애드혹 네트워크 상에서 backoff 알고리즘 수정에 의한 서비스 차별화

(Service Differentiation in Ad Hoc Networks by a Modified Backoff Algorithm)

강 승 석 † 김 진 ††
(Seoung-Seok Kang) (Jin Kim)

요 약 여러 휴대용 이동통신 장비들은 점점 상업적으로 성공하고 있으며, 이동중의 사용자들에게 유용한 서비스를 제공한다. 인터넷의 확장에 따라 이동통신장비들은 텍스트나 멀티미디어 데이타등과 같은 다양한 형태의 데이타를 요구하게 된다. 요구하는 데이타의 형태나 사용자의 등급에 따라서 데이타 서비스에 대한 처리 또한 달라져야 한다. 무선네트워크 상에서 서비스 차별화의 구현은 매우 어려운데, 그 이유는 무선장비 자체의 이동성과 무선채널의 충돌 때문이다. 무선채널의 충돌은 Binary Exponential Backoff(BEB) 알고리즘을 사용하여 해결할 수 있다. 우리는 금, 은, 동으로 명명된 세 가지 종류의 데이타의 흐름을 지원할 수 있는 backoff 알고리즘의 수정에 대하여 논한다. 예를 들어, 금급 데이타 흐름은 가장 높은 우선순위를 가지고 있어 요구되는 목표대역을 만족시켜야 하고, 은급 데이타흐름은 동급에 비하여 충분히 많은 양의 대역폭을 제공하도록 해야 한다. *Ad Hoc* 네트워크에서 사용되는 두개의 트랜스포트 프로토콜인 UDP와 TCP의 병행 사용은 backoff 알고리즘의 수정을 매우 어렵게 한다. UDP와 TCP의 서로 다른 특성 때문에 이를 해결하기 위해 각각의 프로토콜에 서로 다른 형태의 수정된 backoff 알고리즘을 제안한다. 제안한 알고리즘이 트랜스포트 프로토콜의 형태에 관계없이 서로 다른 급의 데이타흐름 간의 서비스를 차별화시킴을 모의실험을 통하여 보였다.

키워드 : backoff 알고리즘, backoff 알고리즘의 수정, *Ad Hoc* 네트워크

Abstract Many portable devices are coming to be commercially successful and provide useful services to mobile users. Mobile devices may request a variety of data types, including text and multimedia data, thanks to the rich content of the Internet. Different types of data and/or different classes of users may need to be treated with different qualities of service. The implementation of service differentiation in wireless networks is very difficult because of device mobility and wireless channel contention when the backoff algorithm is used to resolve contention. Modification of the binary exponential backoff algorithm is one possibility to allow the design of several classes of data traffic flows. We present a study of modifications to the backoff algorithm to support three classes of flows: gold, silver, and bronze. For example, the gold class flows are the highest priority and should satisfy their required target bandwidth, whereas the silver class flows should receive reasonably high bandwidth compared to the bronze class flows. The mixture of the two different transport protocols, UDP and TCP, in *ad hoc* networks raises significant challenges when defining backoff algorithm modifications. Due to the different characteristics of UDP and TCP, different backoff algorithm modifications are applied to each class of packets from the two transport protocols. Nevertheless, we show by means of simulation that our approach of backoff algorithm modification clearly differentiates service between different flows of classes regardless of the type of transport protocol.

Key words : backoff algorithm, backoff algorithm modification, *Ad Hoc* network

† 비 회 원 : Michigan State University, CSE
sskang@cse.msu.edu
†† 정 회 원 : 한림대학교 정보통신공학부 교수
jinkim@hallym.ac.kr

논문접수 : 2003년 6월 2일
심사완료 : 2004년 3월 12일

1. Introduction

The contention-based medium access control mechanism in an *ad hoc* network distributes the wireless bandwidth for all participating mobile hosts in a statistically fair manner. Every host has an equal chance to use the wireless medium for packet transmission when the medium is idle. If two or more hosts try to utilize the medium at the same time, contention takes place and packet collisions are expected. The binary exponential backoff algorithm is introduced to resolve the contention. When a host experiences a collision, the host delays its packet transmission for a random amount of time and sends the packet in case the medium is not busy. The backoff mechanism is shared by all flows generated from the mobile hosts in *ad hoc* networks and the flows are treated equally in the network. Mobile hosts in wireless networks send and receive several types of data including delay-sensitive multimedia data and delay-tolerant text and binary data. While reliable transmission is important for delay-tolerant data such as FTP and email, timeliness is essential for delay-sensitive multimedia and real time data. Some levels of quality of service(QoS) are required to satisfy adequately various service request types. However, it is difficult to provide QoS in wireless *ad hoc* networks compared to wired networks due to device mobility, high error rates, and contention-based medium access control.

Many researchers have studied the provisioning of the quality of service in wireless networks. TDMA-based wireless networks[1-3] employ a reservation mechanism in which a supervising controller allocates time slots to each mobile host. If a mobile host receives more time slots than others, the host will transmit more packets. In addition, the time slots are exclusively used, resulting in guaranteed bandwidth for all hosts. The reservation is maintained in the adjacent cells in order to support mobility. On the contrary, in contention-based CSMA networks, it is not easy to provide QoS because of the competition of wireless resources and the backoff algorithm. Since there is no centralized control mechanism, every host has a statistically equal chance to use the wireless medium and to transmit packets.

Chen *et al.*[4] provide QoS in a CSMA-based network by having a sender adapt to the changing environment by leveraging the quality of voice data using feedback from its receiver. When both hosts are unable to maintain the minimum QoS, the sender executes voice-to-text conversion and transmits text data for voice that is restored at the receiving side. The paper in [5] focuses on the different treatments between two classes of packets at the link layer. Real-time packets are serviced depending on the availability of the bandwidth, and non-real-time packets are serviced based on the availability of buffer space in the non-real-time packet queue. The work in [6-9] proposes the fairness model in contention-based wireless networks. Network-wide fairness may be achieved without global coordination. This mechanism supports some notion of "weighted fairness" in which a flow with larger weight receives better treatment than the flow with smaller weight. For example, a flow with weight 2 receives twice as much bandwidth as a flow with weight 1. The weighted fairness model may be interpreted as a time slot model of TDMA-based network and help to implement the service differentiation of flows. However, it is difficult to define a "unit weight," which should be the same in any network environment. Nandagopal *et al.*[10] realizes service differentiation of several classes of flows by reconfiguring the data sending rates in case of congestion detected at the receiver. The flows with higher weight reduce their sending rate by a small amount, whereas the flows with lower weight throttle the rate drastically. Several researchers have studied the modification of the backoff algorithm in order to provide the service differentiation in wireless networks.

