Provisioning QoS for WiFi-enabled Portable Devices in Home Networks

Eun-Chan Park[1], Nojun Kwak[2], Suk Kyu Lee[3], Jong-Kook Kim[3] and Hwangnam Kim[3]
[1]Department of Information & Communications Engineering, Dongguk University-Seoul
Seoul, Korea
[e-mail: ecpark@dongguk.edu]
[2]Department of Electrical Engineering, Ajou University
Suwon, Korea
[e-mail:nojunk@ajou.ac.kr]
[3]School of Electrical Engineering, Korea University
Seoul, Korea
[e-mail:{sklee25, jongkook, hnkim}@korea.ac.kr]
*Correspondingauthor: Hwangnam Kim

Abstract

Wi-Fi-enabled portable devices have recently been introduced into the consumer electronics market. These devices download or upload content, from or to a host machine, such as a personal computer, a laptop, a home gateway, or a media server. This paper investigates the fairness among multiple Wi-Fi-enabled portable devices in a home network when they are simultaneously communicated with the host machine. First, we present that, a simple IEEE 802.11-based home network suffers from unfairness, and the fairness is exaggerated by the wireless link errors. This unfairness is due to the asymmetric response of the TCP to data-packet loss and to acknowledgment-packet loss, and the wireless link errors that occur in the proximity of any node; the errors affect other wireless devices through the interaction at the interface queue of the home gateway. We propose a QoS-provisioning framework in order to achieve per-device fairness and service differentiation. For this purpose, we introduce the medium access price, which denotes an aggregate value of network-wide traffic load, per-device link usage, and per-device link error rate. We implemented the proposed framework in the ns-2 simulator, and carried out a simulation study to evaluate its performance with respect to fairness, service differentiation, loss and delay. The simulation results indicate that the proposed method enforces the per-device fairness, regardless of the number of devices present and regardless of the level of wireless link errors; furthermore it achieves high link utilization with only a small amount of frame losses.

1. Introduction

Consumer electronics (CEs), such as Personal Digital Assistants (PDAs), Portable Media Players (PMPs), and MP3player have all evolved to do much more than just present pictures and play audio/video. Enhanced service feature scan be continuously added onto the devices, e.g., real-time audio/video streaming, built-in radio/TV tuners, games, digital cameras and social networking; moreover, these features become more enhanced through the Wi-Fi interfaces nowadays [1]. The most common usage of the Wi-Fi interfaces that are installed onto portable devices is downloading contents from a home gateway (or media server), which is simply a desktop personal computer (PC) or a laptop with the functionality of an Access Point (AP) and/or a web server. Another common usage of such the Wi-Fi interfaces is uploading the content stored in their limited built-in memory to free up more memory space for future use, share it with others, or play the content in another audio/video device. The WiFi interface provides pervasive access to the media server via the Internet, and it provides a larger bandwidth than is available through Bluetooth and cellular networks; thus, it is considered as a *de facto* standard to provide wireless networking service for portable devices in a home network.

In this paper, we investigate the QoS issue among such Wi-Fi-enabled portable devices in a home network as they simultaneously download and upload their content via a TCP-like reliable transport protocol. This QoS issue that we deal with in a home network is different from the QoS issue that we can also encounter in handling IEEE 802.11-based WLANs. Since portable are less capable (in perspective of hardware) than desktop or laptop, they cannot usually activate multiple network flows (or network applications) [2]. Additionally, according to the usage property of the portable devices, they are supposed to download the multimedia data from the home gateway, or they upload those data in their memory to the gateway when they are connected to the home network; the downloading devices are MP3 players, smart phones or PMP, while the uploading devices are digital cameras or Camcorders. In order to address the QoS issue in a home network, we first argue that there is an unfair bandwidth sharing among portable devices; the uploading devices are preferred to the downloading ones. This unfairness originates from the property of a TCP-like congestion control, and then it becomes severe due to the IEEE 802.11 MAC protocol [3][4][5][6][7][8][9]. Then we also uncover that the unfair situation becomes worse when wireless link errors come into play in the home network; in the presence of wireless interference, the unfairness among the uploading and the downloading devices becomes exaggerated.

With these observations, we propose a price-driven QoS-provisioning framework, which is located at between the TCP and the MAC protocol, with negligibly minor modification of either protocol. In the proposed framework, we introduce Medium Access Price (MAP) as a measure of (*i*) the overall network-wide load, (*ii*) per-device link usage, and (*iii*) per-device link quality. This is estimated in the MAC layer at the home gateway, and it is delivered to the TCP senders, regardless of whether the senders are wired nodes or wireless nodes. Based on this price, every TCP sender adjusts its rate in order to assure the per-device fairness or service differentiation. Moreover, the framework can be extended to cases where the devices use a UDP-like transport protocol. We implemented the proposed framework in the ns-2 simulator, and conducted an extensive set of simulations to evaluate the compliance of the proposed framework for achieving the objective. The simulation results indicate that the proposed framework achieves the per-device fairness and service differentiation, and it outperforms the

IEEE 802.11-based networks in terms of fairness, throughput, frame delay, and packet loss.
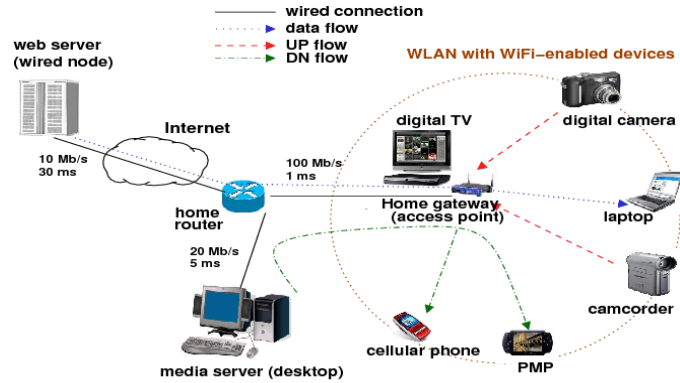
Fig. 1. Network configuration for a home network with three different traffic types.

The rest of the paper is organized as follows. Section 2 analyzes the problems of TCP-induced unfairness and the exaggerated unfairness due to the wireless link errors. The QoS-provisioning framework and algorithm are proposed in Section 3. The simulation results are presented under various network configurations in Section 4. Additionally, previous works that have aimed at achieving fairness in WLANs are addressed in Section 5. Finally, the conclusions are presented in Section 6.

2. Motivation: Unfairness in TCP over IEEE 802.11 MAC

This section first explains that, at the home gateway, the uploading devices are preferred to the downloading devices in the IEEE 802.11-based home networks. Then, it shows that the unfairness is intensified due to the wireless link errors. Note that the home gateway plays the role of (*i*) interfacing all the WiFi-enabled devices as an access point (AP) in an IEEE 802.11-based home network, (*ii*) sending and/or receiving multimedia contents, and (*iii*) relaying data among the devices.

For the purpose of investigating the unfairness discovered in IEEE 802.11-based home networks, we perform an ns-2 simulation study and examine the operational characteristics of the TCP traffic in the IEEE 802.11-based home networks. Various aspects of the wireless channel, such as the capture effect, multi-path fading, or interferences, are assumed to be negligible in order to focus on the issue of the TCP-driven unfairness and the effect of the wireless link errors. **Fig. 1** presents a typical configuration of a home network, in which the Wi-Fi-enabled devices download and/or upload content from/to a home gateway. For example, the laptop can do a web-surf via the home gateway, the PMP can download a movie from the desktop PC, and the camcorder and digital camera can upload their moving pictures and photos to the home gateway connected to a digital TV; they can upload their contents to various user community web sites (e.g., YouTube), or to they can upload on-line photo printing sites; moreover, the smart phone (or MP3 player) can download MP3 files or movies from the server.

The default values of the IEEE 802.11b MAC/PHY parameters [10] are used, and the transmission rate is set to 11 Mb/s in this study. The advertised window size of the TCP flow, denoted by W_{adv}, is set to 64 KB. The buffer size in the home gateway, B, is set to 100

packets [1].TCP/Reno is used, and the TCP packet size is set to 1,500 bytes. With this configuration, the devices uploading data to the home gateway are called the `UP DEVs`, and the devices downloading data from the gateway are called the `DN DEVs`.

2.1 TCP-Driven Fairness

As an initial step, a simple simulation is carried out when the home gateway serves only one `UP DEV` and one `DN DEV`, in order to find the operational properties of the TCP over the IEEE 802.11-operated WLAN. Specifically, the following simulation scenario is used:

- One laptop is used as a `DN DEV`, and one camcorder is used as a `UP DEV`;
- The `DN DEV` is active throughout the whole simulation time of [0s, 140s], while the `UP DEV` is active only in the period of [20s, 120s];
- There are no wireless link errors.

