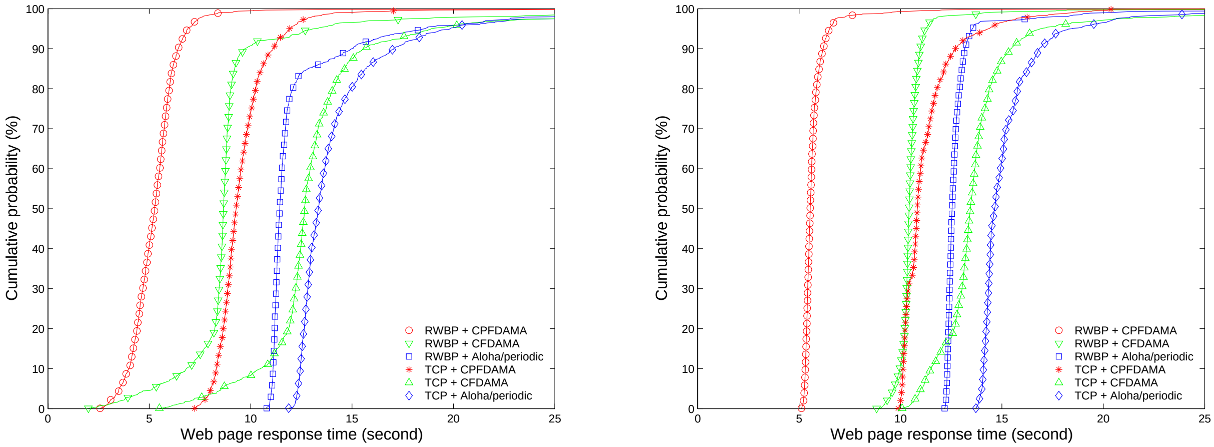
Proxies are put at the boundary of the wireless and wired networks. An end-to-end TCP connection is split into three connections at the proxies. The first one is set up between the server and the upstream proxy; the second is from upstream proxy to downstream proxy; and the third is from the downstream proxy to the client. Normal TCP is used for the server-proxy and proxy-client connections. Receiver Window Backpressure Protocol (RWBP) is designed for the proxy-proxy connection to transfer data over the wireless channel.

The upstream proxy and the hub are connected with a fast link e.g. token ring and the propagation delay between them is negligible. A single FIFO queue is maintained for TCP traffic at the hub. While the proxies keep a per-flow queue and they have to buffer the packets waiting for transmission as well as those packets that have been transmitted but not acknowledged. TCP uses slow start to probe the bandwidth at the beginning of a connection and uses additive increase and multiplicative decrease (AIMD) congestion avoidance to converge to fairness in a distributed manner. RWBP is based on TCP; however RWBP cancels all the congestion control algorithms in TCP and uses per-flow queuing, round robin scheduling [5] and receiver window backpressure for congestion management.

Flow control is done between the downstream proxy and the upstream proxy at the transport layer by using the receiver window. For each RWBP connection, the downstream proxy advertises a receiver window based on the available buffer space for that connection just as in TCP. Similarly flow control is also done between the hub and the upstream proxy at the link layer. The hub advertises a receiver window to the upstream proxy so that its queue will not overflow. Using backpressure, RWBP eliminates congestion losses inside the wireless networks.

C. Performance Evaluations of Web Browsing

In this section, we evaluate the performance of web browsing in a wireless access networks with OPNET. The metric we are interested in is web page response time. There are 117 clients in the network to download web files from 117 Internet servers. The forward channel bandwidth is 4 Mbps and the reverse



(a) Terrestrial wireless networks

(b) Satellite networks

Fig. 11. Web page response time for different transport layer and MAC layer protocols.

channel bandwidth is 240 kbps in wireless networks and 235 kbps in satellite networks. The link bandwidth from each server to the upstream proxy is 4 Mbps; and the link bandwidth from each downstream proxy to its corresponding client is also 4 Mbps. The link delay from each server to the upstream proxy is 40 ms and the link delay from each downstream proxy to its corresponding client is 0.01 ms. The one way propagation delay between the terminals and the hub is 0.01 ms in terrestrial wireless networks and 250 ms in satellite networks. The gateway buffer size is set to the wireless bandwidth delay product [31]. The maximum segment size is 512 bytes.

