Polytechnic
UNIVERSITY

Incentive Engineering in Wireless LAN Based Access Networks

Raymond R.-F. Liao, Rita H. Wouhaybi and Andrew T. Campbell

November 2002

WICAT TR 02-001

WICAT
WIRELESS INTERNET
Center for Advanced Technology

Incentive Engineering in Wireless LAN Based Access Networks

Raymond R.-F. Liao, Rita H. Wouhaybi, and Andrew T. Campbell
Dept. of Electrical Engineering, Columbia University, New York, NY10027, USA.
E-mail: {liao, rita, campbell}@comet.columbia.edu

Abstract

Traffic regulation in public and private wireless LANs face a number of significant challenges, particularly in commercial networks where there is a need for efficient regulation of bursty transactional applications, support for bandwidth reservation services while inhibiting bandwidth hogging by mobile devices, and incentivizing user cooperation. In this paper, we take a new approach to solving these problems by applying incentive engineering techniques to wireless access networks. We design two incentive-based allocation service classes: an instantaneous allocation (IA) class, which provides better throughput, and a stable allocation (SA) class, which provides better allocation stability. Our approach possesses a number of beneficial properties including minimizing the algorithmic and protocol overhead on mobile devices, Nash bargaining fairness for the IA service, and incentive compatibility for mobile users promoting the truthfully selection of service class and bandwidth declaration. We use analysis, simulation and experimental results from a wireless testbed to demonstrate the effectiveness of wireless incentive engineering[1].

1. Introduction

The recent success of IEEE 802.11 [18] wireless local area networks (WLANs) has led to the emergence of wireless internet service providers offering wireless access services at hot spots. The onset of high speed WLAN-based access services, while complementing cellular data services, also highlights the lingering problem of how to price wireless data. Market evidence has shown that the prevailing charging model for wireless access service is block-rate charging [17, 1], which comprises a fixed charge for usage within a block of air time or bytes delivered, and a higher flat rate for any usage that exceeds the block amount. However, this type of charging model is not sensitive to the difference between stable allocation for real-time applications and best effort allocation for data applications. Therefore, without an incentive structure a stable allocation service can be easily overtaken by data applications, leading to the "tragedy of the commons" phenomenon [7].

The current engineering approach taken by cellular networks introduces a tightly controlled (e.g., circuit-based) environment for both wireless voice and data. However, this approach does not scalable well given the increasing diversity of applications and device programmability emerging. In contrast, WLAN inherits both the simplicity and the best-effort service model of the Internet. However, in order to deliver better than best-effort services (e.g., IEEE 802.11e) in a WLAN-based access networks, there is a need for rate regulation techniques [2]. These techniques include traffic shaping at both mobile devices and access points, with the addition of admission control at network access points to enforce service differentiation and fend off any potentially abusive usage such as bandwidth hogging or denial of service attacks. However, access rate regulation is very challenging to get right. Static rate regulation mechanisms are too simple to efficiently control bursty transactional applications such as web browsing, and measurement-based schemes can potentially generate large amounts of control messaging.

In this paper, we address the lack of incentive dilemma and rate regulation challenges discussed above with a market-based mechanism. We introduce a service control parameter called *service purchasing power*, which plays the same role as a service budget but covers the internal price of resource usage. By defining service purchasing power as a non-accumulated and non-replenished budget, we incentivize users to self-differentiate based on their application needs, and reduce the need for per-mobile rate-control messaging. Instead, a *price-service menu* is periodically broadcast to direct the rate adjustment of all mobile devices in a cell. The price-service menu comprises two incentive-based service classes: an *instantaneous allocation (IA)* class, which provides better throughput, and a *stable allocation (SA)* class, which provides better allocation stability.

Our work is inspired by the seminal work of Drexler and Miller [5] on mechanism design for operating systems for the dual purpose of inducing cooperative behavior over computational resources, reducing market transaction cost. In economics theory, incentive engineering stems from the discipline of mechanism design, which structures the users' strategy space such that a user's self-optimizing choice of action is "incentive compatible" with the system optimization goal. One typical example is the Vickery-Clarke-

[1]This work is sponsored by the NSF WIRELESS TECHNOLOGY Award ANI-9979439 and the NYSTAR Wireless Internet Center for Advanced Technology (WICAT).

Groves mechanism [6]. Our work can also be viewed as a continuation of the argument advocated in [19]; that is, monetary charge for network service is better based on systems level architectural issues rather than "economically optimal" marginal cost. When pricing is non-monetary, existing congestion pricing mechanisms [14, 10, 13] are not applicable because non-cooperative users have no incentive to truthfully respond to a "price signal" offered by the network control system. Rather than maximizing the difference between the utility and cost functions, users will solely maximize their utility functions and ignore the cost functions as long as the non-monetary cost is below the non-monetary budget. Consequently, our incentive engineering mechanisms turn this non-cooperative game into an equivalence of a Nash bargaining solution [20, 15], whose operating point has better properties, (i.e., Pareto optimum and Nash bargaining fair), than the Nash equilibrium operating point for a non-cooperative congestion pricing market.

The IA and SA service classes trade off the average amount of allocated bandwidth with allocation stability; that is, a price-service menu provides a ranking of service classes with decreasing bandwidth allocation stability and per-unit internal price, but with increasing prospects for higher average bandwidth allocation. As a result, data applications can opt for the IA service class, which will on average, offer better bandwidth allocation to sessions, but at the cost of more instability in the offered bandwidth. To offset this instability in allocated bandwidth, real-time applications seeking better service quality can pay a premium to use the SA service class, which provides stable bandwidth allocation but usually results in smaller amounts of allocated bandwidth to a session in comparison to the IA service. There lies the inherent trade off in the offered services between the two classes.

There are two ways to design service models that promote self-differentiation among users: differentiated pricing or service classes tailored toward specific user groups. The Paris Metro Price [16] is an example of using two-tier pricing to realize differentiated services without any additional network mechanisms. In contrast, the Alternative Best Effort service proposed in [8] is a good example of designing two alternative service classes, each of which is preferred by data and multimedia users, respectively. Our service model employs both methods. We design differentiated pricing to regulate demand for stable allocation, and differentiated service classes by considering the tradeoff between allocation stability and allocation quantity. Unlike [8] that leverages the trade off between packet loss and throughput but requires modification of packet schedulers, our service differentiation is at the session level and is therefore independent of any packet scheduler.

The rate control for IA service is measurement-based to efficiently regulate bursty transactional applications. The enforcement algorithm only resides inside the networks and does not require users to estimate their own bandwidth demands. We approach the bandwidth reservation problem with a "soft" guarantee on the length of SA service reservation. The "softness" of the guarantee follows the rank of the reservation's internal bid price, which is proportional to the SA service purchasing power. Users with higher SA service purchasing power will less likely see early termination of their reservations. As a result, the SA class does not require users to predict their session lifetime. To make the scheme more usable, a "warning interval" is implemented for sessions in danger of early termination to renegotiate. This feature practically mirrors application operating conditions because typically, applications cannot predict session bandwidth demands nor the session duration in advance.