It is expected that both the `UPDEV` and the `DNDEV` consume almost equal amounts of bandwidth because the `UP DEV` and the home gateway contend for the shared wireless link and IEEE 802.11 the distributed coordination function (DCF) gives the fair channel access opportunity to the contending devices. However, **Fig. 2** shows that the throughput of the `UP DEV` (≈2.93 Mb/s) is much higher than that of the `DN DEV` (≈1.47 Mb/s) when they coexist in the period of [20s, 120s]. This bias toward the `UP DEV` results from the different response of the TCP to the data loss and to the ACK loss. Note that the "ACK packet" denotes the TCP ACK packet rather than the MAC-layer ACK frame, unless otherwise stated. When the `UP DEV` and the `DN DEV` share a wireless link, the home gateway transmits two kinds of packets to the wireless devices: data packets to the `DN DEV`, and ACK packets to the `UP DEV`. Since a TCP sender continuously increases its congestion window size until it detects a packet loss, the wireless link whose capacity is generally smaller than that of the wired link can be easily congested, and some packets are dropped in the buffer (interface queue) at the home gateway. The packet loss invokes congestion control, and the lost packet needs to be retransmitted by the TCP sender. However, the TCP flows of the `UP DEV` and the `DN DEV` react differently when they detect the packet losses that are incurred at the interface queue in the home gateway. As for the `DN DEV`, in response to the packet losses, the receiver transmits duplicate ACKs to inform the sender of the loss, and the sender reduces its congestion window size by half. Thus, the transmission rate of the `DN DEV` is reduced every time it encounters a data packet loss at the home gateway's buffer. On the other hand, as for the `UP DEV`, an ACK packet loss does not affect the TCP throughput as considerably as in the case of `DN DEV`. This results from the **cumulative ACK mechanism** of the TCP. Even though an ACK packet is lost due to the buffer overflow, it does not necessarily invoke the TCP congestion control mechanism. As far as subsequent ACK packets with higher sequence numbers are timely delivered to the `UP DEV` (the TCP sender), the sender recognizes that all packets up to the sequence number specified in the received ACK packets are successfully delivered to the receiver. Thus, the cumulative ACK allows the `UP DEV` to the ACK packet loss. Consequently, the throughput of the `UP DEV` becomes much higher than that of the `DN DEV`.
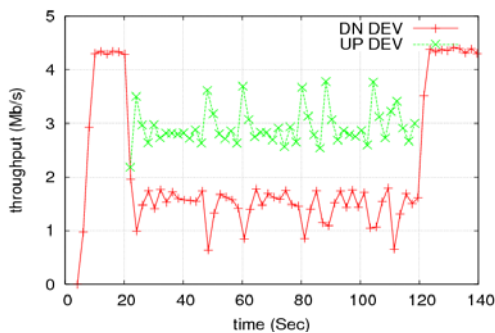
2.2 TCP-unfairness with wireless link errors

---

[1] In the simulation, the buffer size is determined for bounded queuing delay to support the multimedia streaming applications whose QoS is sensitive to delay and/or delay variations. Note that a large buffer is not desirable for these applications.

The next step is to investigate the effect of the wireless link errors on the fair sharing of the wireless link. For this purpose, the simulation scenario is configured such that the UP DEV experiences link errors in the period of [40s, 100s], but the DN DEV does not. The wireless link errors are generated by a two-state semi-Markov-modulated process as follows:

- In the OFF state, the link is error-free, while the link errors are incurred in the ON state at the average rate of 0.8;
- The periods of the OFF and ON states are 100 ms and 75 ms, respectively.
- The state transition probability matrix is given as

$$\begin{pmatrix} P_{off-off} & P_{off-on} \\ P_{on-off} & P_{on-on} \end{pmatrix} = \begin{pmatrix} 0.2 & 0.8 \\ 0.6 & 0.4 \end{pmatrix}. \tag{1}$$

From **Fig. 3**, we observe that even though link errors happen at the proximity of the UP DEV, they degrade the throughput of the DN DEV. When there is no link error in the period of [20s, 40s], the throughput of the DN DEV and the UP DEV are 1.76 Mb/s and 2.91 Mb/s, respectively. However, the link errors at the UP DEV in the period of [40s, 100s] reduce the throughput of the DN DEV to 1.14 Mb/s, as well as reducing the throughput of the UP DEV to 2.14 Mb/s. An interesting observation is that the throughput decrease is more noticeable in the DN DEV than in the UP DEV due to the link errors that occurred in the UP DEV; the throughput of the DN DEV is decreased by about 35% while that of the UP DEV is decreased by about 26%. The reason is as follows. The wireless link errors corrupt both the TCP data packets and their corresponding ACK packets coming from/into the UP DEV. When the TCP ACK packet for the UP DEV is tainted, the home gateway tries to retransmit it until the number of retransmissions reaches the maximum limit. Thus, the TCP ACK packet is not dequeued until it is transmitted successfully; moreover the interface queue continues to grow with the newly arriving packets. Finally, the newly arrived packets are dropped, regardless of TCP data packets or ACK packets. As explained in Section 2.1, packet dropping puts a more severe influence on the DN DEV than on the UP DEV. Additionally, the MAC-layer retransmission of the TCP ACK packet for the UP DEV increases the waiting time of the TCP data packet for the DN DEV because they share the queue at the home gateway. Consequently, the increased round trip time (RTT) of the TCP data packet decreases the throughput of the DN DEV.

Fig. 2. TCP-driven unfairness between the UP DEV and the DN DEV that operates in a IEEE 802.11-operated WLAN.
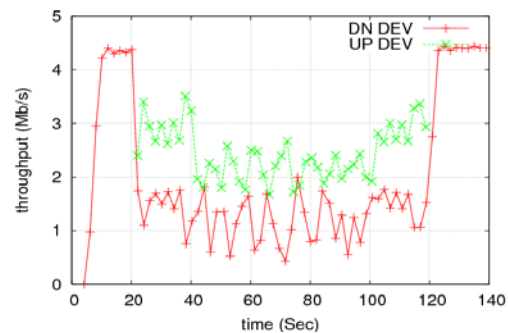
Fig. 3. TCP-driven unfairness in the presence of the wirelesslink errors in a IEEE 802.11-operated WLAN

3. Feedback Control Framework

In this section, we introduce a notion of the **medium access price** (**MAP**), which is used as a measure to control the per-device medium access probability for QoS provisioning, then we propose a cross-layer feedback framework based on the MAP, and finally, we demonstrate a possible extension of the proposed framework which can be applied to the case where a transport-layer rate control is not available (e.g., UDP).

3.1 MAP calculator

The MAP is feedback information that represents an aggregate value of the network-wide traffic load, the per-device link usage, and per-device link quality. The MAP is estimated only in the home gateway, and it is delivered via a single MAP-bit to the TCP senders. The design rationales of the MAP calculation are (*i*) to prevent the interface queue at the home gateway from overflowing, (*ii*) to assure fairness among the devices, and (*iii*) to differentiate services according to QoS specification. To achieve these goals, the MAP for the *i*th device, denoted as P_i, is computed as follows:

$$\mathsf{P}_i(t) = \left\{ \left(\mathsf{P}_i^a(t) \times \mathsf{P}_i^r(t) \right) + \mathsf{P}^n(t) \right\}^+ , \qquad (2)$$

- P_i^a : the access price for the *i*th device to determine the link usage per device;
- P_i^r : the link quality price for the *i*th device;
- P^n : the network-wide access price to represent the aggregate traffic load.

Here, $\{\cdot\}^+$ denotes the largest non-negative value less than or equal to 1. Note that the two former terms in (2) are updated at every T_{dev} while the last one is updated at every T_{net}.

3.1.1 Per-device MAP

The per-device MAP is made up of the link access price (P_i^a) and the link reliability price (P_i^r), denoting to what extent each device use the link bandwidth in the presence of wireless link errors.