Deng and Chang[11] propose a priority scheme that higher priority host has shorter backoff time by reducing the amount of interframe space (IFS). The authors in [12] suggest three parameters by which wireless hosts receive statistical or absolute QoS guarantees. First, different backoff increase functions are applied to hosts with different priorities. For example, the host having the highest priority multiplies by 2 for the next backoff range, the medium-priority host by 6, and the lowest by 8 when there is a contention. Second, the host with different priority level waits for different amount of

DIFS time before transmitting its RTS or data packet. Third, the host with having different priority is allowed to use different maximum frame length transmitted at once.

When a host receives a packet that exceeds the maximum frame length assigned to a given priority, it may drop the packet or fragment it into smaller packets. The authors in [13] also employs the priority scheme in wireless networks. When a host has a packet to transmit, it enters the priority contention period in which the highest priority host emits the longest black-burst signal. After a DIFS period, the host senses that the channel is idle and acquires the smallest ID, while other hosts waits for sensing the idle channel toreceive their ID. Hosts with higher-ID (lower priority) frames are blocked until all lower-ID (higher priority) frames have completely transmitted their frames.

The differentiated service architecture [14] provides that certain users and applications may receive better service than others at a higher cost. The architecture focuses not on individual packet flows, but on the sets of flows with similar service requirements. After negotiating the service level agreement, a marker inserts the service class of packets in the DS[15] field of the IP packet headers. According to [15], the DS field is defined to supersede the existing definitions of the IPv4 TOS octet and the IPv6 Traffic Class octet. The DS field decides a routing behavior from one router to the next. In *ad hoc* networks, each and every host plays a role of a router. Each router examines incoming packet headers and forwards them depending on the header information.

Service differentiation in contention-based wireless *ad hoc* networks may be achieved by controlling the behaviors of the backoff algorithm, which does not require any feedback from receivers. Every flow belongs to exactly one class. Each class defines its own backoff definition. The sender of a flow marks its class to all outgoing packets in the DS field of their IP headers. Before forwarding packets, a host ready to transmit packets in the *ad hoc* network checks the wireless medium. If no host uses the medium, the host forwards a packet to its neighboring host. When the medium is in use, the hosts back off their counters and wait a random amount of time for the next transmission attempt. To perform a backoff action, each host looks up the incoming packets and applies different backoff algorithms depending on the classes of packets. Modification of the backoff algorithm enables the design of several class definitions.

This paper consists of the following sections. In section 2, the binary exponential backoff algorithm is redefined. The characteristics of the backoff parameters and three service classes for both UDP and TCP are defined using the backoff parameters in section 3. The simulation environment is described in section 4. Simulations performed from different number of sources illustrate the results of the service differentiation among the three classes in section 5. Finally, conclusions and future work follow in section 6.

2. Modification of Backoff Algorithm

In RTS/CTS based CSMA networks, a binary exponential backoff (BEB) algorithm is used to resolve contentions. The sending host transmits RTS packets during a *random slot time*. The sender retransmits if and only if the host does not receive a CTS packet in response to its RTS. It randomly selects, with uniform distribution, an integer number in (0, BO-1) where BO is a backoff counter, and schedules to retransmit in the chosen random number of slot times. When a CTS is received after an RTS, the backoff counter is reset to the defined minimum value, BO_{min}.

When a CTS is not received after an RTS, BO is adjusted by

$$BO = {}_{\text{minimum}}(BO * 2 + 1, BO_{\max}) \qquad (1)$$

where BO_{max} is the maximum number of the backoff counter. Hosts ready to send packets check whether their wireless medium is idle. When the medium is busy, hosts that are trying to send RTS or data packets generate random numbers. Then they compute the waiting time during the contention period by the function

$$wtime = (random() \bmod BO) * SlotTime \qquad (2)$$

According to the 802.11 specification [16], the predefined values of the constants are as follows :

BO_{min} = 31, BO_{max} = 1023, and SlotTime = 20 μsec.

To provide service differentiation in a contention-based wireless network, the standard binary exponential backoff algorithm may be modified. When a host generates a random number during the contention period, the following equation, which is the modification of equation 2 is applied:

$$wtime = \{ A + (random() \bmod BO) / B \} * SlotTime \quad (3)$$

The default values of A and B are zero and one, respectively. That is, when the default values are assigned to the above equation, the resulting equation will be the same as the traditional BEB. It is possible to consider that some prioritized flows may generate *wtime* independent of BO ranged from zero to A-1. When the value B becomes larger than one, the generated random number becomes smaller, which leads to a comparatively smaller waiting time. Whenever a CTS is not received after an RTS, the backoff counter assignment equation 1 can be rewritten as

$$BO = {}_{minimum}(\text{round}(BO*C) + D, BO_{max}) \quad (4)$$

The defaults of C and D are two and one, respectively. When the value C becomes small, the increment of the backoff value becomes small. Moreover, if the value C is less than one, the value of BO decreases. Similar results are expected when the value D becomes negative. Because the value C can be a positive real number, the multiplication operation may generate a real number. In contrast, the value BO should be an integer. As a result, the round function will round off the real number to an integer.

One additional equation is defined as follows:

$$wtime = (random() \bmod A) * SlotTime \quad (5)$$

This equation will generate a random waiting time with uniform distribution. The waiting time is an integer between zero and A-1, which means the generated integer waiting time does not depend on the backoff counter (BO). Any class of flows using this equation receives the highest priority to forward packets. In summary, the combination of the four variables produces several possible ways of backoff behaviors. The value of the variable A should be a positive integer. The values B and C can be any positive real number. The value D will be any number. The following section describes the behaviors depending on the values of the four variables, and proposes some suggested values for the variables depending on the type of flows.

3. Service Class Definitions

In the previous section, three modified backoff equations are introduced. When a backoff counter (BO) value is computed by equation 4, the actual waiting time of a packet is defined by equation 3. If the generated waiting time of packets from a flow f1 is smaller than that from a flow f2 most of the time, it is more likely for a host transmitting packets of the flow f1 to acquire the wireless medium than a host sending packets of the flow f2. In this case, it is said that the class of flow f1 is higher than that of the flow f2. The combination of the four backoff parameters, A, B, C, and D in equations 3 to 5 produces many possible flow classes of waiting time. This paper provides a study of classification both UDP and TCP with three classes called gold, silver, and bronze. The first subsection describes the characteristics of the backoff parameters and the following two subsections give the definitions of the three classes for both transport protocols.

