Optimal Pricing for Selfish Users and Prefetching in Heterogeneous Wireless Networks

Jonathan Y. Lau and Ben Liang

Department of Electrical and Computer Engineering, University of Toronto
{lauj, liang}@comm.utoronto.ca

***Abstract*— Prefetching has been shown to be an effective technique for reducing resource cost and delay in heterogeneous wireless networks. However, in modern wireless local area networks, there is little centralized management, with no control of upper-level functions such as prefetching, and so users are free to behave selfishly. This work focuses on how pricing can be used to control the suboptimality that results from prefetching and selfish users in heterogeneous wireless networks, and how the perceived cost for the user can be optimized. We derive an analytic model to characterize the optimal network and Nash Equilibrium prefetching strategies. We present a pricing scheme that optimizes the best achievable perceived cost when the network is in a Nash Equilibrium.**

I. Introduction

Speculative prefetching, a technique for predicting and retrieving content before it is actually required, has been shown to be effective in reducing perceived delay in applications where user behavior is predictable. Many aspects of prefetching has been studied for web traffic in traditional homogeneous networks. Work has focused on prefetch prediction strategies [1], [2], and [3], mobile wireless networks [4], [5], [6], [7], and multi-user effects [8].

While prefetching has been studied for many years, it has only been deployed more recently on a large scale. The recent versions of the Mozilla/Firefox web browsers have support for special prefetch tags to instruct the browser to prefetch links [9]. In this implementation, the information used to make prefetching decisions is supplied by the server. The Google search engine now utilizes this prefetching feature to reduce access delay for top search results [10].

Now, with the popularization of wireless networks, many access technologies have emerged, but no single technology is best suited for every application. For example, while cellular access can be made to be ubiquitous, it is expensive and has a limited bandwidth. On the other hand, wireless local area networking technologies (WLAN) such as WiFi offer high bandwidths at low cost, but have limited coverage. In order to create efficient networking solutions, access technologies should be made to cooperate in heterogeneous networks. For this reason, we examine the effects of prefetching for a two-tier heterogeneous network.

In modern wireless network access standards such as WiFi, there tends to be little centralized management, and no support for higher-level functions such as prefetching. Therefore, it is important to consider the network behavior when users base their behaviors on their own best interests.

Recent research has examined the optimal network prefetching strategy for two-tier heterogeneous networks [11]. However, we make the observation that the optimal prefetching strategy for a network is not the same as the optimal strategy for an individual user [12]. For example, if all users use a network optimal prefetching strategy with the exception of one defecting user, the defecting user can reduce its cost by increasing its level of prefetching. However, by doing so, the defecting user increases the network load, and so increases costs for all other users in the network. In a classical result, Nash [13] shows that a symmetric equilibrium exists for any finite multi-player game. Hence, if all users are homogeneous and behave selfishly in selecting prefetching strategies, the resulting prefetching strategy is at Nash Equilibrium, and generally suboptimal.

In this paper, we derive an analytic model to calculate and compare the network optimal prefetching strategy and the Nash Equilibrium stategy. The main contribution of this work is the optimization of the pricing scheme such that the best achievable perceived cost, from the perspective of the user, is minimized, while the network is stable, or at Nash Equilibrium.

We begin by describing our system model in Section II. In Section III, we characterize the expected perceived cost, and compare the optimal and stable prefetching strategies. Section IV discusses how pricing can be optimized. Finally, we conclude with Section V.

II. System Model

We first describe the mobility, network, and traffic models that will be used for this discussion.

A. Mobility Model

We consider a two-tier wireless network consisting of small high-bandwidth hot spots (WLAN) using a technology such as WiFi, scattered throughout a ubiquitous wireless network (CELL) using a technology such as cellular access. Mobile users roam throughout the CELL sporadically entering and leaving WLANs, as shown in Figure 1. Since we are assuming that WLANs are much faster and cheaper to use than the CELL, a user will use a WLAN exclusively when the user has access to the WLAN. Therefore, we can model the mobility of