In mobile and wireless networks, market mechanisms have been applied in a very limited manner. In [3], a revenue framework is proposed to resolve some of the adaptation policy trade-offs. The scheme provides incentives for adaptation by charging sessions that benefit from the adaptation, while compensating sessions that suffer from adaptation. However, the exact calculation of credits and charges are challenging and are not formulated in [3]. In [9], the authors analyze the property of the Paris Metro Pricing [16] scheme within the context of wireless access service. The service offering is limited with no support for allocation stability.

The structure of the paper is as follows. In Section 2 we present an overview of our framework including the allocation price-service menu and messaging protocol. We discuss the incentive-based control algorithms for IA and SA classes in Section 3 and Section 4, respectively. In Section 5, we present the properties of our incentive engineering scheme in the context of the mobile user dominant strategy. We demonstrate that the best strategy for a user is to cooperate with network traffic control. In Section 6, we evaluate our algorithms in an experimental wireless testbed that also supports an emulation capability, which further helps evaluate the system under different conditions and scenarios. Finally, we present some concluding remarks in Section 7.

2. Framework Overview

2.1. Network Model

Figure 1 illustrates a wireless access network architecture in the context of IEEE 802.11b WLAN networks. Note, that the particular cellular network topology shown is for illustration only, and not essential to our framework. We use the terms "mobile device" and "access point" in a generic sense. At access points, per-mobile and per-class traffic regulators are used to regulate downlink traffic. In addition, each mobile device optionally uses per-class traffic regulators in the form of policers or shapers to self-regulate uplink traffic. User profiles containing service specific resource allocation policies are stored at the Authentication, Authorization and Accounting (AAA) server at the mobile device's home network, and delivered to a visiting network by a mobility management protocol. We assume that there is a broadcast channel at the media access control (MAC) layer

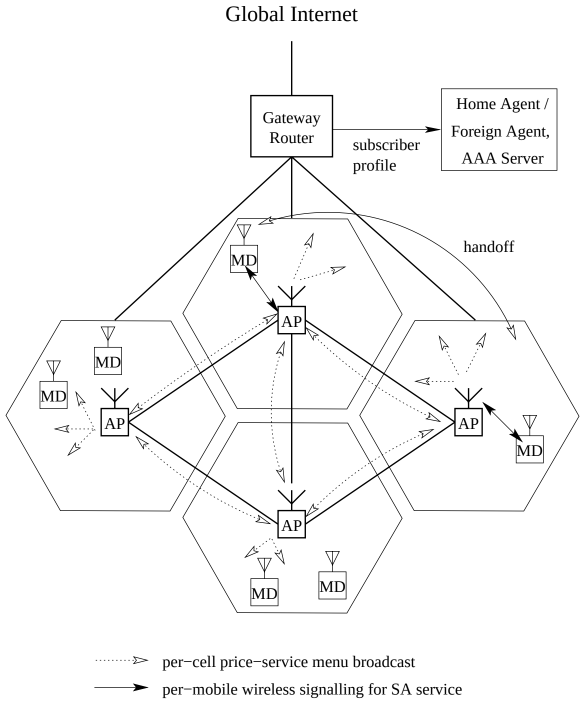


Figure 1. Wireless LAN Based Mobile Access Network

from the access point to all the mobile devices in a particular cell. Our incentive engineering mechanisms are applied at the session bandwidth allocation level, involving traffic regulator modules at mobile devices and access points. Fast time scale packet scheduling algorithms are not affected.

2.2. Service Purchasing Power

Typically, service budget (e.g., the number of "free" minutes within a service plan) is not allowed to be accumulated, otherwise, idle users could carry-over large amounts of unused budget, distorting the market mechanism. Therefore, users have no incentive to conserve budget toward the end of the budget replenishing cycle, and could subsequently start a spending spree, distorting the market mechanism as well.

To address this problem we introduce a parameter called *service purchasing power*, which plays the same role as a service budget but covers the internal price of resource usage. By defining service purchasing power as a non-accumulated and non-replenished budget, we avoid the difficulty of budget control. With the service purchasing power known to the network, the user strategy space is essentially constrained by the price-service menu, which induces user cooperation, avoiding over allocation of bandwidth, and enforcing the truthful declaration of reservation bandwidth. Allowing users to choose between service classes in this manner helps promote self-differentiation among user applications, enabling differentiated resource allocation.

Each user (i.e., mobile device) is assigned a service purchasing power ϑ_i, which plays the role of a "constant budget". The value of ϑ_i for each user is determined by the by the network. Each mobile device partitions its service purchasing power as it wishes into portions for IA and SA allocation, denoted as $\vartheta_{i,I}$ and $\vartheta_{i,S}$, respectively. Since access points are aware of ϑ_i and the corresponding allotment to SA reservation $\vartheta_{i,S}$ through a mobile device initiated reservation request, the IA portion of service purchasing power can be derived as,

$$\vartheta_{i,I} = \vartheta_i - \vartheta_{i,S}. \quad (1)$$

2.3. Price-Service Menu

Our incentive engineering mechanisms use a market-based price to distribute allocation information and regulate bandwidth usage. Each access point l periodically broadcasts a non-monetary price-service menu within its cell driven by price change. The price-service menu comprises the price of the IA and SA classes, $p_{l,I}$ and $p_{l,S}$ respectively, as well as $p_{l,H}$, the price for a subclass of SA called *handoff allocation (HA)*. The HA class enforces price differentiation that any mobile devices with $\vartheta_{i,S} < p_{l,H}$ will be denied handoff for lack of service purchasing power. We represent the price-service menu of access point l in a price vector: $\langle p_{l,I}, p_{l,S}\rangle$. The SA class requires a per-mobile reservation message between a mobile device and its access point as shown in Figure 1. An SA reservation has three parameters: the requested bandwidth quantities b_{i,S_U}, b_{i,S_D} for uplink and downlink, respectively, and the allotted SA portion of service purchasing power $\vartheta_{i,S}$.

3. Incentive Engineering for IA Class

3.1. Baseline IA Algorithm

The baseline price calculation for the IA class is based on the aggregated price-demand function, where the IA price $p_{l,I}$ is interpreted as a common allocation signal for users with different service purchasing power.

Since service purchasing power is a non-accumulated and non-replenished budget, a mobile device i has no benefit in conserving its IA portion of the service purchasing power $\vartheta_{i,I}$. Therefore, the IA price-demand function of each mobile device i is:

$$b_{i,I} = \min\left\{\vartheta_{i,I}/p_{l\ni i,I}(t)\ ,\ b_{i,I}^{\max}\right\}. \quad (2)$$

Here with an abuse of notation, we use $l \ni i$ to denote the cell l where mobile device i is active. $b_{i,I}^{\max}$ is the maximum bandwidth of the IA class (e.g., the wireless channel capacity) that mobile device i may consume.

Summing up both sides of (2) for all the users i, we have the aggregated price-demand function,

$$q_{l,I} = \sum_{i\in l} b_{i,I} = \sum_{i\in l} \min\left\{\vartheta_{i,I}/p_{l,I}(t)\ ,\ b_{i,I}^{\max}\right\}, \quad (3)$$

where $q_{l,I}$ denotes the total available bandwidth for the IA class in cell l.