We evaluate the web access performance for different transport layer protocols and MAC layer protocols by using the realistic web user behavior model described in section IV-A. The results for terrestrial wireless and satellite networks are shown in figure 11(a) and figure 11(b) respectively. For the transport layer protocols, RWBP always outperforms TCP when used for the wireless connections because RWBP doesn’t need to go through the slow start phase and it eliminates congestion losses inside the wireless networks. For the MAC layer protocols, Aloha/periodic stream gives the worse performance therefore it is not suitable for bursty reverse channel traffic. From figure 10 we can see, during the timeout period if there are no packets arriving at a terminal, the assigned channel bandwidth to the terminal is wasted.

Actually the period during which the channel release message propagates from the hub to the terminal cannot be used by any terminal either. While in CFDAMA, the hub can assign the free bandwidth to those terminals with more traffic to send. Therefore CFDAMA can achieve higher efficiency and smaller delay than Aloha/periodic stream. However in CFDAMA, the HTTP requests share the bandwidth with the acknowledgements which could increase their delay. Therefore the throughput of the HTTP requests in the reverse channel decreases correspondingly. While in CPFDAMA, HTTP requests and ACKs are served with different MAC protocols. The ACK traffic causes no interference on the HTTP request traffic. Figure 11(a) and figure 11(b) show RWBP combined with CPFDAMA achieves the best web browsing performance in both short and long delay networks.

In the above, we evaluate web performance over a multiple access wireless channel by using a realistic web traffic model. For the MAC layer protocols, CPFDAMA performs better than Aloha/periodic stream and CFDAMA. For the transport layer, we adopt a new protocol called RWBP which does not have slow start and eliminates congestion losses inside wireless networks. We compare its performance with TCP over the three multiple access protocols. Our results show that RWBP always performs better than TCP in term of web page response time. RWBP combined with CPFDAMA achieve the best performance among all the combinations.

V. Discussions

A. Reducing Reverse Channel Traffic Load

Because RWBP does not use acknowledgements to clock out data packets like in TCP, less frequent acknowledgements are needed in the reverse channel. In RWBP, an acknowledgement is sent when every N data packets are received. By increasing N, we can decrease the acknowledgement frequency. According to the web traffic model in section IV-A, the average page size is 64 KB. Because the data packet size is 512 bytes, on the average each page contains 128 data packets and $128/N$ acknowledgements are needed for each page. One HTTP Get is needed for each page. The size of HTTP Get packets is 432 bytes and

TABLE IV

REVERSE CHANNEL TRAFFIC FOR EACH WEB PAGE WITH DIFFERENT ACKNOWLEDGEMENT FREQUENCY

N	Num of ACKs	Num of HTTP GETS	Total Traffic (bytes)	Normalized Total Traffic
2	64	1	64*40 + 432 = 2992	100%
4	32	1	32*40 + 432 = 1712	57.2%
8	16	1	16*40 + 432 = 1072	35.8%

the size of each acknowledgement packet is 40 bytes. Table IV shows the reverse channel traffic load for different acknowledgement frequency. In [31], we have shown that N can be increased up to eight without degrading the forward channel performance. Only after N is increased beyond sixteen, we see that the throughput in the forward channel begins to decrease. By increasing N from two to eight, the reverse channel traffic can be reduced by about 64%. When the acknowledgement traffic load is reduced, smaller delay can be achieved for both ACKs and HTTP Gets which leads to improved response time performance.

B. Extensions to Support More Applications

In the previous sections, we focus on only one of the Internet applications, specifically web browsing. As shown in table V, our scheme can be extended to support most of the popular Internet applications. Besides web browsing, sometimes a user may produce and transmit packets in a sporadic bursty manner as in Telnet, point of sale transaction. This kind of interactive applications is more sensitive to packet delay. Depending on the propagation delay, the network designer can choose to use different protocols. In terrestrial wireless networks, the terminals can use volume based reservation. Since the propagation delay is small, they can be delivered in a timely manner as long as the requests are scheduled with high priority. However since the propagation delay is long in satellite networks, the reservation overhead becomes too expensive. Random access can be used without violating the delay constraint as long as random access channel bandwidth is sized appropriately. It should be noted that random access is still an option to deliver

TABLE V

REVERSE CHANNEL PACKET TYPES, QOS AND PROPOSED MAC SCHEMES FOR INTERNET APPLICATIONS

Packet Type	QoS	Proposed MAC scheme
Short HTTP GET, Telnet POS, DNS lookup	Delay sensitive	Random access or Reservation (high priority)
Bulk data packets (FTP upload, SMTP)	Delay insensitive throughput sensitive	Reservation
Bulk ACK packets of HTTP, FTP downloads	Both delay and throughput sensitive	Polling

short messages in terrestrial wireless networks.