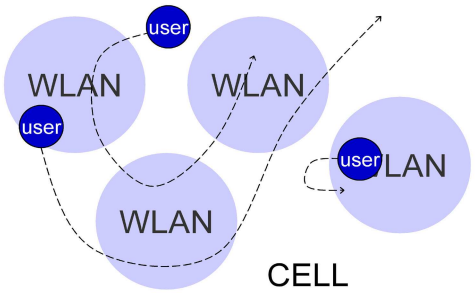

Fig. 1. Mobility in a two-tier heterogeneous network

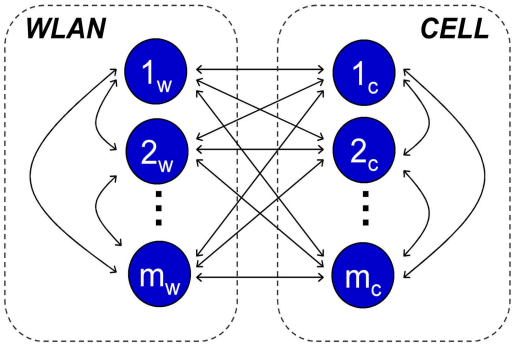

Fig. 2. Mobility as a Markov process

a user using a two-state alternating renewal process ($WLAN$ and $CELL$), where the user is in either a WLAN or the CELL. It is well known that a phase-type (PH) distribution can be used to model any positive distribution [14], and so we use PH random variables to represent the residence time of a user in the $WLAN$ and $CELL$ states. Now, since each PH process can be associated with a Markov process, let m_w and m_c represent the number of phases in the respective $WLAN$ and $CELL$ Markov processes, as shown in Figure 2. The Markov process can be described by an infinitessimal generator matrix

$$\mathbf{A} = \begin{bmatrix} -\mathbf{Q}_w & \mathbf{t}_w\mathbf{a}_c \\ \mathbf{t}_c\mathbf{a}_w & -\mathbf{Q}_c \end{bmatrix}, \quad (1)$$

where $\mathbf{a}_w$, of size $(1 \times m_w)$, are the initial $WLAN$ state probabilities, $-\mathbf{Q}_w$, of size $(m_w \times m_w)$, contains the transition rates between $WLAN$ states, and $\mathbf{t}_w = \mathbf{Q}_w\mathbf{1}$, of size $(m_w \times 1)$, are the absorption rates of $WLAN$ states. Likewise, we have m_c, $\mathbf{a}_c$, $-\mathbf{Q}_c$, and $\mathbf{t}_c$ for $CELL$.

To model user mobility in practical two-tier systems, the entries in (1) can be estimated based on historical data collected by the service provider. The reference [15] provides the implementation and analytical details of such an example.

B. Network Model

Since most applications that can use prefetching, such as web surfing, news forum browsing, or database access, have traffic patterns that are highly asymmetrical, we only consider downlink traffic. We assume that both the base stations and wireless access points in the CELL and WLAN are connected to a high speed backbone network. Thus, the traffic is only limited by the wireless access media. In the CELL, users are given dedicated downlink access channels with a fixed bandwidth β_c. In each WLAN, however, users share a single wireless access point with a fixed bandwidth β_w, where a FIFO queue is formed for downlink traffic[1]. The per-byte prices to use the CELL and WLAN are α_c and α_w respectively.

Next, we first define two costs. *Resource cost*, related to α_w and α_c is the per-byte cost that the service provider charges the user for bandwidth utilization. *Delay cost* is the value of time associated with the delay incurred while a user waits for a requested document to download. When combined together, we obtain the *perceived cost*. Existing literature has shown that a user's perceived value of time can be estimated using the user's level of income [17]. We will denote the user's value of time α_t in dollars per unit time. Users base their behavior on the perceived cost.

C. Traffic Model

Each user generates traffic following a Poisson process with rate λ_r. We assume that with each document that a user accesses, there are K other documents that the user will access next with significant probability[2]. So, with each document that user n downloads, the user's device immediately attempts to prefetch, in the background, the k_n documents that will be accessed next with highest probability. We call k_n the *prefetching strategy* for user n, and we manipulate k_n to control the level of prefetching. Each document has an associated access probability that is provided by the document's source [2], [3], [6], [7], [16]. Clearly, when a user prefetches k_n documents, those documents will be the k_n documents that the user will most probably access next. Hence, when k_n documents are prefetched, the probability that the user will access a prefetched document is a cumulative distribution function $F_p(k_n)$. For convenience, we denote $F_{np}(k_n) = 1 - F_p(k_n)$.