3.1 Characteristics of the Modified Backoff Parameters

The behaviors of the backoff algorithm are controlled by the adjustment of the four backoff parameters. *Ns-2* [17] is used to illustrate the behavior of the modification. The effective wireless transmission range of the simulator is about 250 meters, and the affordable bandwidth is ranged from 1.3 Mbps for pure TCP flows and to 1.7 Mbps for pure UDP flows. Figure 1 shows a simple wireless *ad hoc* network topology. The hosts in the left column are senders, and the hosts in the right are receivers. Each sender generates a TCP flow and the corresponding receiver calculates the number of data packets received within the simulation time. The number of hosts in the simulation varies from 4 to 16 that make 2 to 8 flows. The distance between all neighbor hosts are 10 meters. When a parameter is adjusted to compute the transmission throughput of one flow, the remaining parameters in the flow are set to their defaults. When one parameters in one flow is changed, all the remaining flows keep the defaults throughout the simulation. The simulation runs for 650 seconds and the first 50 seconds of warming-up data are discarded.

Figure 2 displays the bandwidth allocation of a selected flow when A of equation 5 changes from 1 to 32. Only one of the flows uses equation 5, and all other flows use equation 3 with B=1. When the selected flow that uses equation 5 receives the displayed amount of bandwidth, the remaining flow(s) (not shown) share the remaining affordable bandwidth almost equally. If the value A is too small, high competitions occur between TCP packets and ACK packets from the same flow of the modified backoff algorithm. The largest amount of bandwidth is allocated to the specified flow when A=5 in two-flow case, and A=4 in four- and eight-flow cases. As the value A increases, the allocated bandwidth of the flow decreases gradually. When the value A is too large, the overall network performance may degrade because other flows should wait at least 'A' number of slot times and try to transmit packets. The value A is needed to be reasonably small for specialized flows to receive highly prioritized treatment in an *ad hoc* network.

Figure 3 illustrates the bandwidth assignment result of a specific flow that uses equation 3 when the parameter B varies from 1 to 9. The value A in the equation is set to zero for generating results related only with parameter B. Only one selected flow changes the value B. All other parameters are set to their default values for the selected flow as well as for all other participating flows. When B=1, all flows in the network share the bandwidth almost equally. As one of the flows increases its value B, the flow modifying value B receives more bandwidth than the remaining flows. Although the starting points of the graphs are different due to the different number of flows, the three graphs reach above or up to 1 Mbps, and other flows (not shown) experience a scarcity of bandwidth.

Figure 4 shows the behavior of the parameter C of equation 4. When the value C changes from 2 to 0.3, the throughput of a selected flow increases gradually. If the value C becomes too small, the amount of allocated bandwidth decreases due to the

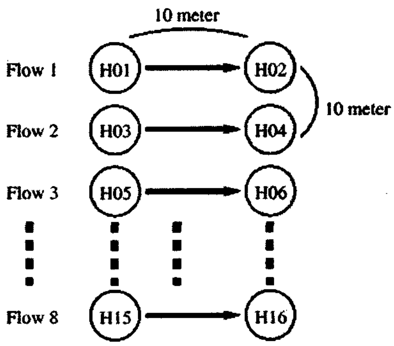


Figure 1 A Network Topology with 8 Flows

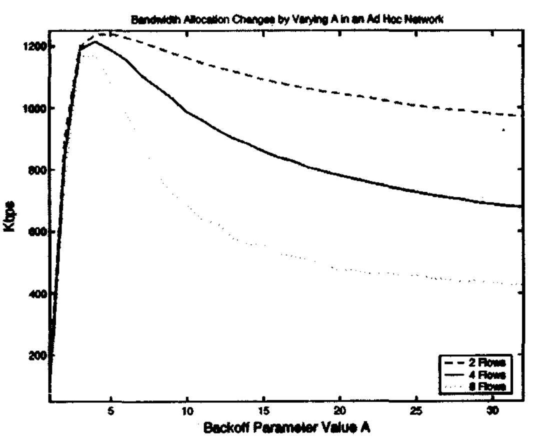


Figure 2 Bandwidth Allocation of 3 Different Number of Flows with Varying Value A

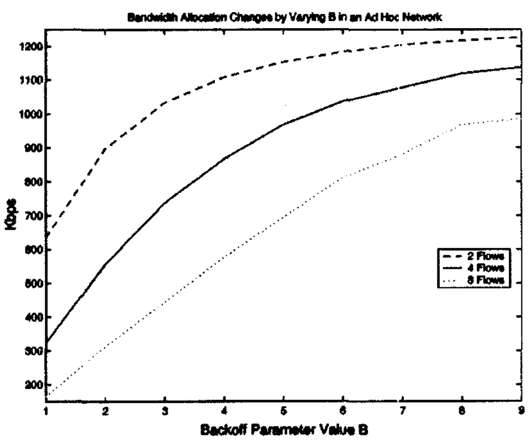


Figure 3 Bandwidth Allocation of 3 Different of Flows with Varying Value B

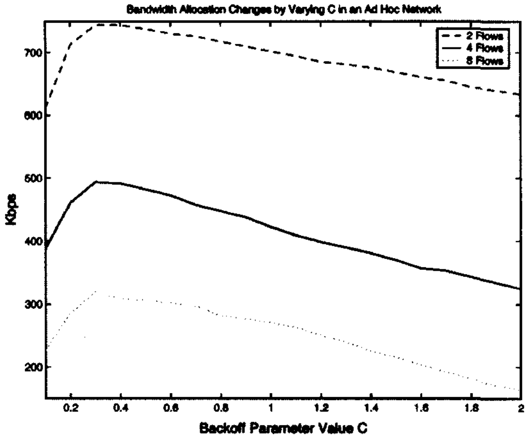


Figure 4 Bandwidth Allocation of 3 Different Number of Flows with Varying Value C

channel competition between the sender's TCP packets and the receiver's ACK packets.

There is one crucial difference between Figure 3 and Figure 4. The graphs in Figure 3 reaches up to the maximum bandwidth supported by the wireless network. However, the graphs in Figure 4 indicate that there are upper bounds due to resetting the backoff counter after successful retransmission. Similar results are produced for parameter D when the value D decreases.

3.2 UDP Flow Classification

The simulation in this paper includes UDP flows that have constant bit rate transmission. That is, even if the highest priority class is assigned to a UDP flow, the flow will not exceed its specified amount of bandwidth usage. This leads us to use equation 5 for the UDP gold class. Each time a CTS packet is not received from its neighboring hosts, the sender's backoff counter is increased and the generated random waiting time will likely be larger than before. When a flow adopts equation 5 the waiting time is not a function of the backoff counter. Using equation 5, the waiting time is a random number with fixed range from zero to A-1. Therefore, UDP gold class flows have the highest priority to win the contention most of the time. By using the simulation results from the previous subsection, the value A is set to 8 because it is reasonably small for the highest priority class of flows.