Link access price (P_i^a)**:** The key idea of the per-device link access price is to give a penalty/incentive to the device that has accessed the wireless medium that is more/less than its fair link usage during a given link monitoring interval (T_{dev}). In order to determine the price, we need to establish the per-device fair link-usage for the *i*th device as

$$U_{f,i} = \phi_i \mu C T_{dev} , \qquad (3)$$

where *C* is the PHY-layer link capacity (it is assumed to be constant during the monitoring time T_{dev}), μ denotes the normalized aggregate utilization, ϕ_i denotes the weight of the ith device such that $\sum_i^N \phi_i = 1$ and *N* is the number of nodes. Note that μC is the effective capacity, and $U_{f,i}$ is the desired amount of traffic that can be sent/received by the ith device during the given time interval of T_{dev}. The aggregate utilization μ can be determined by employing any performance model of the IEEE 802.11 WLAN (e.g., Bianchi's model [11] or Calı's model [12]). The details are given in the appendix. Let us define $t_{f,i} = U_{f,i} / \mu C$ as a fair usage time for the *i*th device that can be exploited during T_{dev}. Then, from (3), ϕ_i can be written as

$$\phi_i = U_{f,i} / \left(\mu C T_{dev} \right) = t_{f,i} / T_{dev} \,, \tag{4}$$

which implies that ϕ_i can be interpreted as a fraction of time that can be used for the ith device during T_{dev}. Consequently, the per-device fair link usage assures the **time-based fairness** [13] (also referred to as the **proportional fairness** [14]). The MAP calculator keeps track of the per-device link usage U_i for the ith device during every interval of T_{dev}. Whenever the ith device is sending/receiving a data frame, it increases U_i by an amount of $\ell + t_{oh} C$, where ℓ denotes the data frame size, and t_{oh} is the overhead to transmit the data frame, such as inter-frame spacing time, back-off, and transmission time for MAC-layer ACK frame.

Let $U_i[k]$ and $\mathsf{P}_i^a[k]$ denote the link usage amount and the per-device access price for the ith device at the kth monitoring interval, respectively. Then, $\mathsf{P}_i^a[k]$ is calculated as

$$\mathsf{P}_i^a[k] = \begin{cases} \lambda \dfrac{U_i[k-1] - U_{f,i}}{U_i[k-1]} & \text{if } U_i[k-1] > 0 \\ -\lambda & \text{Otherwise} \end{cases}, \tag{5}$$

where λ ($0 < \lambda \leq 1$) is a scaling parameter. If any device has excessively occupied the link in the previous monitoring interval, i.e., $U_i[k-1] > U_{f,i}$ then the MAP increases the access price in the current interval to reduce its access opportunity. Otherwise, if $U_i[k-1] < U_{f,i}$, then the MAP decreases the price to compensate the device for its lower link access.

Link reliability price (P_i^r): This is introduced to reflect the link quality into the MAP, and it is adjusted based on the successful transmission rate. The underlying rationales are that (*i*) the home gateway compensates a device that suffers from transmission failures, and (*ii*)the wireless link's state is dynamic and time-varying.

Let $\eta_i[k]$ denote the successful transmission rate of the ith device during the kth monitoring interval. Then, the link reliability price for the ith device, $\mathsf{P}_i^r[k]$, is given as

$$\mathsf{P}_i^r[k] = \eta_i[k-1] \,. \tag{6}$$

In order to estimate the successful transmission rate, η_i, we choose an interim solution which uses `HELLO` frames to probe wireless link state, but it can be enhanced by adopting non-intrusive link state estimation scheme such as the work in [15].At every period of T_{hello}, each device broadcasts `HELLO` frames that contain the information of the sender, the sequence number (optional), and the time period (optional). The home gateway keeps track of the number of the successfully received `HELLO` frames sent from each node during every given interval of $T_{window} = M \times T_{hello}$, where M is a positive integer. If it receives m_i `HELLO` frames from node i during T_{window}, then $\eta_i = m_i / M$. If a certain device frequently experiences transmission failure, the link reliability price is reduced so as to give better opportunity to the device for compensation. However, since `HELLO` frames themselves are intrusive since they consume the link bandwidth, we need to analyze the impact of `HELLO` module on the per-device link usage. Effective per-device link-usage after the overhead of the `HELLO` frame-based link probing module is removed can be determined as follows:

$$U_{f,i} = \mu \cdot C \cdot T_{dev} - N \left(\frac{T_{dev}}{T_{window}} \cdot M \cdot t_{HelloFrame} \cdot C \right) = \left(\mu - N \frac{t_{HelloFrame}}{T_{hello}} \right) \cdot C \cdot T_{dev} \tag{7}$$

where $t_{HelloFrame}$ is the duration of `HELLO` frame and the other terms have already been defined in previous equations.From (7), we perceive that we can alleviate the overhead of sending `HELLO` frames by reducing $t_{HelloFrame}$ or increasing T_{hello}. As for $t_{HelloFrame}$, we can reduce the frame size by including only source MAC address (6 bytes) within the frame . As for T_{hello}, we can use very large value for T_{hello} if the link state is quite stable. Additionally, we can further reduce the overhead if we only use `HELLO` frames when there is no traffic for some devices; in other words, we can use real traffic to estimate link state instead of sending explicit `HELLO` frames to probe the link state.

Remark: Instead of the price in (6), an unreliability price can be used for improving the link utilization. If a node experiences transmission failure with a high probability, then it is deprived of transmission chances under the philosophy that it is not necessary to waste resources for transmitting data frames that probably involve retransmissions. This alternative approach can improve the network-wide utilization when the link state is relatively static. However, recall that the objective of this study is to propose a framework for supporting the per-device fairness and/or service differentiation that is immune to wireless link errors.

3.1.2 Network-wide MAP

The network-wide access price, P^n, is devised to avoid packet loss due to the buffer overflow at the interface queue. The design objective is to tightly regulate the queue length of the home gateway in order to keep it around the desired target value, and to match the input rate of packets with the available link capacity. This objective is similar to that of the active queue management (AQM) scheme for Internet routers [14]. **Table 1** shows the pseudo code to calculate P^n, as well as the state variables and the design parameters.

From the pseudo code in **Table 1**, the difference between the aggregate input traffic amount ($R[j]$) and its target value ($R_{ref}[j]$) at the jth update interval can be rewritten as

$$R[j] - R_{ref}[j] = \left(R[j] - \mu[j]CT_{net} \right) + \gamma \left(q[j] - q_{ref} \right) \\ + \beta \left(\sum_{k=0}^{j} \left(R[k] - \mu[k]CT_{net} \right) + \gamma \left(q[k] - q_{ref} \right) \right). \tag{8}$$
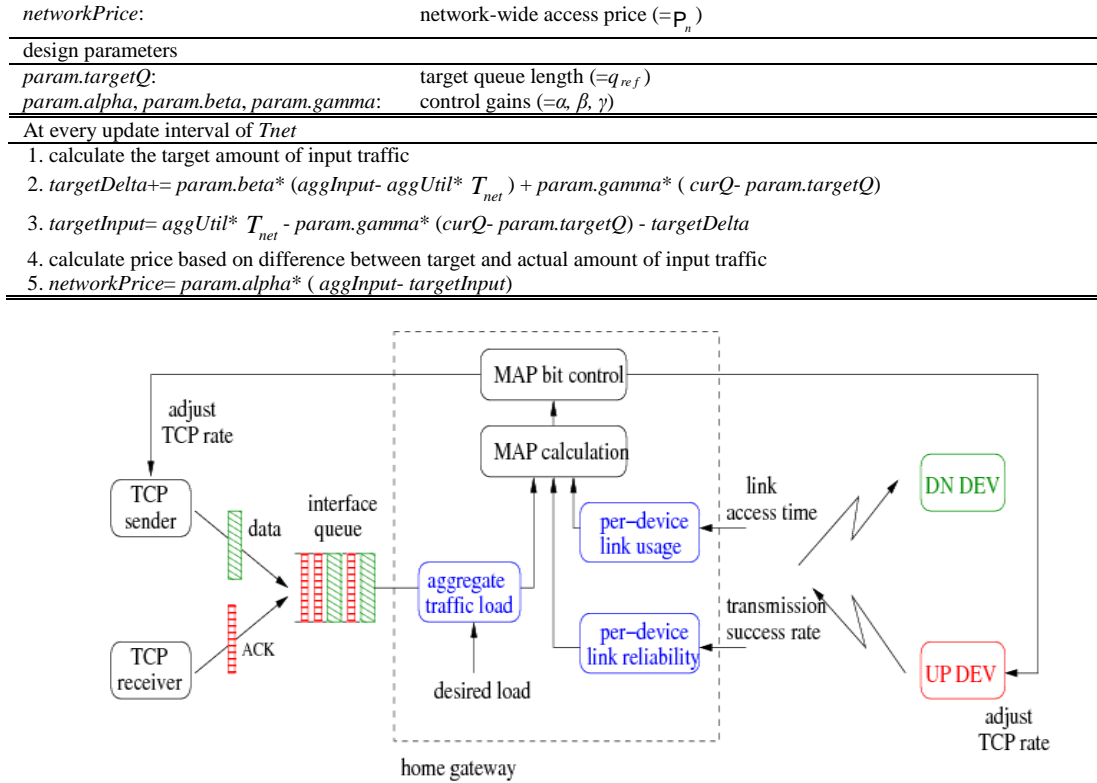
It can be seen from **Table 1** and (8) that the network-wide access price $\left(P^n = \alpha(R[j] - R_{ref}[j]) \right)$ increases (*i*)as the input amount of traffic to the interface queue exceeds the amount that can be served at the link capacity during T_{net} (i.e., $R[j] > \mu[j]CT_{net}$), and (*ii*)as the queue length exceeds its target value (i.e., $q[j] > q_{ref}$). The term ($R[k] - \mu[k]CT_{net}$) in (8) can be approximated with the following relationship:

$$\left(R[k] - \mu[k]CT_{net} \right) = q[k] - q[k-1]. \tag{9}$$

Table 1. Pseudo code to calculate the network-wide access price.