Since the download times of documents is typically small, we assume that a batch of prefetched documents is successfully received when the last document of the batch arrives. Furthermore, we make the assumption of exponentially distributed document sizes for analytical tractability. However, we later show by simulation that the distribution of the document sizes has little effect on the results.

III. Optimal and Stable Prefetching Strategies

As mentioned earlier, prefetching can be optimized for a heterogeneous network to reduce resource cost and delay for

[1]Although it is possible to consider a prioritized queue in the WLAN, given that we are studying selfish users, users would classify all traffic as high-priority, unless a differential pricing scheme is used. However, current wireless services do not differentiate pricing based on traffic class, and in practice it is difficult to implement.

[2]While we do not assume any specific application, caching is implemented on some applications such as web browsing. When documents are cached and users repeatedly access the same document, the rate at which users request documents is effectively decreased. While it is possible to modify our model to accommodate caching by scaling document access rates, the details are application dependent and thus are beyond the scope of this paper.

the user. However, the optimal prefetching strategy is generally not a Nash Equilibrium and therefore selfish users would not choose to behave optimally. In this section, we first give an introduction to the notion of the optimal and stable strategies, followed by the calculation of the expected costs used to determine these strategies, and then we discuss the differences between these strategies.

A. Overview of the Optimal and Stable Strategies

Let $\mathbf{k} = (k_1, \ldots, k_N)$ be the prefetching strategies of users $1, \ldots, N$. Let $c^{(n)}(\mathbf{k})$ be the perceived cost of a document for user n in the WLAN. The perceived cost is a weighted combination of the resource cost and delay cost. Now, given the expression for the expected perceived cost, it is not difficult to determine the network optimal prefetching strategy. If we assume fairness in that all users use the same strategy, we set $k_1 = k_2 = \ldots = k_N = k$. Substituting into $c^{(n)}(\mathbf{k})$, we obtain a single variable function $c^{(n)}(k)$ which can be easily optimized numerically to obtain the optimal prefetching strategy k^*.

Next, suppose $N-1$ users are using the prefetching strategy k_A, and one user is using k_B. That is, say $k_1 = \ldots = k_{N-1} = k_A$ and $k_N = k_B$. Substituting, we obtain a two-variable function $c^{(n)}(k_A, k_B)$. Suppose we fix $k_A = k^*$ and optimize $c^{(n)}(k^*, k_B)$ over k_B. We would see that in general $k_B \neq k^*$, meaning that k^* is not a Nash Equilibrium. That is, user N can gain by prefetching a different amount. To find the Nash Equilibrium, where given the strategy of all users, no single user would choose to behave otherwise, we solve the system of equations

$$\begin{cases} 0 = \frac{\partial c^{(n)}(k_A, k_B)}{\partial k_B} \\ k_A = k_B \end{cases} . \tag{2}$$

We will call the Nash Equilibrium for the perceived cost, k^s, the *stable prefetching strategy*.

Let $c_r^{(n)}(\mathbf{k})$ and $c_d^{(n)}(\mathbf{k})$ be expressions for the expected resource cost and delay. Likewise, we can also find the respective optimal strategies k_r^* and k_d^*, and the stable strategies k_r^s and k_d^s when resource cost alone or delay alone is considered.

B. Analysis of the Expected Costs

We assume that since both the price and bandwidth of the CELL is significantly higher than that of the WLAN, a user would only prefetch in the WLAN. Thus, we calculate the cost to request the next document at the time when a user makes a decision on the number of documents to prefetch in the WLAN.

The costs associated with downloading a document depend on whether the document was previously successfully prefetched, and if not, the current network that the user is in. So, the costs depend on the relationships between T_S, the sojourn time of the WLAN queue, T_r, the request interarrival time, and T_w, the residual residence time of the user in the WLAN. We can calculate the expected resource cost and delay cost by considering the five possible cases, enumerated by l.