For an FTP download, it is very similar to a web transfer with a relatively large file size. Occasionally a user may upload a large file using FTP or send an email with a large attachment using SMTP. This kind of applications will generate bulk data packets for the reverse channel and they are not sensitive to the packet delay. Throughput is more important to them. The long HTTP requests also fall into this category. For the last category of the reverse channel traffic i.e. the transport layer acknowledgements, polling can be used as described in CPDAMA.

C. Congestion and Flow Control

In the following, we will show how the MAC protocol proposed above can be integrated with the congestion control in the transport layer to form an end-to-end resource management mechanism. The hub can keep on monitoring the reverse channel status. If congestion is detected in the random access channel, the hub notifies the terminals about the congestion status and the short messages will be enqueued in the reservation queue with some probability depends on the load. This could release the temporary congestion in the random access channel.

If there are a large amount traffic coming from bulk data transfer, the reverse channel could be congested and the reservation queue size will increase. In oder not to overload the reverse channel and cause packet losses, the MAC layer monitors the average queue size with a low pass filter. It will send a congestion notification to the upper layer once the reservation queue size increases above a higher threshold. After the upper layer receives the congestion notification, it should reduce its sending rate. After the congestion is released and the queue size decreases below a lower threshold, the MAC layer can notify the upper layer to speed up. This cross layer design method can be integrated with the transport layer protocol to form a systematic traffic management mechanism.

VI. Conclusions and Future Work

In this paper, we study the capacity planning and protocols design in wireless access networks for web browsing. We've developed a closed queuing model which can capture the bottleneck effects in both forward and reverse channels. Based on the model, we calculate the number of users that can be supported for a given forward channel bandwidth without reverse channel constraint. We then evaluate the system performance of four MAC protocols using realistic web traffic model. In order to improve the web browsing performance, we developed a new MAC protocol and a new transport layer protocol. Our MAC protocol called CPFDAMA explores the correlation between the forward channel data packets and the reverse channel acknowledgement traffic. Our transport layer protocol RWBP uses per-flow queuing, round robin scheduling and receiver window backpressure for congestion management. We have shown through analysis and simulation that CPFDAMA is a very efficient MAC protocol for web browsing in wireless access network. It can maintain high utilization in the forward channel and can deliver the HTTP requests and acknowledgements with higher throughput and smaller delay compared with existing MAC protocol. The results we get in this paper can be use for the network designer to dimension their networks and to provide QoS to certain number of users.

In this paper, we assume all the users are identical. In the future, we would like to investigate users with different channel conditions. We would like to extend our scheme into multiple classes. It is also interesting to combine our scheme with physical layer scheme such as adaptive modulation and adaptive coding [16] together to further improve the system performance.

References

[1] IEEE 802.16-2001. IEEE standard for local and metropolitan area networks - Part 16: Air interface for fixed broadband wireless access systems. *http://standards.ieee.org/getieee802/download/802.16-2001.pdf,* April 2002.

[2] P. Badford and M. Crovella. Generating representative web workloads for network and server performance evaluation. In *ACM SIGMETRICS*, 1998.

[3] A. Bakre and B. R. Badrinath. Implementation and performance evaluation of indirect TCP. *IEEE Transactions on Computers*, 46(3), March 1997.

[4] A. W. Berger and Y. Kogan. Dimensioning bandwidth for elastic traffic in high speed data networks. *IEEE/ACM Transaction on Networking*, 8(5):643–654, October 2000.

[5] Kongling Chang. IP layer per-flow queuing and credit flow control. Technical report, Ph.D. Thesis Harvard University, Division of engineering and applied science, January 1998.

[6] Hyoung-Kee Choi and John O. Limb. A behavior model of web traffic. In *International Conference of Networking Protocol '99*, September 1999.

[7] D. P. Connors and G. J. Pottie. Response initiated multiple access (RIMA), a medium access control protocol for satellite channels. In *IEEE GLOBECOM'00*, 2000.

[8] M. Crovella and A. Bestavros. Self-similarity in world wide web traffic, evidence and possible causes. *IEEE/ACM Transactions on Networking*, 5(6), December 1997.