When all the $b_{i,I}^{\max}$ are set to channel capacity, the IA price can be simply derived from Equation (3) as:

$$p_{l,I} = \left(\sum_{i \in l} \vartheta_{i,I} \right) / q_{l,I}, \quad (4)$$

The allocation procedure follows two steps: access points use Equation (3) to update the IA price and broadcast price-service menu; mobile devices then use Equation (2) to derive their IA allocations.

3.2. Measurement-based Price Calculation

The resulting allocation, however, could be largely under utilized by short-lived IA applications, such as, web transactions. A straightforward solution to this problem is to dynamically adjust $b_{i,I}^{\max}$ based on the measured bandwidth usage of each mobile device. Since $b_{i,I}^{\max}$ plays the role of limiting bandwidth allocation to a mobile device, by reducing $b_{i,I}^{\max}$ for a mobile device with light usage we can distribute the reduced bandwidth amount to heavy users, and hence, increase the overall bandwidth utilization. For clarity, we denote the potentially time-varying $b_{i,I}^{\max}$ as $b_{i,I}^{\max}(t)$. To handle the boundary condition caused by varying $b_{i,I}^{\max}(t)$, we introduce $\zeta_{i,I}(t)$, the IA unit "bid price" for a mobile device i as

$$\zeta_{i,I}(t) = \vartheta_{i,I} / b_{i,I}^{\max}(t). \quad (5)$$

We sort $\zeta_{i,I}$ in descending order, and denote the kth highest IA unit bid price as $\zeta_{(k),I}$. In addition, we denote $\mathcal{B}(k)$ the subset of users whose $\zeta_{i,I}$ are among the top k. Subsequently, Equation (3) can be formatted as:

$$q_{l,I} = \frac{\sum_{i \in \overline{\mathcal{B}}(k)} \vartheta_{i,I}}{p_{l,I}} + \sum_{i \in \mathcal{B}(k)} b_{i,I}^{\max}(t), \quad (6)$$

where $\zeta_{(k+1),I} \le p_{l,I} < \zeta_{(k),I}\ ,\ k = 0, 1, \ldots, N-1,$

and N is the total number of user. In addition, $\zeta_{(0),I} \triangleq \infty$.

By inverting this equation, we have the following formula for calculating the IA price:

$$p_{l,I} = \frac{\Theta_{all,I} - \Theta_{k,I}}{q_{l,I} - \mathbf{b}_{k,I}}\ \ ,\ \ \mathbf{q}_{k,I} < q_{l,I} \le \mathbf{q}_{k+1,I}, \quad (7)$$

where the partial sums are defined as follows:

$$\Theta_{k,I} \triangleq \sum_{i \in \mathcal{B}(k)} \vartheta_{i,I}\ \ ,\ \ \Theta_{all,I} \triangleq \sum_{i} \vartheta_{i,I}, \quad (8)$$

$$\mathbf{b}_{k,I} \triangleq \sum_{i \in \mathcal{B}(k)} b_{i,I}^{\max}(t), \quad (9)$$

$$\mathbf{q}_{k,I} \triangleq \mathbf{b}_{k,I} + \frac{\Theta_{all,I} - \Theta_{k,I}}{\zeta_{(k),I}}. \quad (10)$$

Therefore, the additional work for access points to calculate the IA price is to maintain a sorted partial sums based on the IA bid price, and search for the bandwidth range

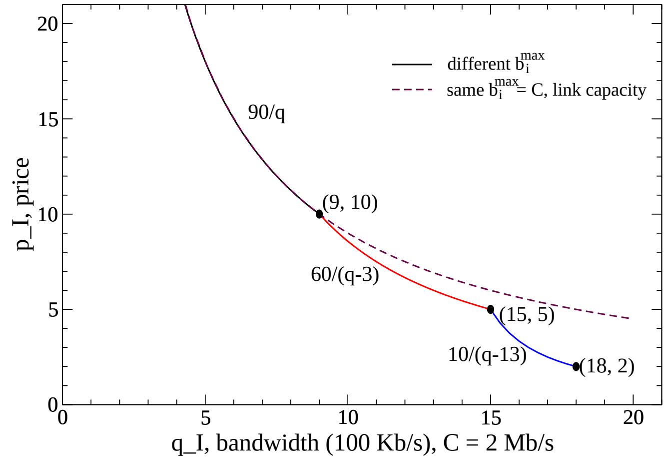


Figure 2. Example of Aggregated IA Price Function

$(\mathbf{q}_{k,I}\ \ ,\ \ \mathbf{q}_{k+1,I}]$ within which the available IA bandwidth $q_{l,I}$ fits.

The aggregated price function in (6) has a piecewise $1/q$ form. Figure 2 illustrates one example with three users, whose $(\vartheta_{i,I}, b_{i,I}^{\max})$ pair are (30, 3), (50, 10), and (10, 5), respectively, where the bandwidth unit is 100 Kb/s. The corresponding $\zeta_{i,I} = \frac{\vartheta_{i,I}}{b_{i,I}^{\max}}$ are 10, 5 and 2, sorted in descending order. These values lead to the first order break points in price, as shown in the figure. Consequently, the aggregated price function is a cascade of three functions: $\frac{90}{q}$, $\frac{60}{q-3}$, and $\frac{10}{q-13}$, respectively. In addition, we also show a curve (the dotted curve in Figure 2) where all $b_{i,I}^{\max}$ are the same, equal to the channel capacity of 2 Mb/s. This example illustrates the effect of adjusting $b_{i,I}^{\max}(t)$ on the price function. The dotted curve is always above the solid curve, which results from the measurement-based adjustment on $b_{i,I}^{\max}$, (i.e., $b_{i,I}^{\max}(t) \le b_{i,I}^{\max}$). The smaller IA price under $b_{i,I}^{\max}(t)$ leads to higher bandwidth allocation for active mobile devices and a higher overall bandwidth utilization.

3.3. Optimistic Rate Allocation with Incomplete Information

The measurement-based IA price calculation, however, requires a per-mobile messaging protocol to notify mobile devices and their corresponding access points of the changed $b_{i,I}^{\max}(t)$. Such an implementation would defeat our design goal of using only a single broadcast message for IA bandwidth allocation. In what follows, we present an enhancement to the IA pricing algorithm that tolerates incomplete information resulting from the reduction of messaging.

With the absence of per-mobile messaging to notify the change of $b_{i,I}^{\max}(t)$, each mobile device simply uses the constant value $b_{i,I}^{\max}$ to derive its allocation rate from the broadcasted price $p_{l,I}$, which is

$$b_{i,I}^{*} = \min\left\{ \vartheta_{i,I} / p_{l,I}^{*}\ ,\ b_{i,I}^{\max} \right\}. \quad (11)$$

Since $b_{i,I}^{\max}(t) \le b_{i,I}^{\max}$, comparing the definitions of $b_{i,I}$

and $b^*_{i,I}$ in (2) and (11), we have $b^*_{i,I} \geq b_{i,I}$. In the worst case, when every mobile device uses up its entire allocation $b^*_{i,I}$, the wireless link will be overloaded by a ratio of $p_{l,I}/p^*_{l,I}$, where $p^*_{l,I}$ is calculated from Equation (7) replacing $b^{\max}_{i,I}(t)$ with $b^{\max}_{i,I}$.