state variables	
curQ:	current queue length (updated whenever a packet enqueues or dequeues) (=q)
aggUtil:	aggregate utilization (=μC)
aggInput:	accumulated amount of input traffic to the interface queue during T_{net} (=R)
targetInput:	target amount of input traffic to the interface queue during T_{net} (=R_{ref})
targetDelta:	dummy variable for recursive calculation

networkPrice:	network-wide access price ($=\mathsf{P}_n$)
design parameters	
param.targetQ:	target queue length ($=q_{ref}$)
param.alpha, *param.beta*, *param.gamma*:	control gains ($=\alpha$, β, γ)

At every update interval of *Tnet*

1. calculate the target amount of input traffic

2. *targetDelta+= param.beta* * (*aggInput- aggUtil* * T_{net}) + *param.gamma* * (*curQ- param.targetQ*)

3. *targetInput= aggUtil* * T_{net} - *param.gamma* * (*curQ- param.targetQ*) - *targetDelta*

4. calculate price based on difference between target and actual amount of input traffic
5. *networkPrice= param.alpha* * (*aggInput- targetInput*)

Fig. 4. Feedback contorl framework of based on the medium access price.

The MAP algorithm can then be easily implemented only with the information of the queue length, without having to estimate the input rate or the link capacity as

$$\mathsf{P}^n[j] = \alpha \left[(\gamma + \beta + 1)q[j] - q[j-1] - rq_{ref} + \beta\gamma \sum_{k=0}^{j} \left(q[k] - q_{ref} \right) \right]. \quad (10)$$

Conclusively, the MAP in (1) has the following desirable properties in perspective of QoS provisioning:

- MAP calculates P_i, regardless of the device's number offlows;
- MAP is flexible to support various QoS policiesby setting thefair share of each device considering its serviceweight;
- MAP requires monitoring the queue lengthof AP and the channel usage of devices to calculatethe media access price from (5) and (10). The devices only convey the access price from the MAC tothe TCP. Thus, the MAP algorithm does not incur large communication overhead and itsimplementation cost will not be significant.

3.2 Feedback framework of medium access price

Based on the MAP, we propose a feedback control framework, where the MAP is delivered to TCP senders and then the TCP senders adjust their sending rate to reduce the congestion at the interface queue of the home gateway. **Fig. 4** shows the proposed feedback architecture.

This approach can be implemented with minimal changes on the standard protocol stack. The main functions of the proposed MAP-based feedback framework are (1) the MAP calculation module, (2) the link quality probing module, and (3) the MAP-bit control module. At every given update interval, the MAP calculation module estimates the MAP based on the

length of the interface queue, the link usage, and the link quality (quantified by the link quality probing module). Based on the value of the MAP, the MAP-bit control module sets a single MAP-bit[2] in the MAC header to deliver the MAP information to the TCP senders in a distributed way. Once a packet (TCP data or ACK) arrives to the home gateway, the MAP-bit control module picks a pseudo random number that is uniformly distributed between zero and one, and it compares this value with the MAP whose value also ranges from zero to one. If the random number is less than the MAP, then the MAP-bit is set, otherwise it is cleared. This MAP-bit is piggybacked on the data frame from the home gateway to the corresponding device. With this binary feedback, the overhead can be minimized for delivering the MAP. On receiving a packet whose MAP-bit is set, the device maps the MAP bit in the MAC header onto the explicit congestion notification (ECN) bit [17]in the IP header, so that the TCP sender adjusts its congestion window size. In this way, the TCP senders recognize both the state and the usage of the wireless link, and they can adapt their sending rates.

3.3 Extension to contention window control and active queue management

Since the fairness in the proposed feedback control framework is actually controlled by the TCP senders in the transport layer, the proposed approach cannot be directly applied when no rate control mechanism is employed in the transport layer, e.g., UDP. Even in this case, the MAP can be used to enforce fairness and service differentiation by extending the proposed framework to the MAC-layer contention window control and the network-layer active queue management scheme (AQM).

First, we consider the extension of the MAP to the contention window control, which is suitable for the uplink UDP traffic. A high value of the MAP indicates that the link is highly loaded. There the probability of link access attempts needs to be decreased by increasing the contention window size. On the other hand, if the value of the MAP is low, then the link can be accessed with a higher probability to enhance the overall utilization. The MAP calculation does not need to be changed, while the MAP-bit control requires a small change. We propose to include the MAP-bit in a MAC-layer ACK frame, as well as in the MAC-layer data frame, in order to cope with a situation where a wireless device only sends data frames to the home gateway without receiving any data frames from the home gateway. If a wireless device successively receives the MAC frames where the MAP-bit is set, then the wireless device increases its contention window size twice; otherwise if it successively receives the packets where the MAP-bit is unset, then it sets the contention window size to the minimum value that is specified by the IEEE 802.11 standard[3].This is one example of contention window control based on the MAP. We can consider another example. Let us denote $cwnd[n]$ as the size of the contention window at the nth control interval and γ_{MAP} as the fraction of the MAC frames that received with the MAP-bit among the total MAC frames received. We can control $cwnd[n]$ depending on the value of γ_{MAP} as

$$cwnd[n] = \begin{cases} \min(cwnd_{max}, cwnd[n-1] \times K_+) & \text{if } \gamma_{MAP} > \max_{th} \\ \max(cwnd_{min}, cwnd[n-1] - K_-) & \text{if } \gamma_{MAP} < \min_{th} \\ cwnd[n-1] & \text{otherwise} \end{cases}, (11)$$

where $cwnd_{max}$ and $cwnd_{min}$ are the maximum and minimum values of $cwnd$ (as specified by the IEEE 802.11), $\max_{th}$ and $\min_{th}$ ($0 < \min_{th} < \max_{th} < 1$) are two threshold values to control

---

[2]We can allocate the MAP-bit without modifying the existing MAC header by utilizing the undefined "subtype" of "frame control" field in the MAC header.

[3]Note that the auto-rate fallback (ARF) mechanism works in a similar way.

cwnd and $K_+(>1)$ and $K_-(>0)$ are the increase and decrease control factors. The control rule in (11) is similar to the philosophy of the *additive increase multiplicative decrease* adopted in a TCP congestion control scheme. While the conventional control method is reactive to transmission failure, the proposed contention window control based on the MAP information is considered as a proactive approach.

Next, we focus on the extension of the MAP to the AQM scheme, in order to enforce fairness or service differentiation with the downlink UDP traffic. Without any proper rate control mechanism implemented on the sender-side, there is no way to drop the excessive packets to achieve fairness. The MAP can be used as an efficient criterion to determine whether or not to drop a packet. Upon a packet arrival, the MAP calculation and the MAP-bit control are performed in the same way as aforementioned. If the value of the MAP is high enough to set the MAP-bit, then the home gateway only drops the packet, instead of setting the MAP-bit to the packet. This approach can improve the per-device fairness without decreasing the overall utilization, even if some devices are greedy while others are idle. Although greedy devices are penalized by a higher value of the per-device access price, they can utilize the unused share for the idle devices. If the channel is under-utilized, then the network-wide access price becomes negative, i.e., $\mathsf{P}^n = \alpha(R[j] - R_{ref}[j]) < 0$, and the actual MAP, which

is the sum of the per-device price and network-wide price, is decreased for the greedy devices to allow them to occupy the unused share of the link for the idle devices. Consequently, the shared wireless link is neither wasted nor overused while maintaining fairness at a reasonable level. Here, we address the flexibility and extensibility of the proposed framework but the implementation and the evaluation of these MAP extensions are omitted as they will form the basis of future research.