UDP silver class and bronze class flows compute waiting time using equation 3. The value A will be the same as used in the UDP gold class. The backoff value B of the silver class should be larger than 1. The simulation in this paper sets the value B=5 for the UDP silver class. Even though the value 5 for B may be large, the UDP silver class flow may not too greedy and therefore does not overconsume the wireless bandwidth due to the constant rate transmission. The silver class may be subdivided into several classes by using different values of B, such as 7 for higher silver class and 2 for lower silver class. UDP bronze class flows use the same equation as that of UDP silver class except that the value B=1.

3.3 TCP Flow Classification

Designing UDP class flows are straightforward due to the constant bit rate (non-greedy) characteristics of the wireless bandwidth demand as long as there is sufficient bandwidth for all traffic classes. The UDP packets are transmitted at a constant rate. A highest UDP class flow will not dominate the wireless network. The highest priority will be assigned to UDP gold class and the lowest to UDP bronze class. However, there is a problem when the UDP class equations are directly applied to TCP classes. TCP has no target bandwidth and therefore a high TCP class flow will consume much more bandwidth than a low TCP class flow. Therefore, careful considerations are needed to design TCP gold and silver class flows.

We do not allow the TCP gold class to share equation 5 with the UDP gold class because TCP flows tend to consume more bandwidth if it is available. When the TCP flows are prioritized, unprioritized neighboring flows suffer from a scarcity of bandwidth. Equation 3 is more suitable for the TCP gold class than equation 5. By means of simulation, we assign the parameter value B for the TCP gold class to three, which is even smaller than the value of the UDP silver class. Both equations 3 and 4 are used to define the TCP silver class flows. The value B for TCP silver class is set to 1.5 and the value C selected in this paper is 0.7. The TCP silver class is prioritized by both backoff parameters, B and C, at the same

Table 1 Descriptions of 3 types of flows class definition for UDP and TCP

Flow Class	Backoff Definition	Values in Simulation
UDP Gold	wtime = (random() mod A) * SlotTime	A = 8
UDP Silver	wtime = {A + (random() mod BO)/B} * SlotTime	A = 8 B = 5
UDP Bronze	wtime = {A + (random() mod BO)} * SlotTime	A = 8
TCP Gold	wtime = {A + (random() mod BO)/B} * SlotTime	A = 8 B = 3
TCP Silver	wtime = {A + (random() mod BO)/B} * SlotTime BO = minimum (BO * C + 1, BOmax	A = 8 B = 1.5 C = 0.7
TCP Bronze	wtime = {A + (random() MOD BO)} * SlotTime	A = 8

time. When a host transmitting TCP silver class packets does not receive a CTS packet from its neighbors, the new backoff counter (BO) of the flow is multiplied by 0.7 and the value BO becomes smaller than before. TCP bronze class flows share the same backoff algorithm definition with the UDP bronze class flows, which is the default BEB equation. All the flow class descriptions are summarized in Table 1. All class definitions except the TCP silver class use the default values in equation 4. The first column shows the six types of transport protocol classes. The second column lists the backoff equations used by each class definition. The third column shows the selection of the backoff parameter values used by the equations in the second column. All values omitted in the third column are assumed to be their defaults in the traditional BEB.

4. Simulation Environment

The *ns-2*[12] simulator is used to evaluate our approach to service differentiation in *ad hoc* networks. When a TCP or UDP flow is established, its service class is specified as well. The class information of the flow is marked in the TOS field[15] of every IP packet header. The Medium Access Control (MAC) entity examines the header information and backs off according to the packet class, if needed. The MAC entity selects a random waiting time for transmission in one of three cases. First, the entity is ready to transmit a RTS packet while the MAC channel is busy. Second, it is about to send a data packet while the channel is in use. Third, a host receives a MAC-level ACK packet after successful data transmission. Every host chooses a waiting time using the modified backoff algorithm according to the packet's class. RTS packets receive the same treatment as their associated data packets. When a host receives a MAC-level ACK packet from its neighboring host, its MAC entity looks up the header of the queue. If there is a packet waiting, it chooses a random waiting time depending on the packet's class. When there are no waiting packets in the queue, the bronze class backoff algorithm is performed.

To study the performance of the service classification approach, our simulation runs for 650 seconds, and the first 50 seconds of warming up data are discarded. The values of the four variables are defined in Table 2. Any variable omitted in the table has its default value. Each flow comes from one of the two traffic sources, CBR with UDP and FTP with TCP. All packet sizes are 1000 Bytes, and the throughput is calculated at the receiver in Kbps. Dynamic source routing (DSR) is used for the *ad hoc* routing algorithm. The size of a queue in MAC layer is 100, which means the entity is able to hold at most 100 packets including data packets, routing packets, and all other control packets in *ad hoc* networks. When a routing packet comes in a MAC layer queue, the packet is placed at the head of the queue and transmits ahead of data packets. Most simulations are performed in an

Table 2 Simulation Environment

Simulation Time	600 sec
Pause Time	300 sec
Movement Speed	2 meter/sec
Simulation Area	800m by 400m
Number of Hosts	15
Packet Size	1000 Byte
Ad hoc Routing	DSR
Queue Length in MAC	100

800 meter by 400 meter field in which 15 hosts are scattered at random positions. All hosts remain still for the first 300 seconds and then move constantly for the remaining 300 seconds at a speed of 2 meter per second. The destination of each and every host is decided randomly. When a host reaches at the random destination, the host chooses the next random destination and repeats until the end of simulation. The simulation environment is summarized in Table 2.

5. Simulation Results

Simulation results are generated for three different network topologies: single-hop, double-hop, and multi-hop routing networks. In a single-hop network, any host in the network is able to communicate to all other hosts directly. There is no hidden terminal and/or exposed terminal problem[11] in this network. Double-hop wireless network requires at least one intermediate host which helps forwarding packets from a sender to its receiver. The hosts in the above two networks do not move during the simulation. However, multi-hop networks contain 15 hosts all of which remain still for first 300 seconds and then move around for remaining 300 seconds in the network. All results are the average of 10 simulation runs.

5.1 Single-Hop Network Simulation

Figure 5 shows a simple result of service differentiation with three UDP flows. The hosts in the left column are senders and the hosts in the right are receivers. All hosts transmit UDP packets at a constant rate of 700 Kbps. All hosts remain still during all simulation time and the distance of each and every neighboring host is 20 meters. The ns-2 simulator supports a wireless transmission range of 250 meters. It implies that each and every host is able to contact all other hosts directly in the network. The simulator follows the 802.11 specification[13] and provides a maximum bandwidth of 2 Mbps. However, due to several overhead and wave effects, the affordable bandwidth reaches up to 1.7 Mbps for pure UDP flows and up to 1.3 Mbps for pure TCP flows. When the total bandwidth required by all participating hosts in an *ad hoc* network is smaller than the affordable

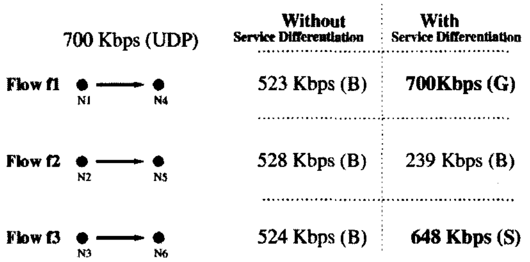


Figure 5 Service Different with 3 UDP Flows (20-meters distance)

bandwidth in the network, then service differentiation is not attractive. However, if the total requested bandwidth is much larger in the *ad hoc* network, service differentiation provides as the re-distribution of total affordable wireless bandwidth to participating flows based on the class of the flows.