The rate allocation algorithm tolerates this discrepancy at mobile devices due to incomplete information. It "optimistically" controls the extent of over-allocation by measuring the actual system load and adjusting $b^{\max}_{i,I}(t)$ adaptively. The measurement algorithm operates over discrete time t_n slotted by τ, the same measurement window used for demand measurements. τ is limited by the response time of the control system to change the regulator shaping rate. In Section 6.2, we will measure the minimum value of τ sustainable in an experimental wireless testbed. The algorithm measures the up and downlink average rates $\bar{b}^{up}_{i,I}(t_n)$ and $\bar{b}^{down}_{i,I}(t_n)$, respectively, over the interval $(t_{n-1}\ ,\ t_n]$.

The value of $b^{\max}_{i,I}(t_n)$ is calculated according to the measured average rate $\bar{b}_{i,I}(t_n) = \bar{b}^{up}_{i,I}(t_n) + \bar{b}^{down}_{i,I}(t_n)$ in the past τ interval:

$$b^{\max}_{i,I}(t_n) = \min\{\gamma \bar{b}_{i,I}(t_n)\ ,\ b^{\max}_{i,I}\}. \tag{12}$$

Here $\gamma \geq 1$ controls the extent of over-allocation. When $\gamma = 1$, $b^{\max}_{i,I}(t)$ is calculated based on the average rate, which leads to the maximum extent of over-allocation. When $\gamma \gg 1$, $b^{\max}_{i,I}(t) = b^{\max}_{i,I}$, namely the adjustment is disabled and no over-allocation is allowed.

The IA traffic load measurement is calculated with respect to the actual usage (not the amount of reservations) of the SA traffic, that is:

$$\rho_{l,I} = \frac{\sum_i \bar{b}_{i,I}(t_n)}{\mathcal{C}(1 - \rho_{l,S}(t_n))}, \tag{13}$$

where $\mathcal{C}$ is the channel capacity, and $\rho_{l,S}(t_n)$ is SA traffic load. The value of γ is adjusted based on the system load condition ρ as follows:

$$\gamma(t_n) = \begin{cases} \min\{2\gamma(t_{n-1})\ ,\ \gamma_{\max}\} & \rho > threshold \\ \max\{1,\ \gamma(t_{n-1})(1 - dec)\} & \rho < \kappa * threshold; \\ \gamma(t_{n-1}) & otherwise. \end{cases} \tag{14}$$

where $\gamma_{\max} \triangleq \max\{b^{\max}_{i,I}/b^{\max}_{i,I}(t) \mid b^{\max}_{i,I}(t) > 0\}$. Here $\gamma_{\max}$ caps the value of γ because increasing γ beyond $\gamma_{\max}$ has no effect when all the mobile devices have $b^{\max}_{j,I}(t)$ reaching the absolute maximum $b^{\max}_{j,I}$.

The goal of Equation 14 is to keep ρ within a range (i.e., between κ and 100%) of the threshold load that triggers excessive delay, as discussed in [2]. In this paper, we set the $threshold$ value to 90%. When ρ exceeds the threshold load, γ is doubled over every τ interval to quickly reduce the extent of over-allocation. When ρ falls below $\kappa * threshold$, γ is reduced by a factor of dec until reaching 1. The purpose of this is to increase the extent of over-allocation (i.e., broadcasting a smaller value of $p^*_{l,I}$), encouraging bandwidth usage. In Section 6.2, we will experiment with the setting of the parameters κ and dec. As a safe guard against frequent variations of $p^*_{l,I}$, we introduce a control parameter $\delta = 5\%$ such that $p^*_{l,I}(new)$ is only broadcast when the change is larger than δ percentage, (i.e., $|1 - p^*_{l,I}(old)/p^*_{l,I}(new)| > \delta$).

The rate enforcement procedure also executes only at the access points and without the participation of mobile devices. In WLAN environments, this means traffic regulation is performed only on the downlink. Since mobile devices are only aware of $b^*_{i,I}$, the traffic regulation algorithm uses $b^*_{i,I}$ rather than $b_{i,I}$ as the rate limit. The peak rate of downlink shaping is set as follows:

$$b^{down}_{i,I}(t_n) = \begin{cases} b^*_{i,I} - \bar{b}^{up}_{i,I}(t_n) & \bar{b}^{up}_{i,I}(t_n) \leq b^*_{i,I}; \\ & otherwise, \text{(deny any} \\ 0 & \text{downlink access).} \end{cases} \tag{15}$$

This approach is effective enough to control web browsing because of the asymmetry of web traffic with most traffic over the downlink. For heavy uplink users, enforcement is indirectly performed by stopping downlink traffic. If this approach still fails, the corresponding mobile device will be treated as non-compliant, and any future access to network services can be denied. Note, that this downlink-only shaping approach is also practical for the SA reservations, as discussed in the next section.

4. Incentive Engineering for SA Class

4.1. Baseline SA Algorithm

The SA reservation message comprises the triplet of uplink and downlink bandwidth quantities, and the service purchasing power $(b_{i,S_U}, b_{i,S_D}, \vartheta_{i,S})$. Unlike the IA class, an SA bandwidth request needs to pass admission control based on resource availability and mobility prediction, which is extended from the conventional handoff admission control algorithms found in the literature [4]. Once admitted, the allocation is guaranteed as long as the corresponding SA unit bid price satisfies $\zeta_{i,S} \geq p_{l \ni i,S}(t)$, where,

$$\zeta_{i,S} \triangleq \frac{\vartheta_{i,S}}{b_{i,S_U} + b_{i,S_D}}, \tag{16}$$

and $p_{l \ni i,S}(t)$ denotes the non-monetary price of SA class in cell l where the mobile device i is active.

When $\zeta_{i,S} < p_{l \ni i,S}(t)$, mobile device i's SA allocation is considered to be *under probation*. In this case, the allocation guarantee is revoked only when the SA allocation has been continuously under probation for an interval of T_S. Therefore, T_S is the minimum interval over which an SA allocation is guaranteed, and applications have at least T_S amount of time for rate-adaptation or renegotiation.

Figure 3 shows the price-demand functions for both IA and SA classes, with the IA price-demand function shown from right to left and the SA price-demand function from left to right. The intersection of these two functions gives the spot IA price[2] $p_{l,I}$. The figure also shows the decreas-

[2]The actual IA price is lower because unused SA bandwidth is also allocated to IA traffic.