TABLE I
INTEGRATION LIMITS

Case l	$t_{w,1}^{(l)}$	$t_{w,2}^{(l)}$	$t_{r,1}^{(l)}$	$t_{r,2}^{(l)}$
$T_S < T_r < T_w$	t_S	t_w	t_S	∞
$T_S < T_w < T_r$	t_S	∞	t_w	∞
$T_r < T_S < T_w$	t_S	∞	0	t_S
$T_r < T_w < T_S$	0	t_S	0	t_w
$T_w < T_S, T_r$	0	t_S	t_w	∞

If we consider integrals of the form

$$\int_0^\infty \int_{t_{w,1}^{(l)}}^{t_{w,2}^{(l)}} \int_{t_{r,1}^{(l)}}^{t_{r,2}^{(l)}} (\cdot) dt_r dt_w dt_S \tag{3}$$

then the integration limits for t_r and t_w are as shown in Table I for each case.

Before we can discuss the costs, we must first describe the behavior of the WLAN queue. Suppose $\mu = \frac{\beta_w}{W}$ is the service rate for a single document in the WLAN. When user n accesses a document, this document may or may not have been prefetched. If the document was previously prefetched, then the user only requests k_n documents as prefetches in preparation for the next document access. If, however, the document was not prefetched, then the user must request k_n+1 documents. To simplify the analysis, we assume that the user always makes requests for k_n documents. This assumption is valid because when k_n is small, the queue is rarely occupied, so k_n has little effect. When k_n is large, however, k_n and $k_n + 1$ are similar. This assumption is shown to be valid by simulation in Section IV.

To calculate the distribution of the WLAN sojourn time, we use the well known Pollaczek-Khinchin formula for the Laplace transform of the sojourn time density of an $M/G/1$ queue [18]

$$f_{T_S}^*(\mathbf{k}; s) = \frac{s(1 - \rho(\mathbf{k}))f_X^*(\mathbf{k}; s)}{s - \lambda + \lambda f_X^*(\mathbf{k}; s)} \tag{4}$$

where $\rho(\mathbf{k}) = \frac{\lambda_r}{\mu} \sum_{n=1}^N k_n$ is the utilization of the WLAN, $f_X^*(\mathbf{k}; s)$ is the Laplace transform of the density for a batch of documents, and $\lambda = N\lambda_r$ is the aggregated WLAN traffic.

Since we are assuming that documents are exponentially distributed, the service time X_n for a batch of k_n documents for user n is Erlang distributed with Laplace transformed density function

$$f_{X_n}^*(k_n; s) = \left(\frac{\mu}{s + \mu}\right)^{k_n} . \tag{5}$$

When there are N identical and independent (iid) users in the WLAN, the arrival process at the WLAN, an aggregation of iid Poisson processes, is also Poisson, and the Laplace transform of the density of general service time X for a batch of documents is

$$f_X^*(\mathbf{k}; s) = \frac{1}{N} \sum_{n=1}^N \left(\frac{\mu}{s + \mu}\right)^{k_n} . \tag{6}$$

We must also first calculate the residual residence time of a user in a WLAN. An important property of PH renewal processes is that the residual time of a PH renewal process is a PH random variable. In fact, we can calculate the density function of the WLAN residual residence time by $f_{T_w}(t_w) = \mathbf{q}\, e^{-\mathbf{Q}_w t_w}\, \mathbf{Q}_w\, \mathbf{1}$, where $\mathbf{q} = (\mathbf{a}_w\, \mathbf{Q}_w^{-1}\, \mathbf{1})^{-1} \mathbf{a}_w\, \mathbf{Q}_w^{-1}$ [14]. Let $\mathbf{Q}_w$ be diagonalized as $\mathbf{Q}_w = \mathbf{V} diag\{\nu_j\} \mathbf{V}^{-1}$, where ν_j are the eigenvalues of $\mathbf{Q}_w$ and $\mathbf{V}$ contains the eigenvectors of $\mathbf{Q}_w$. The WLAN residual residence time is therefore

$$f_{T_w}(t_w) = \mathbf{q}\, \mathbf{V}\, diag\{\nu_j\, e^{-\nu_j t_w}\}\, \mathbf{V}^{-1}\, \mathbf{1}. \tag{7}$$

We next describe the costs involved in each case.