4. Performance Evaluation

We consider a WiFi-based home network consisting of a home gateway and several wireless devices, as shown in **Fig. 1**. Throughout the performance evaluation, the following three typical types of WiFi devices are considered, according to their traffic characteristics:

- `TYPE-1 DEV` interacts with the corresponding wired nodes to consider FTP or P2P download, web-surfing and the messenger, e.g., laptops or PDAs;
- `TYPE-2 DEV` downloads the media contents from the home gateway or from a desktop PC that is connected to the gateway, e.g., MP3 players or PMP;
- `TYPE-3 DEV` uploads their contents to the home gateway or to the web server in the wired networks, e.g., a digital camera or camcorder.

Note that `TYPE-2` and `TYPE-3 DEV` generate the assumed traffic of portable devices while `TYPE-1` produces ordinary wireless traffic in IEEE 802.11-based WLANs. It is assumed that all the devices are randomly placed in an area (100 m × 100 m) so that they are within the carrier sensing range of each device. Hence, there is little concern for hidden devices, and the Ready-To-Send (RTS) / Clear-To-Send (CTS) mechanism is not used. Moreover, the mobility of device is not considered in this study. The control parameters of the MAP-enabled framework, α, β, γ, and λ, are set to 0.03, 0.03, 0.05 and 0.8, respectively; furthermore the update intervals, T_{net} and T_{dev}, are set to 10 ms and 20 ms, respectively. The target value of the AP interface queue is set to 50 packets and the other simulation parameters remain unchanged from those that were presented in Section 2.

4.1 Validation of the MAP-enabled WLAN

We firstly confirm that the MAP-enabled WLAN solves the issue of the unfairness between the `UP DEV` and the `DN DEV` due to the TCP asymmetry and the wireless link errors, as presented in **Fig. 2–3**. The same simulation scenario is used for this study, as was used in Section 2.
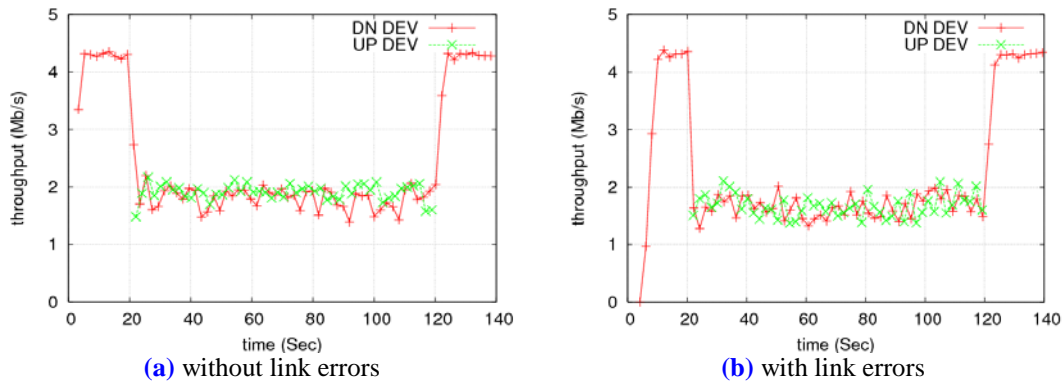

(a) without link errors   (b) with link errors

Fig. 5. Per-device throughput in a MAP-enabled WLAN.

Figs. 5-(a) and **(b)** show the TCP throughput of the `UP DEV` and the `DN DEV` with and without the wireless link errors, respectively. If there is no link error (see **Fig. 5-(a)**), then both the `UP DEV` and the `DN DEV` almost equally share the wireless link capacity when they coexist in the period of [20s, 120s]: the average throughput of the `DN DEV` and the `UP DEV` are 1.82 and 1.91 Mb/s, respectively. When it is compared to the case without the MAP mechanism (see **Fig. 2**), the throughput ratio of the `DN DEV` to the `UP DEV` is remarkably increased from 0.50 to 0.90. In addition to enhancing the fairness, the MAP framework significantly reduces the throughput fluctuation, which is meaningful for the multimedia streaming services. Next, attention is given to the case with the wireless link errors, as shown in **Fig. 5-(b)**. The average throughputs of the `DN DEV` are 1.66, 1.62, and 1.78 Mb/s in the periods of [20s, 40s], [40s, 100s], and [100s, 120s], respectively; those of the `UP DEV` are 1.79, 1.60, and 1.79 Mb/s in the corresponding periods, respectively. Recall that the error occurs in the period of [40s, 100s]. During the whole simulation time when the `DN DEV` and the `UP DEV` coexist, the throughput ratio between both devices ranges between 0.93 and 1.01. By comparing **Fig. 5-(b)** with **Fig. 3**, it can be observed that the performance of the proposed MAP framework is not significantly affected by the wireless link errors due to the introduction of the link reliability price.

4.2 Performance with respect to per-device throughput

This simulation is performed to investigate the per-device fairness of MAP-enabled WLAN, compared with that of the IEEE 802.11-based WLAN. The simulation scenarios are configured as follows: (*i*) `TYPE-1 DEV` is active throughout the whole simulation period and it continuously communicates with the corresponding node, (*ii*) `TYPE-2 DEV` starts to download the multimedia data during the interval of [20s, 120s], (*iii*) `TYPE-3 DEV` uploads the data during the interval of [40s, 140s].

Fig. 6 compares the per-device throughput of the IEEE 802.11-based devices with that of MAP-enabled devices when the wireless link errors occur to `TYPE-1 DEV` during the interval of [60s, 100s]. Note that the model of the wireless link errors is the same as that specified in Section 2. In the case of the IEEE 802.11-based devices (**Fig. 6-(a)**), there is no considerable

discrepancy between the two types of devices (`TYPE-1 DEV` and `TYPE-2 DEV`) in the period of [20s, 40s] when there is no uploading device. The slight difference between the aforementioned two types of devices is mainly due to the difference of the RTT of TCP flows. As previously seen in Section 2, the `UP DEV`, i.e., `TYPE-3 DEV` occupies most of the network bandwidth in the period of [40s, 120s]. Accordingly, the throughput of `TYPE-1 DEV` and `TYPE-2 DEV` are significantly reduced in this period. In the erroneous period of [60s, 100s], the per-device fairness drastically deteriorates, i.e., the throughput of `TYPE-1`, `TYPE-2`, and `TYPE-3 DEV`s are 0.31, 1.18, and 2.47 Mb/s, respectively. However, **Fig. 6-(b)** confirms that the MAP-enabled device successfully achieves the fairness, regardless of the number of active devices, their types, and wireless link errors. It is also shown that even though the wireless link errors disrupt the performance of `TYPE-1 DEV`, they do not severely affect the other devices and per-device fairness.

4.3 Performance with respect to fairness

In this section, the per-device fairness is quantitatively evaluated in terms of the fairness index. The fairness index is defined as follows:

$$F = \frac{\left(\sum_{i=1}^{N} \rho_i\right)^2}{\left(N \sum_{i=1}^{N} \rho_i^2\right)}, \tag{12}$$

where N is the number of devices, and ρ_i is the throughput of the ith device. The ideal value of F becomes one when all the devices have the same throughput.

Fig. 7 compares the fairness index between the IEEE 802.11-based and the MAP-enabled devices. The same simulation configuration as was previously used is adopted here, except that all the devices are active in the total simulation time of 100 seconds, and the transmission rate is set to 5.5 Mb/s in order to clearly see the bottleneck effect of the wireless link on the fairness among the devices. The metrics F is measured as the number of nodes for each type of device increases[4]. As presented in **Fig. 7**, F in the MAP-enabled network is remarkably higher than that in the IEEE 802.11 network, regardless of the number of devices.

Then, we carried out two additional studies to investigate the fairness index further, and we present the results in **Table 2** and **3**. As for the first study whose results are presented in **Table 2**, we employed the same simulation configuration as we used for **Fig. 7**, but we enable different number and type of flows for each device; in the table, `UP`, `DN`, `UP/DN` flow mean a upload, a download, an interactive flow, respectively. As shown in **Table 2**, the fairness index values in MAP-enabled WLAN is higher than those values measured in IEEE 802.11 legacy WLAN.

As for the second study, we used a different network configuration, where there are five different devices, each device has a different number of flows together with a different active period, which is as follows: (*i*)`DEV1` has one `DN` flow in [0s, 200s], (*ii*) `DEV2` has one `UP` flow in [0s, 200s], (*iii*) `DEV3` has one `DN` flow and one `UP` flow in [50s, 200s], (*iv*) `DEV4` has two `DN` flow in [100s,200s], and (*v*) `DEV5` has 2 `UP` flows in [150s, 200s]. The specific results are in **Table 3**. We made a similar observation to previous studies: MAP-enabled WLAN outperforms IEEE 802.11-based WLAN in perspective of fairness.