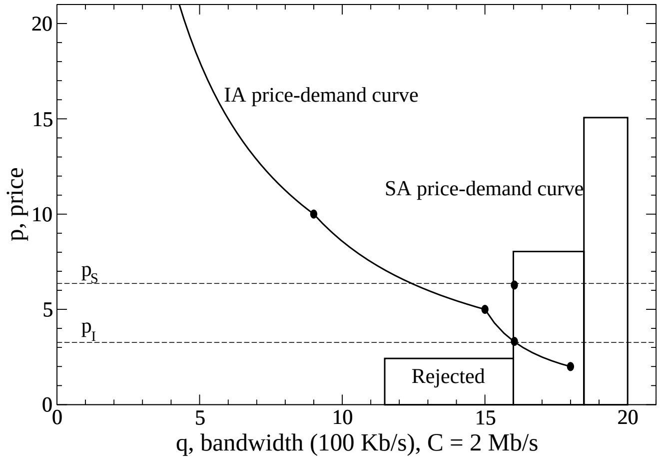


Figure 3. Example of Aggregated IA & SA Price Function

ing allocation reliability for the SA class, following the descending bidding price. When $p_{l,S}$ increases, the SA sessions whose unit bid price $\zeta_{i,S} < p_{l,S}$ (e.g., the price block falls below the price line $p_{l,S}$ in the figure) will be put under probation. These mobile devices have a T_S interval to renegotiate for less quantity or increase $\vartheta_{i,S}$, the service purchasing power for the SA class. When T_S times out, the corresponding reservations will be revoked.

The SA price $p_{l,S}(t)$ is calculated based on the demand for the SA bandwidth from existing and handoff mobile devices. The purpose is to give preference to requests with higher bid prices. In addition, the IA price is considered as well to reduce the probability that it rises higher than the SA price, which leads to additional SA sessions entering probation, reducing the disincentive for switching from the IA to SA class.

The SA admission control algorithm measures traffic load conditional on the bid price $\zeta_{i,S}$ in order to support the price-differentiated admission procedure. We denote the measured SA bandwidth demand as $\overline{\lambda}(t|p_{l,S})$. Here λ denotes the demand over the measurement window τ. In practice, we quantized the price $p_{l,S}$ into $\{p_K\}$. Refer to [12] for the details of the quantization procedure. Subsequently, we have

$$\overline{\lambda}(t|p_k) = \alpha \frac{\text{cnt\_b}_k}{t - t_{n-1}} + (1-\alpha)\overline{\lambda}(t_{n-1}|p_k), \qquad (17)$$

where $t_{n-1} < t \leq t_n$, $k = 0, \ldots, K$ and cnt_b$_k$ is the sum of $b_{i,S}$ arrived within $(t_{n-1}\,,\,t]$ whose bid price $\zeta_{i,S} \in [p_k\,,\,p_{k+1})$.

We calculate $p_{l,S}(t, \zeta_{i,S})$ over the interval t_{out}, which is the shortest time-out interval among all sessions under probation. When there is no session under probation, $t_{out} = T_S$. Therefore, t_{out} is the minimum interval at the end of which additional SA bandwidth is guaranteed to be available. Note that the departure of SA sessions within the t_{out} interval is not counted on, because their occurrence is statistical with no guarantee.

The SA price chosen from the quantized price set $\{p_k\}$ needs to satisfy two constraints. The first constraint is that future SA demand regulated by price p_k should not exceed the available SA bandwidth, as shown in Inequality (18). The right-hand-side of the inequality is the available SA bandwidth. The left-hand-side of the inequality is the predicted average SA demand whose bid price is no less than p_k. The control parameter $\gamma > 0$ is used to adjust SA demand estimation.

$$\gamma t_{out} \sum_{i \geq k} \overline{\lambda}(t|p_i) \leq (1 - \rho_{l,S})\mathcal{C} \qquad (18)$$

The second constraint represented by Inequality (19) relates to the condition that the SA price should remain higher than the IA price throughout the t_{out} interval. The right-hand-side of the inequality is the estimated IA price when all the allowable SA demand is met.

$$p_k \geq \frac{\Theta_{all,I}}{\rho_{l,I}\mathcal{C} - \gamma t_{out} \sum_{i \geq k} \overline{\lambda}(t|p_i)}. \qquad (19)$$

The choice of $p_{l,S}$ is then decided as:

$$p_{l,S} = \min\{p_k \,|\, p_k \text{ satisfies Ineq } (18) \text{ and } (19)\}. \qquad (20)$$

The baseline SA allocation algorithm is shown in Figure 4.

---

admission_control { // arrival of reservation request
  update $\overline{\lambda}(t)$;
  if ($b_{i,S} > (1 - \rho_{l,S})\,\mathcal{C}$)
    reject(); // no enough bandwidth
  elseif ($\zeta_{i,S} < p_{l,S}$)
    reject(); // bid price too low
  else
    calculate_SA_price();
}
calculate_SA_price {
  update $(1 - \rho_{l,S})\mathcal{C}$; // available SA bandwidth
  update t_{out};
  search for the smallest p_k in the quantized
  price set $\{p_k\}$ such that
    $\gamma t_{out} \sum_{i \geq k} \overline{\lambda}(t|p_i) \leq (1 - \rho_{l,S})\mathcal{C}$;
    $p_k \geq \frac{\Theta_{all,I}}{\rho_{l,I}\mathcal{C} - \gamma t_{out} \sum_{i \geq k} \overline{\lambda}(t|p_i)}$;
  $p_{l,S} = p_k$;
  broadcast price-service menu;
}
maintain_SA_allocation {
  maintain sorted list of $\zeta_{(k),S}$;
  if ($\zeta_{i,S} < p_{l,S}$ AND i is not under probation)
    put i under probation, start timeout timer;
  if ($\zeta_{i,S} \geq p_{l,S}$ AND i is under probation)
    move i out of probation;
  if (i is under probation and timer expires)
    remove i's SA reservation;
}

Figure 4. Baseline SA Allocation Algorithm at Access Point

4.2. IA Allocation Pegging

So far we address the bandwidth-hogging problem by allowing users with higher bid prices to preempt the incumbent lower-price users after a warning interval. However, additional mechanisms are needed to discourage bursty data applications from switching from the IA to the SA service because preemption from the SA to IA service does not penalize bursty data applications.

One disincentive is the usage accounting model that can be used with the block-rate service charge. We can count the SA usage minutes by the holding time of the reservation regardless of actual bandwidth consumption. The second disincentive is the higher SA price over the IA price. The constraints (18) and (19) in Section 4.1 enforce that $p_{l,S}(t) > p_{l,I}(t)$. However, these two disincentives are insufficient to guarantee that the throughput offered to an IA service user will be always larger than the corresponding throughput the user would received using an albeit "hypothetical SA session". The complexity comes from the fact that there is no admission control for the IA service class traffic. The IA price (and subsequently the SA price) can rise sharply with a surge in IA demand. However, in the case of previously admitted SA sessions, their reservations will be maintained for T_S seconds when their SA unit valuation fall below SA price, (i.e., under probation with $\zeta_{i,S} < p_{l \ni i,S}$). Therefore, the same SA allocation stabilizing mechanism also provides an incentive for IA users to switch to the SA service if they have prior knowledge of the increase in IA demand.

To remove this arbitrage possibility, we explicitly calculate $\Gamma_i(t)$, the accumulated throughput surplus of an IA session in comparison to its hypothetical SA session. When $\Gamma_i(t)$ is in danger of becoming negative, the allocation for the IA session is pegged to the previous amount. The details of this algorithm are presented in [12], due to lack of space here.

5. Mobile Device Strategy

The incentive algorithms discussed in the previous section are designed to constrain a mobile device's strategy space while minimizing the amount of signalling overhead. In this section, we briefly discuss the properties of the resulting allocations. Refer to [12] for a detailed discussion.