1) Case 1: $T_S < T_r < T_w$: In this case, the prefetched documents arrive before the next request and before the user leaves the WLAN. If the next document that the user requests was prefetched, then there is no additional resource cost, and the delay is zero. If, however, the next document that the user requests was not prefetched, which occurs with probability $F_{np}(k_n)$, then the user must make a request on the WLAN, which will incur an additional resource cost of α_w. To find the expected time to service a request for a *single* document in the WLAN, we first find the expected waiting time using

$$\frac{\lambda E[X^2]}{2(1-\rho(\mathbf{k}))} \tag{8}$$

where $E[X^2] = \frac{1}{N}\sum_{n=1}^{N} \frac{k_n + k_n^2}{\mu^2}$ [18], is the second raw moment of the Erlang distributed batch service time. Therefore, we obtain the expected delay for a single document

$$E[T_S^1] = \frac{1}{\mu} + \sum_{n=1}^{N} \frac{k_n + k_n^2}{\mu^2}. \tag{9}$$

Note that this is different from the sojourn time for a batch of documents.

2) Case 2: $T_S < T_w < T_r$: In this case, the prefetched documents arrive before the next request, but the user's next request arrives after the user has left the WLAN. Hence, if a document was not prefetched, which occurs with probability $F_{np}(k_n)$, the user must make a request $T_r - T_w$ seconds after it enters the CELL. When the user makes the next request, the user may be in a WLAN or in a CELL. We can determine the probabilities of being in either a WLAN or the CELL after t seconds using the matrix exponential $\mathbf{P}(t) = e^{\mathbf{A}t}$ [19]. Assuming that $\mathbf{A}$ is diagonalizable with eigenvalues σ_i, we can write

$$\mathbf{P}(t) = \mathbf{U} diag\{e^{-\sigma_i t}\} \mathbf{U}^{-1} = \sum_{i=1}^{M} \mathbf{P}^{(i)} e^{-\sigma_i t} \tag{10}$$

where $\mathbf{U}$ contains the eigenvectors of $\mathbf{A}$, $\mathbf{P}^{(i)} = \mathbf{u}_i \mathbf{u}_i'$, and $M = m_c + m_w$. So, given that a user has just entered the CELL, the probability $p_{cw}(t)$ that the user is in a WLAN t seconds later is given by

$$p_{cw}(t) = \sum_{i=1}^{M} a_{c,i} \left(\sum_{j=1}^{m_w} P_{ij}^{(i)} \right) e^{-\sigma_i t} = \sum_{i=1}^{M} \omega_i^{(cw)} e^{-\sigma_i t} \tag{11}$$

where $P_{ij}^{(i)}$ is the ij^{th} entry of $\mathbf{P}^{(i)}$, $\omega_i^{(cw)}$ are constants, and $a_{c,i}$ are the elements of $\mathbf{a}_c$. Likewise, we can compute $\omega_i^{(cc)}$ and $p_{cc}(t)$ for the probability that the user is in a CELL. Therefore, given that the user has just entered the CELL, the resource cost for a request t seconds later is

$$r_{RQ}(t) = p_{cw}(t)\alpha_w + p_{cc}(t)\alpha_c \tag{12}$$

and the delay is

$$d_{RQ}(t) = p_{cw}(t) E[T_S^1] + p_{cc}(t) D, \tag{13}$$

where $D = \frac{W}{\beta_c}$ is the expected download time in the CELL. We weight the contribution to the expected costs by the probability that a request is made by $F_{np}(k_n)$.

3) Case 3: $T_r < T_S < T_w$: In this case, the user makes the next request before the prefetched documents arrive on the WLAN, so the user makes a new request on the WLAN. Since we expect that T_r is typically much larger than T_S, the occurrence of $T_r < T_S$ means that either T_r is very small, or T_S is very large, and the WLAN queue is backlogged. There is a strong correlation between the sojourn time of the prefetched documents and that of the newly requested document. To simplify the analysis, we assume that the two sojourn times are the same. Thus, the resource cost is α_w and the delay is T_S.