The result of the bandwidth allocation without service differentiation is listed in the first column in Figure 5. Because all the packets transmitted have statistically equal chances to use the wireless medium, all flows receive similar bandwidth. Suppose the flow f1 becomes UDP gold class flow and the flow f3 changes its class to the UDP silver class. The flow f2 remains as a UDP bronze class. The second column in Figure 5 shows the result of the service differentiation. The gold class flow receives all the requested bandwidth and the silver class flow utilizes reasonably more bandwidth compared to the bronze class flow. The characters on the left side of every number is a class notation of the relevant transport flow. The bronze class has no notation due to simplicity of the table. The first column of "with service differentiation" is the same as the second column in Figure 5. Table 2 indicates that all bandwidth allocation follows very similar patterns based on the class of flows. The result of Table 3 comes from the same topology of Figure 5 except that the three flows generated by six hosts are from FTP with TCP rather than CBR with UDP. All flows share similar amounts of wireless bandwidth in case of no service differentiation. The six columns under "with service differentiation" are the result of class allocations to the three TCP flows. The gold class flow receives

Table 3 Bandwidth Allocation of 3 UDP flows with all combination of classes in Kbps

Flow ID	Without	With Service Differentiation					
Flow *f1*	523	G : 700	G : 700	s : 651	s : 647	239	239
Flow *f2*	528	329	s : 649	G : 700	241	G : 700	s : 649
Flow *f3*	524	s : 648	239	236	G : 700	s : 649	G : 700

the largest bandwidth and the bronze class flow use the smallest wireless resources. Table 3 of TCP flows follows similar behaviors as in Table 2 of UDP flows. If the TCP gold class flow employs the backoff equation of the UDP silver class in Table 1, the newly allocated bandwidth for TCP gold class flow will be around 620 Kbps. The remaining two flows receive around 413 Kbps for TCP silver class flow and 230 Kbps for TCP bronze class flow. In case the TCP gold class uses the backoff equation of the UDP gold class, the allocated bandwidth for the TCP gold class flow reaches up to more than 1 Mbps, TCP silver class to 164 Kbps, and the TCP bronze class to 58 Kbps during the whole simulation time in the *ad hoc* network. It is clear from those simulation results that the service class assignments described in section 3 provide good service differentiations for the single hop network.

5.2 Double-Hop Network Simulation

Figure 6 displays a double-hop routing topology. The hosts in the left column are senders, and the hosts in the right are receivers. The hosts in the middle are responsible for forwarding packets. The intermediate hosts are required because each and every host has a distance of 200 meters from every other host. When a sender transmits a packet, it follows the shortest path to the destination, resulting in two-hop routing. When the number of hops increases, the network-wide throughput decreases because the sending host and the forwarding host contend with each other to acquire the wireless bandwidth. All UDP senders transmit packets at a constant rate of 400 Kbps. The result in the case without service differentiation shows that the throughput of flow f2 is the lowest because of the high competition at the forwarding host N5. The reason is as follows. There are two pairs of flows to compete each other: (f1, f2) and (f2, f3). Flows f1 and f3 do not contend each other due to the out of transmission range. When the flow f1 is active, the flow f3 is also active as well. The flow f2 may wait for the next contention period. In contrast, when the flow f2 becomes active, both f1 and f3 does not transmit any packet. The probability becomes smaller for the flow f2 to win the competition because the flow f2 has to win both flows f1 and f3 at the same time during the contention period. To see the distinct effects of the service differentiation, different classes are assigned to the three flows: UDP gold for flow f2, UDP silver for flow f1, and bronze for flow f3. The result of the two-hop bandwidth allocations with service differentiation is displayed on the right in Figure 6. The UDP gold class flow reaches its target bandwidth even though f2 experiences the highest contention. The UDP silver class flow f1 outperforms the bronze class and receives reasonably high bandwidth. Table 5 shows the complete cases of the six combination of three UDP classes. The third column under "with service differentiation" contains the same result in Figure 6. The three flows share the bandwidth reasonably according to their assigned UDP classes.

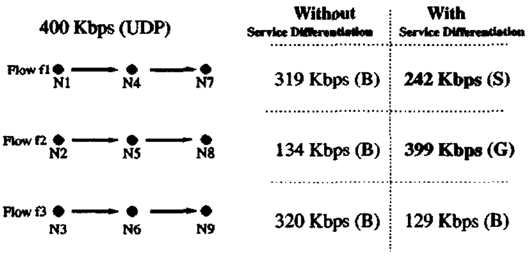


Figure 6 Service Different with Three UDP Flows (200 meter distance)

Table 6 is a result using the same topology of Figure 6 except that the three flows are from FTP applications with TCP rather than CBR with UDP. There are two different results in Table 5 compared with the results in Table 4. First, the

Table 4 Bandwidth Allocation of 3 TCP flows with all combination of classes in Kbps

Flow ID	Without	With Service Differentiation					
Flow *f1*	423	G : 545	G : 545	s : 456	s : 456	259	257
Flow *f2*	426	263	s : 462	G : 549	266	G : 551	s : 462
Flow *f3*	419	s : 458	258	260	G : 544	s : 455	G : 546

Table 5 Three UDP Flows Different Class Allocations in Kbps

Flow ID	Without	With Service Differentiation					
Flow *f1*	319	**G : 384**	**G : 400**	**s : 242**	**s : 400**	163	163
Flow *f2*	134	21	**s : 400**	**G : 399**	336	**G : 343**	**s : 215**
Flow *f3*	320	**s : 360**	359	129	**G : 394**	**s : 207**	**G : 287**

Table 6 Three TCP Flows with Different Class Allocations in Kbps

Flow ID	Without	With Service Differentiation					
Flow *f1*	117	**G : 372**	**G : 442**	**s : 185**	**s : 150**	0.21	0.83
Flow *f2*	312	127	**s : 185**	**G : 441**	137	**G : 403**	**s : 178**
Flow *f3*	188	**s : 122**	0.08	0.33	**G : 333**	**s : 207**	**G : 418**

bandwidth allocation patterns are different between pure-UDP flows and pure-TCP flows without service differentiation. The flow f2 receives the most bandwidth in the 3-TCP-flow network. According to the detailed simulation results of 3 TCP flows, the wireless resource allocation swings back and forth between the two sets of flows, (f1, f2) and (f2, f3). When the wireless medium is used by, suppose, a (f1, f2) pair of flows, they exchange RTS and CTS packets to reserve the wireless medium and tend to use the medium for a while and the flow f3 experiences a scarcity of bandwidth. At some time after f3 acquiring reasonable amount of bandwidth, the wireless resource allocation shifts to the second pair of (f2, f3). Figure 7 illustrates the packets received for three TCP flows without service differentiation over time. In case that the line segment is horizontal, the flow does not receive wireless bandwidth at the time. The line of flow f2 increases over time and receives the most bandwidth.