Proposition 1 *The IA allocation mechanism is Nash Bargaining Fair [15] with different budget $\vartheta_{i,I}$ and maximum bandwidth $b_i^{\max}$.*

This is a special case (i.e., a single link allocation) of the asymmetric Nash Bargaining solution given in [20].

Proposition 2 *For wireless users preferring high throughput, the dominant strategy is to subscribe to the IA service.*

The best alternative strategy is the hypothetical SA session used in the allocation pegging algorithm, which subscribes to the SA service and aggressively renegotiates its bid quantity whenever $p_{l,S}$ is less than the previous bid price. However, the allocation pegging algorithm ensures that $\Gamma_i \geq 0$. Therefore, the accumulated throughput of an IA session is always the highest among any other alternative strategy.

Proposition 3 *For wireless users preferring allocation stability, the dominant strategy is to subscribe to the SA service.*

This is governed by the ranked allocation stability of SA service and the additional T_S warning interval when the bid price falls below $p_{l,S}$. In contrast, the corresponding IA service allocation is constantly changing and can fall below the SA bid quantity. Note, that the condition $\Gamma_i \geq 0$ does not prevent this because it only acts on the accumulated throughput, not the instantaneous throughput value. Because the service purchasing power ϑ_i is non accumulated, mobile devices have no incentive to save it. Since the SA allocation stability is ranked by $\vartheta_{i,S}/q_{i,S}$, inflating $q_{i,S}$ will reduce allocation stability, while deflating $q_{i,S}$ will affect application performance. Therefore, we have,

Proposition 4 *The dominant strategy for a wireless user of the SA service is to truthfully declare the required bandwidth amount $q_{i,S}$.*

For IA service users, the measurement-based allocation removes any strategic play by mobile devices. Because the access point subtracts $\vartheta_{i,S}$ from the remaining portion of the IA class, (i.e., $\vartheta_{i,I} = \vartheta_i - \vartheta_{i,S}$). The only decision remaining open to the mobile device is to decide how to split its service purchasing power ϑ_i into $\vartheta_{i,S}$ and $\vartheta_{i,I}$ amounts.

The actual partition of ϑ_i between the IA and SA service classes is determined by the utility function of a user, $u_i(q_{i,I}, p_{i,S})$, which is a function of the allocation quantity of the IA class, and the bid price (which is an indicator of allocation stability) of the SA reservation. Therefore, the optimum partition of ϑ_i is calculated by

$$\begin{aligned} \text{optimal\_}\vartheta_{i,S} &= \arg\max\{u_i(q_{i,I}, p_{i,S})\} & (21) \\ &= \arg\max\{u_i(\frac{\vartheta_i - \vartheta_{i,S}}{p_{l \ni i,I}}, \frac{\vartheta_{i,S}}{q_{i,S}})\} & (22) \end{aligned}$$

A mobile device's strategy is to decide optimal_$\vartheta_{i,S}$ based on its SA service demand $q_{i,S}$ and the IA service price $p_{l \ni i,I}$.

An example of $u_i(q_{i,I}, p_{i,S})$ can have a form of

$$u_i(q_{i,I}, p_{i,S}) = \alpha\, q_{i,I} + \beta\, p_{i,S} = \alpha \left(\frac{\vartheta_i - \vartheta_{i,S}}{p_{l \ni i,I}} \right) + \beta \left(\frac{\vartheta_{i,S}}{q_{i,S}} \right), \quad (23)$$

where α and β are control parameters. In this case,

$$\text{optimal\_}\vartheta_{i,S} = \begin{cases} \vartheta_i, & \beta/q_{i,S} > \alpha/p_{l \ni i,I} \\ 0, & \beta/q_{i,S} < \alpha/p_{l \ni i,I} \\ x, x \in [0, \vartheta], & otherwise \end{cases} \quad (24)$$

This example provides a good intuitive strategy: when the utility valuation for the stability of the SA allocation is more important, it is optimal to use all the service purchasing power to bid for an SA allocation; when the utility valuation for the amount of IA allocation is higher, the opposite is optimal. This simple strategy can be further enhanced: a mobile device can reduce the SA bid price and only increase it at the end of the probation interval T_S.

6. Experimental Results

6.1. Wireless Testbed

As a means to test the feasibility of the proposed algorithms, we implement the proposed algorithms in an experimental wireless testbed. Figure 5 shows our testbed. We use Linux PCs and laptops as access points and mobile devices, respectively. Access points are interconnected to each other using 10BaseT Ethernet, forming a wireless packet cellular network using IEEE 802.11b wireless radios. The access points rely on the Linux Traffic Control module (TC) for traffic shaping [11] in order to assign each mobile device its allocated bandwidth. We modified the IEEE 802.11 wireless device driver to enable traffic snooping for measuring the bandwidth consumption of each mobile device. Importantly, our testbed can also operates in a simulation/emulation mode so that the same algorithms can be evaluated with a larger number of access points and mobile devices.

6.2. Parameter Tuning

The first test conducted focuses on measuring the response of the Linux TC traffic shaper since the measurement-based algorithm relies heavily on the TC traffic shaper for restricting the bandwidth consumption of mobile devices. We measure the response of TC to changes in shaping rate. We repeated this test ten times with different rate values and observe a 16 ms response time. This value influences the choice of τ used in the IA measurement algorithm. However, after conducting several experiments on the testbed while varying the parameter τ we observed that decreasing it below 30 ms does not bring any benefit for the overall performance of the system and algorithms. Thus, the experiments discussed below uses 30 ms for the measurement interval and the price broadcasting period.

The second set of experiments focus on tuning the parameters dec and κ of Equation 14. Both parameters affect the tradeoff between improving the utilization of IA bandwidth unused by idle mobile devices, and reducing the chance of congestion when those idle mobile devices become active again. The experimental scenario comprises two mobile devices consuming IA bandwidth. The first mobile device downloads a large file while the second mobile

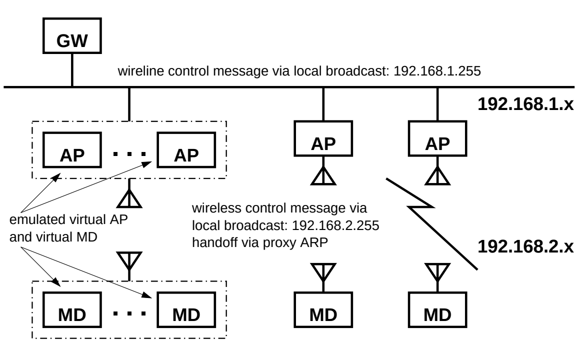
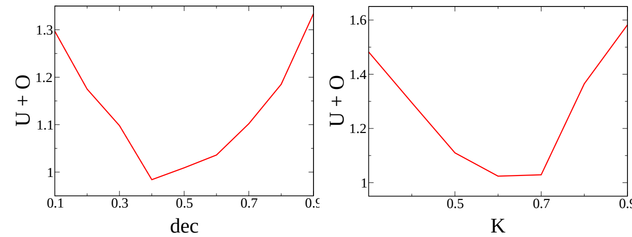