[9] M. E. Crovella, M. S. Taqqu, and A. Bestavros. *Heavy-tailed probability distributions in the world wide web, Chapter 1 of a practical guide to heavy tails*. Chapman & Hall, 1998.

[10] S. Deng. Empirical model of WWW document arrivals at access link. In *IEEE ICC'96*, June 1996.

[11] C. Eklund, R. B. Marks, K. L. Stanwood, and S. Wang. IEEE standard 802.16: A technical overview of the wirelessMAN air interface for broadband wireless access. *IEEE communications magazine*, June 2000.

[12] V. Erceg and et al. A model for the multipath delay profile of fixed wireless channels. *IEEE Journal on Selected Areas in Communications*, 17(3):399–410, March 1999.

[13] T. R. Henderson and R. H. Katz. Transport protocols for Internet-compatible satellite networks. *IEEE J. Select. Areas Comm.*, 17:326–344, February 1999.

[14] D. P. Heyman, T. V. Lakshman, and A. L. Neidhardt. A new method for analyzing feedback-based protocols with applications to engineering web traffic over the Internet. In *ACM SIGMETRICS'97*, 1997.

[15] A. Hung, M.-J. Montpetit, and G. Kesidis. ATM via satellite: a framework and implementation. *Wireless Networks*, 4(2), April 1998.

[16] T. Keller and L. Hanzo. Adaptive modulation techniques for duplex OFDM transmission. *IEEE Transaction on Vehicular Technology*, 49(5):1893–1906, September 2000.

[17] Leonard Kleinrock and Fouad A. Tobagi. Packet switching in radio channels: Part I-Carrier sense multiple access modes and their throughput-delay characteristics. *IEEE Transaction on Communications*, 23(12):1400–1416, December 1375.

[18] I. Koffman and V. Roman. Broadband wireless access solutions based on OFDM access in IEEE 802.16. *IEEE communications magazine*, April 2002.

[19] Tho Le-Ngoc and I. Mohammed Jahangir. Performance analysis of CFDAMA-PB protocol for packet satellite communications. *IEEE Transactions on Communications*, 46(9), September 1998.

[20] Bruce A. Mah. An empirical model of HTTP network traffic. In *IEEE INFOCOM'97*, 1997.

[21] D. Miorandi, A. A. Kherani, and E. Altman. A Queueing Model for HTTP Traffic over IEEE 802.11 WLANs. In *To appear in Proc. of ITC Specialist Seminar on Performance Evaluation of Wireless and Mobile Systems*, sep 2004.

[22] T. S. Rappaport. *Wireless communications: Principles and practice, 2nd edition*. Prentice Hall, 2002.

[23] H.-P. Schwefel, D. P. Heyman, M. Jobmann, and D. Hoellisch. Accuracy of TCP performance models. In *Internet Performance and Control of Network Systems III, SPIE*, 2001.

[24] N. K. Shankaranarayanan, Zhimei Jiang, and Partho Mishra. Performance of a shared packet wireless network with interactive data users. *ACM Mobile Networks and Applications*, 8:279–293, 2003.

[25] F. D. Smith, F. Hernandez, K. Jeffay, and D. Ott. What TCP/IP protocol headers can tell us about the web. In *ACM SIGMETRICS*, 2001.

[26] Ivana Stojanovic, Manish Airy, Huzur Saran, and David Gesbert. Performance of TCP/IP over next generation broadband wireless access networks. In *The 4th International Symposium on Wireless Personal Multimedia Systems*, September 2001.

[27] Hughes Network System. DirecPC web page. In *http://www.direcpc.com*, December 2002.

[28] K. S. Trivedi. *Probability and statistics with reliability, queueing and computer science applications, 2nd edition*. John Wiley and Sons, 2002.

[29] R. W. Wolff. *Stochastic modeling and the theory of queues*. Prentice Hall, 1989.

[30] K. Wongthavarawat and A. Ganz. Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems. *International Journal of Communication Systems*, 16:81–96, 2003.

[31] Xiaoming Zhou and J. S. Baras. Congestion management in access networks with long propagation delay. Technical report, CSHCN TR 2003-23, http://www.isr.umd.edu/CSHCN, July 2003.

[32] Xiaoming Zhou, N. Liu, and J. S. Baras. Web access over a multiple accesss channel: evaluations and improvements. In *IEEE ICC'2004*, June 2004.