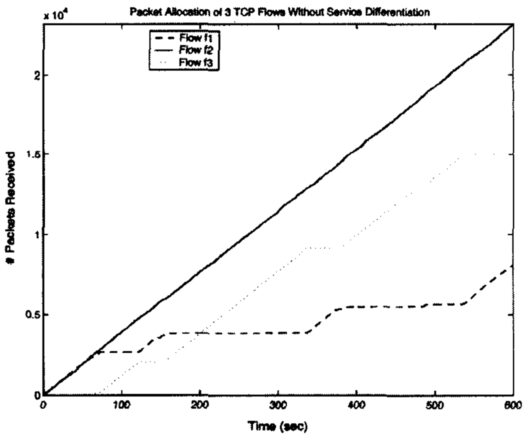


Figure 7 Number of packets received with 3 TCP flows W/O service differentiation

A second difference between Table 4 and Table 5 is the bandwidth allocation patterns of two columns, one and four, with service differentiation. When a TCP gold class and a TCP silver class becomes a pair of active flows, e.g., (f1, f2) or (f2, f3), the four allocation examples show reasonable behavior. In other words, the largest amount of bandwidth is assigned to the gold class flow and the smallest to the bronze class flow. In contrast, the first and the fourth columns in Table 5 show that the silver TCP flow does not receive enough bandwidth compared to the amount received by the TCP bronze class flow. It is because there are bandwidth allocation swings between (gold f1, bronze f2) flow pair and (bronze f2, silver f3) pair in the first column of Table 5. The pair containing the TCP gold class flow receives more bandwidth than the pair with TCP silver class. The fourth column results from the similar reason.

5.3 Multi-Hop Network Simulation

Figures 8 and 9 illustrate the number of packets

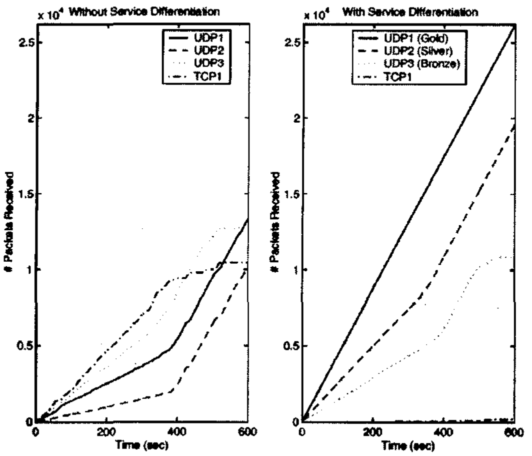


Figure 8 Number of Packets Received with 3 UDP and 1 TCP Flows

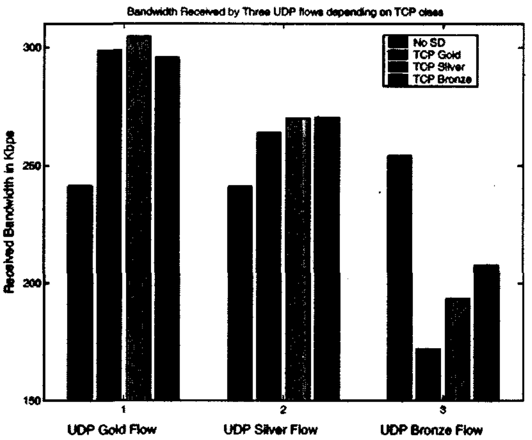


Figure 9 Bandwidth Allocation of 3 UDP flows by changing class of 1 TCP flow

received during the simulation time in an *ad hoc* network with three UDP flows and one TCP flow. There are 15 hosts randomly spread in an 800 meter by 400 meter field. Six hosts communicate with UDP packets and two hosts generate one TCP flow. Three UDP senders transmit packets at a constant rate of 350 Kbps. Some hosts perform sending and forwarding packets, while others perform receiving and forwarding. However, no hosts are assigned as a sender and a receiver at the same time. All hosts remain still for the first 300 seconds and move constantly towards random destinations for next 300 seconds.

The left figure in Figure 8 displays one example of the accumulation of packets received at each receiver over time without service differentiation. Different random locations and random movement patterns of participating hosts may produce different results of the accumulation of packets. This paper averages the results of ten simulation runs and the outcome in Figure 8 is from one of the ten runs. The slopes of two lines, UDP1 and UDP2, indicate slow increase in bandwidth until 400 seconds and they receive comparatively more bandwidth after that. Suppose the UDP gold class is assigned to the flow labeled UDP1, UDP silver class to UDP2, bronze class to UDP3, and TCP silver class to TCP1. The result of the bandwidth allocation of four flows after the class assignment is shown on the right in Figure 8. The gold class flow receives almost up to its target bandwidth(350 Kbps) and the silver class flow receives more bandwidth than before (from 170 Kbps to 261 Kbps). Moreover, the line for UDP gold class flow is almost straight, which means the packets are received regularly. This type of flow is desirable when receiving real time or multimedia data. In contrast, the TCP flow receives a very small amount of bandwidth at this time. The TCP source requires timely acknowledgements from its receiver to increase the transmission window size. Sometimes, it is possible that some prioritized traffic flows may deprive TCP sources of the chances to use wireless bandwidth. When a TCP flow frequently loses a chance to acquire the wireless medium within appropriate timeout period, the TCP sender shrinks its congestion window size drastically, resulting in low performance. In contrast, if a TCP flow becomes highly prioritized by using the parameters of the UDP gold class, then the TCP flow will dominate the wireless network. On the contrary, a prioritized UDP flow may not dominate the network because the UDP flow produces packets at a constant rate and the bandwidth consumption will not exceed a predefined limit.