Figure 5. Experimental Wireless Testbed



(a) dec (b) κ

Figure 6. Parameter Setting

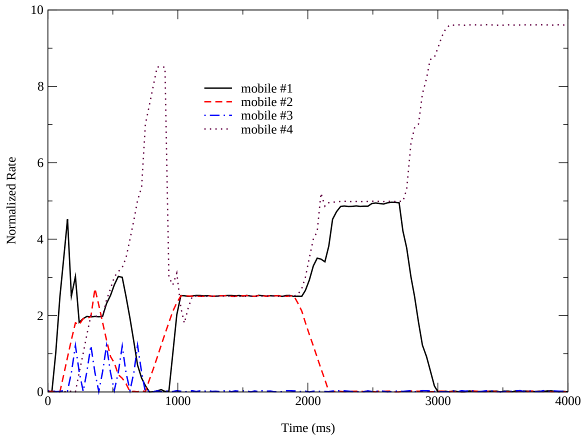
device performs a sequence of web transactions with on-off bursty traffic. When the second mobile device is idle (i.e., in an "off" interval), bandwidth could be under utilized. We measure the total amount of bits left unused during an "off" interval and denote it as U. When the second mobile device is active (i.e., in an "on" interval), congestion could happen. We measure the total amount of bandwidth allocation (i.e., $b_{i,I}^{\max}(t_n)$ in Equation 12) for both active mobile devices during an "on" interval and denote it as O. For the parameter dec, intuitively, a large dec leads to the quicker allocation of unused bandwidth to active mobile devices, but also to a larger chance of congestion when idle mobile devices become active. Therefore, a large dec means a smaller U but a larger O. In Figure 6(a), we plot the value $U + O$ against different values of dec. It shows that to minimize $U + O$, the optimal value of dec is 0.4.

We repeat the same experiments for κ in Equation 14, which decides the lower threshold to invoke adjustment of γ. A small κ means that the algorithm is satisfied with a lower bandwidth utilization, and not to redistribute unused bandwidth to active mobile devices. In contrast, a large κ will increase bandwidth utilization (reduce U) but also increase the chance of congestion (increase O) as well because the average load of the system will be operating at a higher level. In addition, a large κ also leads to more frequent oscillations in IA bandwidth and price changes. Figure 6(b) shows the evaluation results by plotting $U + O$ against different values of κ. Once again, we observe that to minimize $U+O$, the optimal choice of κ should be in the region of $[0.6, 0.7]$, which basically means that bandwidth should be redistributed to active mobile devices only when the bandwidth is 60% to 70% under the $threshold$ (0.9) value. In what follows, we set κ to 0.7 and dec to 0.4.

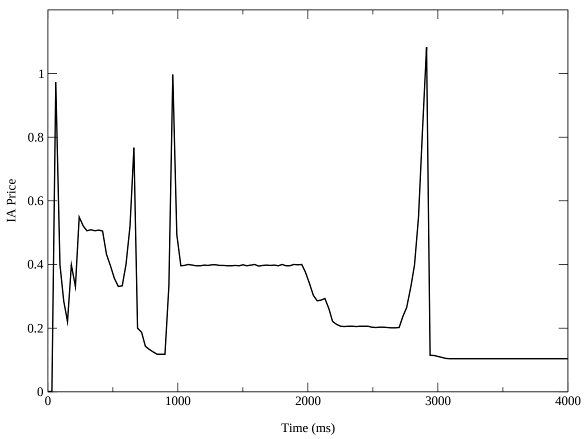
6.3. IA and SA Allocation Algorithms

In this test, we use three mobile notebooks sharing the IA bandwidth in a single cell. Two of the mobile devices (#1 and #2) have identical service purchasing power ϑ_i, while the third mobile device (#3) subscribes to a premium service plan with ϑ_i equal to twice of the ϑ_i for mobile devices #1 and #2.

The first experiment presented in this section is designed to show the behavior of applications using IA allocations. The applications are either bursty in nature such as web browsing or greedy such as FTP download. Figure 7(a) shows the normalized throughput traces of the four mobile



(a) Throughput Normalized by $\vartheta_{i,I}$

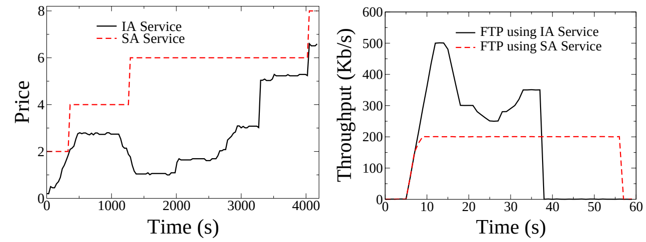


(b) Price

Figure 7. IA Allocation Experiment

devices. Three of the mobile devices have the same service plan, thus the same purchasing power, however, mobile device #2 subscribes to a premium plan giving it a purchasing power that is double that of the other mobile devices. In the experiment, mobile devices #1 and #2 generate web traffic, while mobile device #3 is checking emails and mobile device #4 is downloading using FTP. The throughput growth indicates that starting early has an initial advantage. However, as the traffic enforcement mechanism takes effect, the measured throughput quickly settles to the theoretical allocation values, which are Nash Bargaining Fair, (i.e., mobile device #2 receives twice the throughput as mobile devices #1, #3 and #4 individually receive). In Figure 7(a), because the throughput measurement is normalized by service purchasing power, therefore, between 1000 and 3000 ms, all mobile devices receive the same normalized throughput. Figure 7(b) shows the changes in price for the same experiment. The spikes in price correspond to the traffic surges in throughput measurement. These narrow price spikes also indicate the effectiveness of our pricing mechanism in regulating traffic.

The second experiment is intended to show the interactions between IA and SA traffic and the incentive offered to users to declare truthfully whether their traffic should be carried as IA or SA. Figure 8(a) shows the IA and SA prices with respect to the change in traffic. The figure starts in an



(a) Test A: Price (b) Test B: Throughput

Figure 8. IA/SA Allocation Experiment

initial state where no traffic exists for both classes, then IA traffic is present in the network and drives the price higher. An increase in the SA price takes place as new traffic is generated. Then the IA price drops back as the traffic decreases while SA keeps increasing as more demand is generated. Finally, we can observe an increase in IA traffic as more traffic is generated. We also observe the discrete change in the SA price due to price quantization. We also conducted an experiment trying to compare using SA and IA for a greedy application in order to demonstrate the incentive of using IA for such applications. The trace in Figure 8(b) shows the download of the same file under similar network conditions for IA and SA. Although SA provides a stable allocation, IA proves more advantageous as bursts can occur allowing the download to complete earlier.

6.4. Pricing Dynamics

In this experiment, we use the emulation platform to focus on the pricing dynamics between SA and IA allocations. The cell capacity is set to 1 Mb/s. We simulate 50000 SA service requests according to the Poisson process. The average arrival interval is 5s. The average holding time, (i.e, without early termination) is 15s. The request quantity is uniformly distributed in $[10\ ,\ 100]$ Kb/s. This translate into an average SA load of 15%. The IA traffic activity is generated by activating a random number of mobile devices every second. The random number is uniformly distributed in $[1\ ,\ 20]$. Each user's service purchasing power ϑ_i is randomly assigned from two types: 50 and 100. The SA arrival measurement $\overline{\lambda}(t|p_k)$ is segmented over $K = 20$ quantized price segments, with $p_K = 10$.