---

[4]For convenience, Fig. 7 presents the number of devices per device type, and so the total number of devices is three times larger than that number.
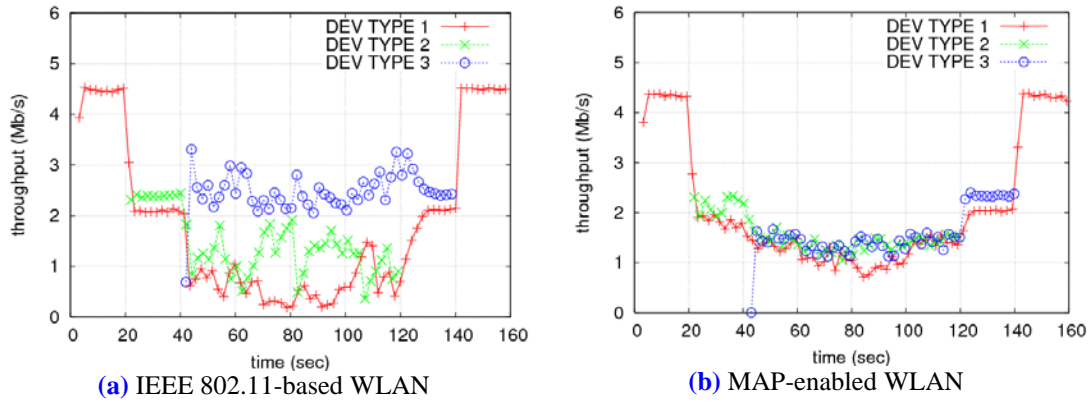
(a) IEEE 802.11-based WLAN   (b) MAP-enabled WLAN

Fig. 6. Comparison of the per-device throughput in the presence of the wireless link errors.
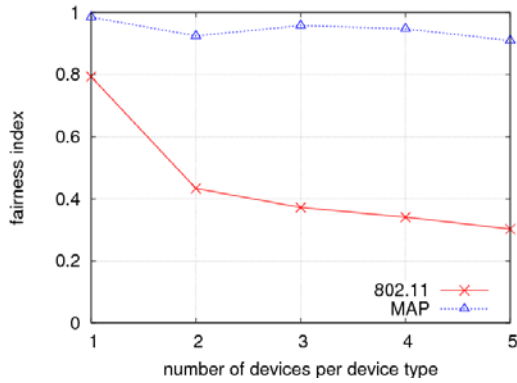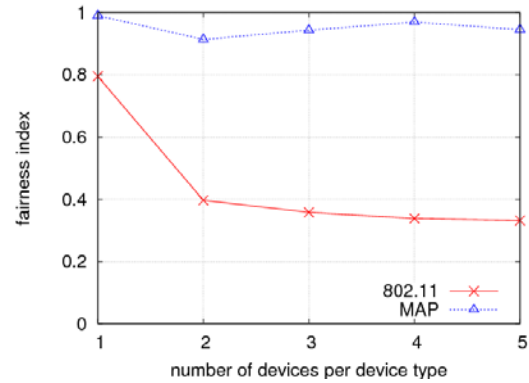

Fig. 7. Comparison of the fairness index when (*i*) DEV1 has two DN flows and one interactive flow, (*ii*) DEV2 has one DN flow, and (*iii*) DEV3 has one UP flow.

Fig. 8. Comparison of the faireness index when (*i*) DEV1 has two DN flows and one interactive flow, (*ii*) DEV2 has one DN flow, and (*iii*) DEV3 has one UP flow and one interactive flow.

Table 2. Comparison of the fairness index when each device has different numbers of flows.

Number of flows									Fairness index	
DEV1			DEV2			DEV3			IEEE 802.11	MAP
UP	DN	UP/DN	UP	DN	UP/DN	UP	DN	UP/DN		
0	2	1	0	1	0	1	0	0	0.79	0.98
0	2	1	0	1	0	1	0	1	0.79	0.99
0	2	1	0	1	0	1	1	1	0.80	0.99
0	2	1	1	1	1	1	1	1	0.52	0.96
1	1	1	0	1	0	1	0	0	0.52	0.88
1	1	1	0	1	0	1	0	1	0.49	0.91
1	1	1	0	1	0	1	1	1	0.55	0.94
1	1	1	1	1	0	1	1	1	0.44	0.89

Table 3. Performance Comparison of Fairness index when five deviceshave different numbers of flows and it is active in different intervals.

Time interval	802.11	MAP-enabled
[0s, 50s]	0.946	0.998
[50s, 100s]	0.849	0.995
[100s, 150s]	0.691	0.994
[150s, 200s]	0.583	0.994

4.4 Performance with respect to delay and loss

We now evaluate the performance of the MAP-enabled network in terms of delay and loss. The simulation configuration is the same as that used in the first experimentation of the previous section. As shown in **Fig. 9**, the loss rate of the IEEE 802.11-based WLAN increases by up to 15% as the number of devices increases; however, the loss rate of the MAP-enabled WLAN is approximately not higher than 0.25% for the entire range of N. The high loss rate of the IEEE 802.11-based WLAN causes the QoS of the multimedia service to significantly deteriorate. These results indicate that a large portion of the aggregate throughput in the IEEE 802.11-based network is wasted in retransmitting the TCP packets due to the high loss rate, and so it reduces the effective throughput, i.e., good put.

Fig. 10 shows that the average end-to-end delay of the IEEE 802.11-based WLAN is not much affected by the number of devices; however at a maximum, it is two times larger than that in the MAP-enabled network. The reason is that the buffer at the home gateway in the former network is almost filled with frames, regardless of the number of devices, while the buffer in the latter network is controlled so that it is regulated around its target value. Also, one noteworthy feature of the MAP-enabled WLAN is that the average delay can be controlled by setting the desirable queue length to a proper value, which is useful to satisfy the required QoS of real-time service, and that is not possible in the IEEE 802.11-based WLANs.

## 4.4 Performance with respect to per-device fairness

As the final simulation study, we evaluate the performance of the MAP-enabled WLAN with respect to the weighted fairness. In the MAP-enabled network, an arbitrary service weight (ϕ_i in (3) of Section 3.2) can be assigned for each device. In this simulation, the weights are set as follows: the weights of `TYPE-1 DEV (DEV1/DEV2)`, `TYPE-2 DEV(DEV3/DEV4)`, and `TYPE-3 DEV(DEV5/DEV6)` are set to 0.083, 0.25, and 0.167, respectively. The weights of `TYPE-2 DEV/TYPE-3 DEV` are three/two times higher than that of`TYPE-1 DEV`, respectively. Note that the IEEE 802.11 does not support this kind of service differentiation; therefore, these results are not presented in this simulation.

Fig. 11-(a) presents the per-device throughput when there is no link error. As shown in the figure, the per-device throughput is almost in proportion to the service weight, and the throughputs for the devices that have the same weight (e.g., `DEV1`and `DEV2`) are almost equal. On the other hand, **Fig. 11-(b)** presents the throughput differentiation among differently weighted devices, when all devices encounter wireless link errors in the period of [30s, 70s]. Even in this case, MAP-enabled WLAN can differentiate per-device throughput according to the device's weight. These simulation results imply the flexibility of the MAP-based framework, in that it can realize the various fairness policies by assigning the appropriate service weights to the devices.

5. Related Work

This section briefly discusses several previous approaches to achieve the TCP fairness in the IEEE 802.11 WLANs, *even though they do not directly target home networks*. The approaches are classified into four categories: (*i*) a MAC-layer solution that involves the differentiated channel access mechanism; (*ii*) a TCP-layer solution that adjusts the TCP window size; (*iii*) a queuing-based solution that resorts to the differentiated packet dropping/marking between the TCP data and the ACK packets; and (*iv*) a rate-control mechanism that limits the aggregate uplink traffic rate.