Figure 9 summarizes the results of averaging ten simulation runs by changing the TCP flow class. Three UDP flows (350 Kbps) and one TCP flow compete for the wireless medium. Figure 9 shows only the average bandwidth of three UDP flows. The leftmost bars in each UDP flow are the average of ten simulation runs without service differentiation. The remaining three bars represent the average bandwidth allocated to the UDP flows with class assignment for four participating flows. The leftmost bars show that UDP1 receives the least bandwidth and UDP3 has the most. Based on this result, UDP1 is assigned to the UDP gold class, UDP2 becomes a UDP silver class, and UDP3 remains bronze class. The single TCP flow class changes its class from TCP gold to TCP silver to bronze class. The right-side three bars display the average bandwidth of the three types of UDP flows depending on the changing TCP flow class. The result shown in Figure 8 is one outcome from ten simulation runs illustrated in Figure 9. The UDP gold class flow gains near the target bandwidth in most cases regardless of the TCP class change. UDP silver class flow receives comparatively high bandwidth and insensitive to the TCP class assignment. However, the allocated bandwidth for UDP bronze class flow shrinks down and the change of TCP flow affects the assignment of the UDP bronze class flow. The allocations of bandwidth for TCP flow are as follows: 136 Kbps for no service differentiation, 102 Kbps for TCP gold, 93 Kbps for TCP silver, and 94 Kbps for TCP bronze class. Even though the TCP flow is assigned a higher priority, the TCP flow itself is sensitive to packet lost, and shrinks its congestion window size. As the UDP flows dominate the bandwidth, the TCP flow receives the remaining available bandwidth regardless of its flow class. This behavior is desirable because the delay-sensitive real time flows receive their desirable bandwidth, and the available bandwidth can be allocated to the delay-tolerant TCP packets.

Figure 10 and Figure 11 illustrates the result of bandwidth allocation from three TCP flows and one UDP flow. This simulation includes the same conditions as performed in the previous example except that this includes 3 TCP and 1 UDP flows rather than 3 UDP and 1 TCP flows. The left figure in Figure 10 shows the bandwidth allocation

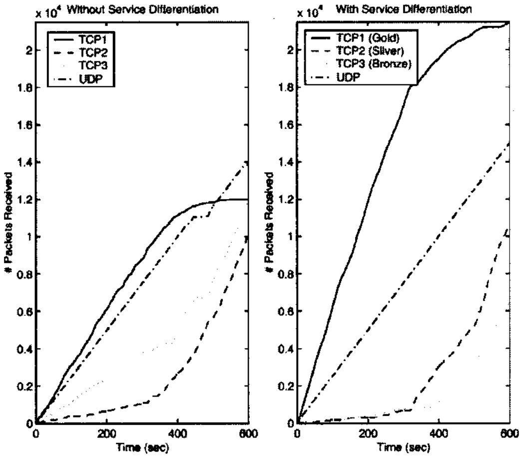


Figure 10 Number of Packets Received with 3 TCP and 1 UDP Flows

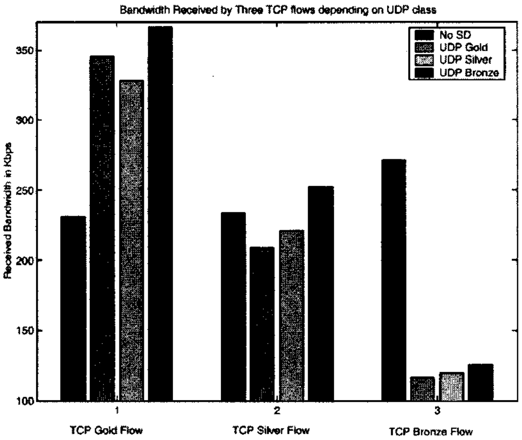


Figure 11 Bandwidth Allocation of 3 TCP flows by changing class of 1 UDP flow

trace of three TCP flows and one UDP flow at 200 Kbps without service differentiation. Suppose the UDP flow becomes a UDP silver class flow and the three TCP flows are assigned exclusively to one of the three TCP service classes. The right figure in Figure 10 displays the traces of receiving packets when the service classes are assigned to the four flows. The TCP gold class flow receives more bandwidth than without service differentiation (from 160 Kbps to 286 Kbps). The silver UDP class flow has the highest priority and receives its required target bandwidth of 200 Kbps. The line of the UDP flow is almost straight, which implies low packet delay and is attractive to delay-sensitive applications. The silver and bronze TCP flows receives 141 Kbps and 86 Kbps, respectively Figure 11 shows the average bandwidth allocation of ten simulation runs. The first bars of each TCP flow come from the simulation with traditional BEB. The flow with the smallest average bandwidth becomes a TCP gold class flow, the flow with the largest is assigned to a TCP bronze class flow, and the remaining flow is set to a TCP silver class. The fourth flow in this simulation is a UDP flow at a constant rate of 200 Kbps. Three different UDP classes are assigned to this UDP flow at a time. The second bars from the left of each TCP flow show the bandwidth allocation of three different classes of TCP flows when the single UDP flow becomes UDP gold class. Because the UDP gold flow has a target bandwidth, it does not exceed its usage of the wireless resources. The remaining available bandwidth is consumed mostly by the TCP gold class flow. The two right-side bars of each TCP flow are the average bandwidth allocation results in case of UDP silver class and of UDP bronze class flow. All three TCP flows are affected by the UDP class assignment, but the differences between the height of neighboring bars in each TCP flow are marginal due to UDP's predefined bandwidth usage limit, which is different from the greedy characteristics of TCP flow that has no upper usage limit. The average bandwidth assigned to the UDP flow is as follows: 152 Kbps for without service differentiation, 197 Kbps for UDP gold class flow, 183 Kbps for UDP silver class flow, and 132 Kbps for UDP bronze class flow. Regardless of some prioritized TCP flows that dominate the wireless bandwidth, the UDP flow with gold or silver class receives its acceptable bandwidth, which is desirable for the delay-sensitive flows.

Figure 12 depicts the average bandwidth allocation of six flows, three for UDP flows and three for TCP flows, with and without service differentiation. Among the 15 hosts in an 800 meter by 400 meter field, six hosts send and forward packets, and another six hosts receive and forward packets. The remaining three hosts participate to

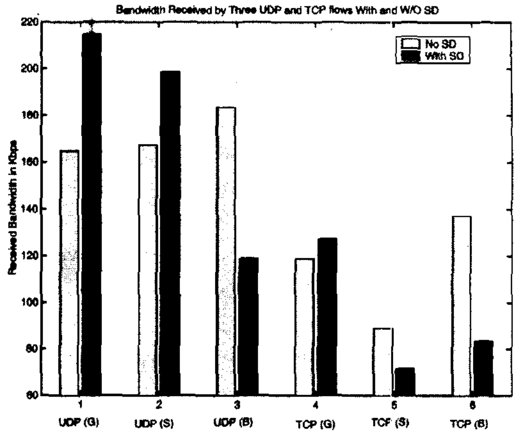


Figure 12 Bandwidth Change of 3 UDP and 3 TCP flows With and W/O Differentiation

forward packets, if available. The pause time of the simulation is the same as previous simulations of 300 seconds. Three UDP senders produce packets at a constant rate of 250 Kbps. The first three flows are for UDP flows and the remaining three are for TCP flows in Figure 12. The left bars in each flow represent the average allocated bandwidth of ten simulation outcomes without service differentiation. The right bars are the average bandwidth allocation with service differentiation. Each of the flow class is labeled on the right of the transport protocol at the bottom of each bar. 'G' stands for gold class, 'S' for silver class, and 'B' for bronze class. Prioritized UDP gold and silver class flow as well as TCP gold class flow receives more bandwidth than without service differentiation. However, the TCP silver class flow sends smaller number of packets than without service differentiation. It is because the two UDP flows consume a large amount of bandwidth and the TCP silver class flow loses chances to win the competition to acquire the wireless medium. In Figure 12, both bronze classes and the TCP silver class lose their bandwidth to high prioritized flows. However, the difference between the cases with and without service differentiation is that the TCP silver class flow is smaller than that of both bronze classes.