The warning interval is $T_S = 20s$ for most of the cases. In addition, the demand estimation parameter $\alpha = 0.7$. In Figure 9, we plot the price pair $\{p_{l,I}, p_{l,S}\}$. We observe that because the SA prices are chosen from a set of quantized price values, SA prices are concentrated at a few values, and hence are relatively stable with respect to changes in the IA prices.

Figure 10(a) illustrates the call drop ratio and early termination ratio for each of the 20 quantized prices. We observe the effect of ranking on the admission success probability and allocation stability. The early termination ratio for small SA valuations is zero because all the calls within those quantized price segments are blocked given the market price $p_{l,S}$. The sharp drop in both the call blocking and

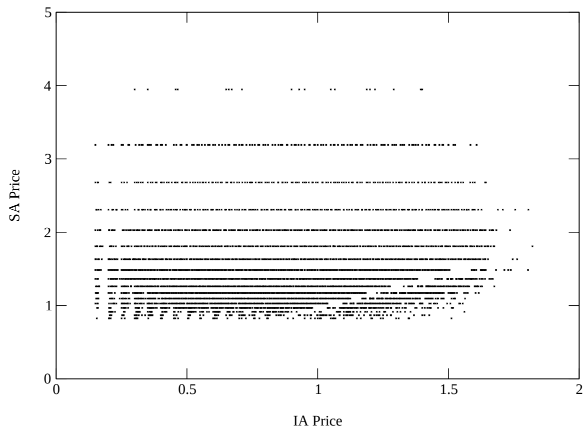
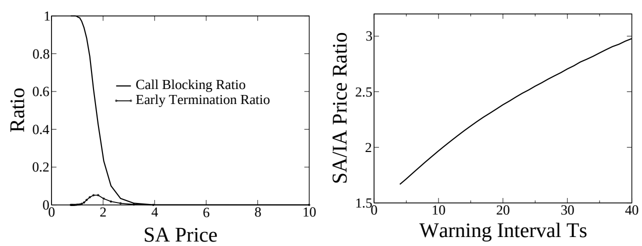


Figure 9. Relation between SA and IA Prices



(a) Allocation Stability Ranking (b) Effect of T_S

Figure 10. SA Service

the early termination ratio for high quantized prices indicate the incentive for SA users not to inflate their bandwidth requests. The effect of the warning interval T_S on the SA price is shown in Figure 10(b). Here we rerun the simulation for different T_S. The results indicate that additional service purchasing power is needed for stabilizing allocations. This value increases with T_S because with a large T_S sessions under probation will have a longer lifetime, and the bandwidth market will have less liquidity.

Our incentive engineering mechanisms do not guarantee bandwidth reservation, to avoid distorting the market price. However, a third party broker may act as an additional source of service purchasing power to sessions under probation, and hence provide guaranteed allocation stability without distorting the incentive mechanisms.

7. Conclusion

In this paper, we provided a solution to the problem of engineering incentives for wireless access services, which offer both higher throughput (IA) for bursty data applications and more stable allocation (SA) for real-time applications. The approach includes the use of service purchasing power and a price-service menu to effectively constrain the user strategy space to a set of cooperative behaviors, leading to fair usage of IA services and truthfully self-differentiation in SA service. The protocol and algorithm design also aims at minimizing protocol overhead on mobile devices. The rate enforcement algorithm controls the downlink traffic tolerating incomplete information caused by the absence of mobile devices' participation in the control algorithm. The reservation based SA service relieves users from declaring session lifetime, and gives early warning of any pending allocation degradation while keeping potential arbitrage between IA and SA services to zero.

References

[1] J. Altmann and K. Chu. A Proposal for a Flexible Service Plan that is Attrative to Users and Internet Service Providers. In *Proc. IEEE INFOCOM*, Alaska, USA, April 2001.

[2] M. Barry, A. Campbell, and A. Veres. Distributed Control Algorithm for Service Differentiation in Wireless Packet Networks. In *Proc. IEEE INFOCOM*, 2001.

[3] V. Bharghavan, K.-W. Lee, S. Lu, S. Ha, J.-R. Li, and D. Dwyer. The TIMELY Adaptive Resource Management Architecture. *IEEE Personal Commun. Mag.*, August 1998.

[4] S. Choi and K. G. Shin. A Comparative Study of Bandwidth Reservation and Admission Control Schemes in QOS-Sensitive Cellular Networks. *ACM Baltzer J. Wireless Networks (WINET)*, 6(4):289–305, 2000.

[5] K. E. Drexler and M. S. Miller. Incentive engineering for computational resource management. In B. Huberman, editor, *The Ecology of Computation*. Elsevier Science Publishers/North-Holland, 1988.

[6] D. Fundenberg and J. Tirole. *Game Theory*. MIT Press, Cambridge, Mass., 1991.

[7] G. Hardin. The Tragedy of the Commons. *Science*, 162:1243–1248, 1968.

[8] P. Hurley, M. Kara, J.-Y. L. Boudec, and P. Thiran. ABE: Providing a Low-Delay Service within Best-Effort . *IEEE Network Mag.*, May/June 2001.

[9] R. Jain, T. Mullen, and R. Hausman. Analysis of Paris Metro Pricing Strategy for QOS with a Single Service Provider. In *Proc. IEEE/IFIP Int'l Workshop on Quality of Service*, Karlsruhe, Germany, June 2001.

[10] F. P. Kelly. Charging and rate control for elastic traffic. *European Trans. Telecommunications*, 8:33–37, 1997.

[11] A. Kuznetsov. Linux Traffic Control (TC). http://www.sparre.dk/pub/linux/tc.

[12] R. R.-F. Liao, R. H. Wouhaybi, and A. T. Campbell. Incentive Engineering in Wireless LAN based Access Networks. Technical report, Dept. of Electrical Engineering, Columbia University, August 2002.

[13] S. H. Low, F. Paganini, and J. C. Doyle. Internet Congestion Control: An Analytical Perspective. *to appear in IEEE Control Systems Magazine*, December 2001.

[14] J. K. MacKie-Mason and H. R. Varian. Pricing congestible network resources. *IEEE Journal on Selected Areas in Communications*, 13(7):1141–1149, September 1995.

[15] J. Nash. The Bargaining Problem. *Econometrica*, 18(2):155–162, April 1950.

[16] A. M. Odlyzko. Paris Metro Pricing for the Internet. In *Proc. ACM Conference on Electronic Commerce (EC'99)*, pages 140–147, 1999.

[17] A. M. Odlyzko. Internet Pricing and the History of Communications. *Computer Networks*, 36:493–517, 2001.

[18] I. P802.11. IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, D2.0, November 1997.

[19] S. Shenker, D. Clark, D. Estrin, and S. Herzog. Pricing in Computer Networks: Reshaping the Research Agenda. *ACM Comput. Commun. Review*, 26(2):19–43, 1996.

[20] H. Yaïche, R. Mazumdar, and C. Rosenberg. A Game Theoretic Framework for Bandwidth Allocation and Pricing in Braodband Networks. *IEEE/ACM Trans. Networking*, 8(5):667–678, October 2000.