The method in [7]utilizes the IEEE 802.11e [19] service dfferentiation scheme to provide the TCP fairness. This dfferentiates the media access scheme for TCP data from that for ACK packets with different inter-frame spaces, minimum contention window sizes, and transmission opportunity (TXOP) bursting. Also, the approach in [20] allows the AP to transmit multiple frames in a burst, and the burst size is adjusted based on the collision probability that is monitored in the AP to maximize the wireless link utilization and provide fairness between the uplink and downlink data flows.
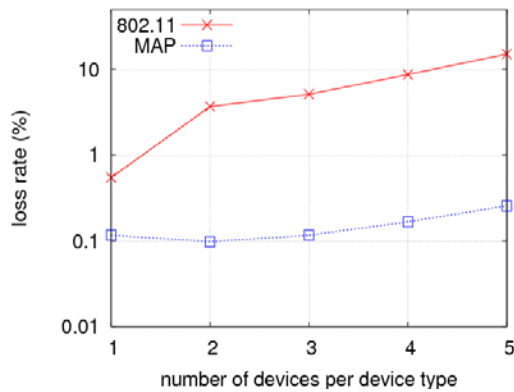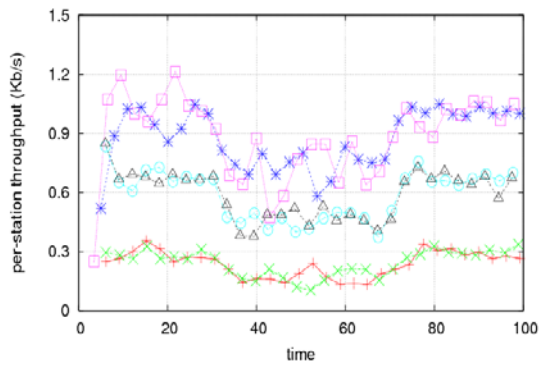

Fig. 9. Comparison of the loss rate measured in the IEEE 802.11-based and the MAP-enabled WLANs.
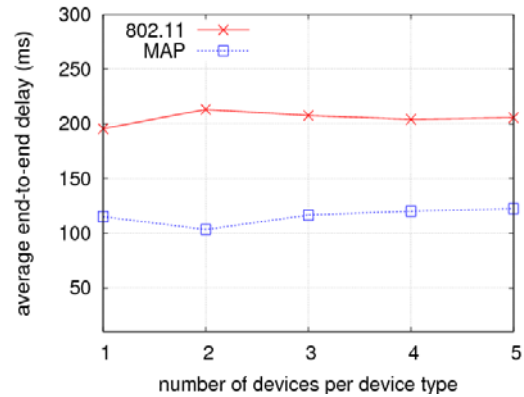
Fig. 10. Comparison of the end-to-end delay measured in the IEEE 802.11-based and the MAP-enabled WLANs.
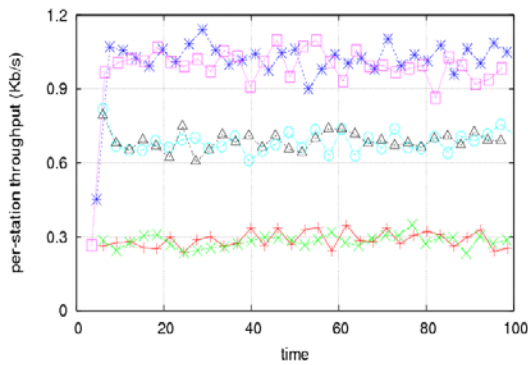
(a) IEEE 802.11-based WLAN

(b) MAP-enabled WLAN

Fig. 11. Per-device throughput in the MAP-enabled WLAN when the devices have different service weights.

A transport-layer solution [1], [21] can mitigate the TCP unfairness in the IEEE 802.11 MAC. In [1], it has been pointed out that the TCP fairness is affected by the buffer availability at the AP. As a solution, the AP is entitled to manipulate the advertised TCP window size for the TCP ACK passing through it. The approach in [21] extends the idea that was presented in [1] to improve the overall link utilization. When modifying the advertised TCP window size, the approach in [21] considers the achievable maximum TCP window size, as well as the buffer availability.

On the other hand, this fairness issue can be dealt with in the IP layer by devising a proper queue management scheme [22][6][23] or by devising a rate control scheme [24]. The approach in [22] employs a SPM-AF (selective packet marking scheme with ACK filtering) scheme and combines this with the LAS (least attained service) scheduling to give a higher service opportunity to the downlink TCP data packets and to assure the per-flow fairness. When a TCP ACK packet arrives, the AP removes the redundant ACK packets that belong to the same connection in the queue. The approach proposed in [6] performs the per-flow scheduling in the AP to achieve the fairness. It treats the data queue and the ACK queue differently, and it determines the optimal scheduling probabilities for these per-flow queues. In a similar way, the queue management scheme proposed in [23] manages the virtual per-flow queue in the AP and it makes decision on packet drop based on the per-flow queue length. The rate control algorithm in [24] mitigates the unfairness problem by implementing a token bucket-based rate limiter in the AP. The rate limiter controls the rate of the aggregate uplink traffic so as to provide fairness.

Compared to the previous studies, the proposed scheme is based on a cross-layer approach. The MAP is estimated in the AP and it is delivered to the TCP senders so that they can adjust their sending rates in a distributed way. Thus, the proposed approach does not require a per-flow/station buffer, nor does it involve per-flow/station scheduling. The MAP plays a key role in providing the per-device fairness and high utilization, and it is an effective measure that can reflect the MAC/PHY information such as the per-device link usage/quality and the network-wide load. Unlike the previous approaches, this scheme is the flexible and extensible, and it is generalized framework from the following aspects, which is its unique property:

- The proposed framework can support the weighted fairness and the proportional fairness, where the former is an essential building block for service differentiation and the latter is effective when stations have different PHY transmission rate.
- The MAP information can be used for the contention window control and queue management scheme to control the UDP traffic for fairness, so that this scheme works regardless of whether a station uses TCP or UDP.
- The MAP takes both the per-device access price and the network-wide access price into account, and so the proposed method does not decrease the aggregate utilization to provide fairness even when some stations are greedy while others are idle.

6. Conclusion

In this paper, we have investigated the issue of fairness and/or service differentiation among the Wi-Fi-enabled portable devices in home networks when they simultaneously upload and download contents in the presence of wireless link errors. We have first presented that the service in the IEEE 802.11-based home networks is biased toward the uploading devices due to the asymmetric behavior of the TCP congestion control. It has also been shown that the unfairness problem becomes worsen due to the wireless link errors. In order to resolve the unfairness, we have introduced the medium access price (MAP) and devised the price-based QoS-provisioning framework to support per-device fairness. In the proposed approach, the MAC layer at the home gateway measures the link usage and the reliability for each device, and the network-wide utilization, and then it quantifies it with the MAP. According to the MAP, the TCP senders adjust their sending rates to assure better per-device fairness while maximizing the link utilization simultaneously. The simulation results clearly indicate that the proposed MAP-based framework outperforms the existing TCP over IEEE 802.11 MAC, with respect to fairness, delay, and loss, regardless of wireless link errors. Our future work will

apply the proposed framework to multi-hop wireless networks by extending it to the contention window control.

References

[1] Research and Markets, "In-Depth Analysis: Consumer Electronics Devices Warming up to Wi-Fi to their offering." http://www.researchandmarkets.com/reports/c43740, Mar. 2006. Article(CrossRefLink).

[2] Chunyu Hu, Hwangnam Kim, Jennifer C. Hou, Dennis Chi and Sai Shankar N, "A Distributed Approach of Proportional Bandwidth Allocation for Real-Time Services in Ultra Wide Band (UWB) WPANs," *IEEE Transactions on Parallel and Distributed System*, vol. 21, no. 11, Nov. 2010.Article (CrossRef Link).

[3] Seok-hoon Hong, Yong-hun Lee, Jae-Yoon Jung and Doug Young Suh, "A Cross-Layer Approach to Fair Resource Allocation for Multimedia Service in WiMAX," *KSII Transactions on Internet and Information Systems*, vol. 4, no. 6, pp. 1006-1022, 2010. Article (CrossRef Link)

[4] XiaohuGe, Cheng-Xiang Wang, Yang Yang, Lei Shu, Chuang Liu and Lin Xiang, "AFSO: An Adaptive Frame Size Optimization Mechanism for 802.11 Networks," *KSII Transactions on Internet and Information Systems*, vol. 4, no. 3, pp. 205-223, 2010. Article (CrossRef Link).

[5] S. Pilosof, R. Ramjee, D. Raz, Y. Shavitt and P. Sinha, "Understanding TCP fairness over wireless LAN," in *Proc. of IEEE INFOCOM*, 2003. Article (CrossRef Link).

[6] Y. Wu, Z. Niu and J. Zheng, "Study of the TCP upstream/downstream unfairness issue with per-flow queuing over infrastructure-mode WLANs," *Wireless Communications and Mobile Computing*, vol. 5, no. 4, pp. 459–471, 2005. Article (CrossRef Link).

[7] D.J. Leith, P. Clifford, D. Malone and A. Ng, "TCP fairness in 802.11e WLANs," IEEE Communications letters, vol. 9, no. 11, pp. 964–966, 2005. Article (CrossRef Link).