6. Conclusion and Future Work

This paper has proposed that modification to the backoff algorithm in *ad hoc* networks enables the specification of differentiated classes of service. We present an example of classifying UDP and TCP flows and how to modify the traditional binary exponential backoff algorithm depending on the classification of flows. Several simulation results show that high priority UDP flows receive packets regularly and up to their requested amount of bandwidth. In addition, TCP flows may be classified into several classes and receive bandwidth based on the class of flows. A mixture of the two transport protocols with service differentiation maintains the characteristics of the service differentiation. However, due to the lack of target bandwidth, TCP flows are very sensitive to the surrounding environment and careful considerations are needed to define TCP flow classes. Our simulation results clearly show that the modification of the backoff algorithm works very well for provisioning the service differentiation of packets depending on their classes in single-hop, double-hop, and multi-hop wireless *ad hoc* networks.

This paper did not provide a mechanism that decides which flow is set to given service class. If there are too many high priority classes in an *ad hoc* network, then the overall performance may be degraded due to high competition and frequent collisions. Therefore, some notion of a "bandwidth broker"[16] is required to distribute the wireless resources reasonably according to the characteristics of flows and the availability of the network resources. Moreover, the mechanism should contain guaranteed minimum service for low class flows and avoid bandwidth starvation.

References

[1] S. Das, R. Jayaram, Naveen K. Kakani and S. K. Sen, "A Resource Reservation Mechanisms for Mobile Nodes in the Internet," Proceedings of IEEE Vehicular Technology (VTC Spr '99), Houston, Texas, May 1999.

[2] R. Jayaram, Naveen K. Kakani, S. K. Das and S. K. Sen, "A Call Admission and Control for Quality-of-Service (QoS) Provisioning in Next Generation Wireless Networks," Proceedings of the 5th International Workshop on Mobile Multimedia Communication (MoMuc'98), Berlin, Germany, pp. 121-129, Oct 1998.

[3] Suresh Singh, "Quality of Service Guarantees in Mobile Computing," Journal of Computer Communications, Vol. 19, 1996, pp. 359-371.

[4] T.-W. Chen, M. Gerla, M. Kazantzidis, Y. Romanenko, and I. Slain, "Experiments on QoS Adaptation for Improving End User Speech Perception over Multi-hop Wireless Networks," Proceedings of QoS Mini Conference in conjunction with IEEE ICC'99, Vancouver, Canada, June 1999.

[5] S. Das, M. Chatterjee, and N. Kahani, "QoS Provisioning in Wireless Multimedia Networks," Wireless Communications and Networking Conference (WCNC '99), New Orleans, Lousiana, October 1999.

[6] Can E. Koksal, Hisham I. Kassab, and Hari Balakrishnan, "An Analysis of Short-Term Fairness in Wireless Media Access Protocols," Proc. of ACM SIGMETRICS, June 2000.

[7] K. Tang and M. Gerla, "Fair Sharing of MAC under TCP in Wireless ad hoc Networks," Proc. of IEEE MMT '99, Venice, Italy, October 1999.

[8] T. Nandagopal, T. Kim, X. Gao and V. Bharghavan, "Achieving MAC Layer Fairness in Wireless Packet Networks," ACM Mobicom 2000, Boston, MA, August 2000.

[9] Brahim Bensaou, Yu Wang, and Chi Chung Ko, "Fair Medium Access in 802.11 based Wireless Ad-Hoc Networks," The Workshop on Mobile ad hoc Networking and Computing (MobiHOC), Boston, MA, August 2000.

[10] T. Nandagopal, T. Kim, P. Sinha, V. Bharghavan, "Service Differentiation Through End-to-end Rate Control in Low Bandwidth Wireless Packet," IEEE International Workshop on Mobile Multimedia Communications '99, San Diego, CA, November 1999.

[11] D. Deng and R. Chang, "A Priority Scheme for IEEE 802.11 DCF AccessMethod," IEICE Transaction Communication, pp. 96-102, Jan 1999.

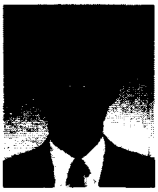
[12] I. Aad and C. Castelluccia, "Differentiation Mechanisms for IEEE 802.11," Proc. of InfoCom, April 2001.

[13] J. Sheu, C. Liu, S. Wu, and Y. Tseng, "A Priority MAC Protocol to support Real-time Multimedia Traffic in ad hoc Networks," European Wireless, Feb. 2002.

[14] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Services," Internet RFC 2475, IETF Network Working Group, December 1998.

[15] K. Nichols, S. Blake, F. Baker, and D. Black, "Definition of the Differentiated Services Field (DS field) in the Ipv4 and Ipv6 Header," Internet RFC 2474, IETF Network Working Group, December 1998.

[16] IEEE Computer Society LAN MAN Standard Committee, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Standard 802.11-1997, The Institute of Electrical and Electronics Engineers, New York, New York, 1997.
[17] The network simulator, ns-2, from http://www.isi.edu/nsnam/ns/
[18] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LANs," ACM SIGCOMM '94, London, UK, September, 1994.



강 승 석

1992년 고려대학교 전산과학과 이학사 1992년~1996년 (주)유공. 1996년 Michigan State University, CSE 공학석사 1998년~현재 Michigan State University, CSE 박사과정. 관심분야는 네트워크, 운영체제

김 진

1984년 2월 고려대학교 물리학과 이학사 1990년 8월 Michigan State University, 공학석사. 1996년 12월 Michigan State University, 공학박사. 1997년 3월~2000년 8월 건국대학교 자연과학대학 컴퓨터과학전공 조교수. 2000년 9월~현재 한림대학교 정보통신공학부 부교수. 관심분야는 생물정보학, 알고리즘, 데이타베이스, 인공지능, 광통신망