[8] A.M. Kholaif and T.D. Todd, "WlanVOIP capacity allocation using an adaptive voice packetization server," *Computer Communications*, vol. 30, no. 13, pp. 2661 – 2675, 2007. Article (CrossRef Link).

[9] J. Yoo, H. Luo, and C. kwon Kim, "Joint uplink/downlink opportunistic scheduling for wi-fiwlans," *Computer Communications*, vol. 31, no. 14, pp. 3372 – 3383, 2008. Article (CrossRef Link).

[10] IEEE 802.11 WG, "Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specification," *IEEE 802.11 Standard*, 1999. Article (CrossRef Link).

[11] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal of Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, Mar. 2000.Article (CrossRef Link).

[12] F. Cali, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE802.11 protocol to achieve a theoretical throughput limit," *IEEE Trans. On Networking*, vol. 8, no. 6, pp. 785–799, Dec. 2000. Article (CrossRef Link).

[13] G. Tan and J. Guttag, "Time-based fairness improves performance in multi-rate WLANs," in *Proc. of USENIX*, 2004. Article (CrossRef Link).

[14] A. Banchs, P. Serrano and H. Oliver, "Proportional fair throughput allocation in multi-rate IEEE 802.11e wireless LANs," *Wireless Networks*, vol. 13, pp. 649–662, Oct. 2007. Article (CrossRef Link).

[15] Shinuk Woo and Hwangnam Kim, "Estimating Link Reliability in Wireless Networks: An Empirical Study and Interference Modeling," in *Proc. of IEEE INFOCOM 2010 (mini-conference)*, San Diego, USA, Mar. 2010. Article (CrossRef Link).

[16] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. on Networking*, vol. 1, no. 4, pp. 397–413, 1993. Article (CrossRef Link)

[17] K. Ramakrishnan and S. Floyd, "A proposal to add explicit congestion notification (ECN) to IP," *IETF RFC 2481*, Jan. 1999. Article (CrossRef Link).

[18] E.C. Park, H. Lim, K.J. Park and C.H. Choi, "Analysis and design of the virtual rate control algorithm for stabilizing queues in TCP networks," *Computer Networks*, vol. 44, no. 1, pp. 17–41,

Jan. 2004. Article (CrossRef Link).

[19] IEEE 802.11 WG, "Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specification: Medium access control (MAC) quality of service (QoS) enhancements," *IEEE 802.11e/D6.0*, Nov. 2003. Article (CrossRef Link).

[20] F. Cali, M. Conti and E. Gregori, "Dynamic tuning of the IEEE802.11 protocol to achieve a theoretical throughput limit," *IEEE Trans. On Networking*, vol. 8, no. 6, pp. 785–799, Dec. 2000. Article (CrossRef Link).

[21] E.J. Lee, H.T. Lim, S.J. Seok and C.H. Kang, "A scheme for enhancing TCP fairness and throughput in IEEE 802.11WLANs," in *Proc. of IFIP NETWORKING*, 2007. Article (CrossRef Link).

[22] M. Gong, Q. Wu and C. Williamson, "Queue management strategies to improve TCP fairness in IEEE 802.11 wireless LANs," in *Proc. of workshop on Resource Allocation in Wireless Networks (RAWNET)*, 2006. Article (CrossRef Link).

[23] X. Lin, X. Chang and J.K.Muppala, "VQ-RED: An efficient virtual queue management approach to improve fairness in infrastructure WLAN," in *Proc. of IEEE Local Computer Networks (LCN)*, 2005. Article (CrossRef Link).

[24] N. Blefari-Melazzi, A. Detti, I. Habib, A. Ordine and S. Salsano, "TCP fairness issues in IEEE 802.11 networks: Problem analysis and solutions based on rate control," *IEEE Trans. on Wireless Communications*, vol. 6, no. 4, pp. 1346–1355, Apr. 2007. Article (CrossRef Link).

[25] H. Kim and J.C. Hou, "Improving protocol capacity with model based frame scheduling in IEEE 802.11-operated WLANs," in *Proc. of ACM MOBICOM 2003*, Sep. 2003. Article (CrossRef Link).

Appendix

We summarize the procedure of deriving the (normalized) link capacity in IEEE 802.11-based network using the p-persistent model. Note that the capacity is used for calculating fair per-device link usage. Even though there are several analytical models [9][10][21], we choose the Cal`ı *et al.* model to determine the link capacity in the proposed MAP-based framework. Cal`ı's model focuses on the interval between two consecutive successful transmissions of data framest_v, referred to as virtual transmission time. Based on the observation that the wireless link experiences alternating idle periods and collision periods till ith as a successful transmission with an additional idle period, the model firstly represents the link activities as a renewal process and sets the renewal point as the end of each successful transmission. Then, the expected virtual transmission time can be expressed as

$$E[t_V] = E[N_C] \cdot t_C + E[t_{idle}] \cdot (E[N_C]+1) + t_D \tag{A-1}$$

where N_C is the number of collisions taken before a successful transmission, t_C is one collision period consisting of one collided-frame time and an extended inter-frame space time, t_{idle} is the duration of consecutive idle slots (between two busy periods, which are any combination of a successful transmission and a collision), t_D is the successful transmission time which includes two short inter-frame space times, one data frame transmission time, and one ACK frame time. After finding out the successful transmission probability P_S, the collision probability P_C, and the idle probability P_I, all of which depend on the number of nodes that attempt to transmit their frames (see [10] for a detailed account of the derivation), both N_C and t_{idle} are determined according to the geometric distribution with parameters $P_S/(P_S + P_C)$ and $1-P_I$, respectively. Hence,

$$E[N_C] = \frac{P_S}{P_C} \text{ and } E[t_{idle}] = \frac{P_I}{1-P_I} t_{slot}. \tag{A-2}$$

Plugging (A-2) into (A-1), we obtain the expression of $E[t_V]$. The link (normalized) capacity μ can now be computed by dividing the payload over $E[t_V]$:

$$\mu = \frac{E[t_{payload}]}{E[t_V]}, \qquad (A-3)$$

where $E[t_{payload}]$ is the average size of payload in time units. With the aforementioned procedure, the home gateway determines the current utilization with the current number of devices N and the average payload size $t_{payload}$ every T_{dev} interval. Note that N and $t_{payload}$ can be easily kept track of at the gateway and then both of them can be plugged into the procedure to determine the current (theoretically) available link capacity.

Eun-Chan Park received his Ph.D degree from the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, Korea, in2006. He is currently an assistant professor at Dongguk University. His research interests include performance analysis, resource allocation, and quality of service, and congestion control, cross-layer optimization in wired and wireless networks.

NojunKwak received his B.S., M.S., and Ph.D. degrees from the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, Korea, in 1997, 1999 and 2003 respectively. During his Ph. D. study, he visited Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign one year from 2001 to 2002. From 2003 to 2006, he worked for Samsung Electronics. In 2006, he joined Seoul National University as a BK21 assistant professor. Since 2007, he has joined Ajou University, Suwon, Korea where he is currently an associate professor. His research interests include pattern recognition, neural networks, machine learning, data mining, image processing, and their applications.

Suk Kyu Lee received his B.S. degree in Computer Science at University of Illinois at Urbana-Champaign in 2009. Since 2009, he is a graduate student in School of Electrical Engineering at Korea University. He had worked at NCSA, National Center for Supercomputing Application, at University of Illinois at Urbana-Champaign from 2007 to 2009 as an undergraduate research assistant. His research interests are in the areas of network virtualization and wireless networks and multimedia data streaming.

Jong-Kook Kim is currently an Associate Professor at Korea University, Seoul Korea. He received his M.S. and Ph.D. degrees in electrical and computer engineering from Purdue University in May 2000 and August 2004, respectively. He received his B.S. degree in electronic engineering from Korea University, Seoul, Korea in 1998. He has worked at Samsung SDS's IT R & D Center from 2005 to 2007. His research interests include heterogeneous distributed computing, real-time mobile computing, computer architecture, performance measures, resource management, evolutionary heuristics, energy-aware computing, efficient computing, shipboard computing, and distributed compilers. He is a member of the IEEE, IEEE Computer Society, and ACM.

Hwangnam Kim received his Ph.D. degree in Computer Science at the University of Illinois at Urbana-Champaign in 2004. He has been currently an associate professor in the School of Electrical Engineering at Korea University, Korea since 2006. His research focus is in the areas of network modeling and simulation, wireless sensor networks, community wireless network, pervasive/ubiquitous computing, and cyber physical system.