4) Case 4: $T_r < T_w < T_S$: In this case, the user makes the next request before the prefetched documents arrive on the WLAN, so the user makes a new request on the WLAN. However, the user leaves the WLAN before T_S, the arrival time of the original prefetch, so it is impossible for the request to arrive on the WLAN in time. Hence, $T_w - T_r$ seconds after the request, the user leaves the WLAN and makes a request on the CELL. The resulting resource cost for both the WLAN and the CELL is $\alpha_w + \alpha_c$ and the delay cost is $T_w - T_r + D$.

5) Case 5: $T_w < T_S, T_r$: In this case, the user leaves the WLAN before the arrival of the prefetched documents and the next request. When the user does make the next request, the user could be in a WLAN or a CELL. This case is similar to Case 2, with the exception that $r_{RQ}(t)$ and $d_{RQ}(t)$ are not weighted by $F_{np}(k_n)$.

The integrands and integration limits from the five cases are summarized in Table II and Table I respectively. Using the integrands $r^{(l)}(t_r, t_w, t_S)$ We can obtain the resource cost by

$$c_r^{(n)}(\mathbf{k}) = \sum_{l=1}^{5} \int_0^{\infty} \int_{t_{w,1}^{(l)}}^{t_{w,2}^{(l)}} \int_{t_{r,1}^{(l)}}^{t_{r,2}^{(l)}} r^{(l)}(t_r, t_w, t_S) dt_r dt_w dt_S. \tag{14}$$

Likewise, we can obtain the delay cost $c_d^{(n)}(\mathbf{k})$ using the integrands $d^{(l)}(t_r, t_w, t_S)$. The resulting expected resource cost is

$$c_r^{(n)}(\mathbf{k}) = \mathbf{q}\mathbf{V} diag \left\{ \zeta_j \left(1 - F_p(k_n) f_{T_S}^*(\mathbf{k}; \nu_j + \lambda_r) \right) + \frac{\alpha_c \lambda_r}{\nu_j + \lambda_r} \right.$$
$$\left. - \alpha_c f_{T_S}^*(\mathbf{k}; \nu_j) + \frac{\alpha_c \nu_j}{\nu_j + \lambda_r} f_{T_S}^*(\mathbf{k}; \nu_j + \lambda_r) \right\} \mathbf{V}^{-1} \mathbf{1} + \alpha_w k_n \tag{15}$$

TABLE II

INTEGRANDS

Case l	$r^{(l)}(t_r, t_w, t_S)$	$d^{(l)}(t_r, t_w, t_S)$
$T_S < T_r < T_w$	$F_{np}(k_i)\alpha_w$	$F_{np}(k_i)E[T_S^1]$
$T_S < T_w < T_r$	$F_{np}(k_i)r_{RQ}(T_r - T_w)$	$F_{np}(k_i)d_{RQ}(T_r - T_w)$
$T_r < T_S < T_w$	α_w	T_S
$T_r < T_w < T_S$	$(\alpha_w + \alpha_c)$	$\alpha_t(t_r - t_w + D)$
$T_w < T_S, T_r$	$r_{RQ}(T_r - T_w)$	$d_{RQ}(T_r - T_w)$

and the expected delay cost is

$$c_d^{(n)}(\mathbf{k}) = \mathbf{q}\mathbf{V} diag \left\{ \frac{F_{np}(k_n)E[T_S^1]\lambda_r}{\lambda_r + \lambda_w} f^*_{T_S}(\mathbf{k}; \lambda_r + \lambda_w) - \left. \frac{\partial f^*_{T_S}(\mathbf{k}; s)}{\partial s} \right|_{s=\nu_j} \right.$$
$$\left. + \left. \frac{\partial f^*_{T_S}(\mathbf{k}; s)}{\partial s} \right|_{s=\nu_j+\lambda_r} + \left(- \frac{\nu_j}{(\nu_j + \lambda_r)\lambda_r} + \frac{D\nu_j}{\nu_j + \lambda_r} \right) f^*_{T_S}(\mathbf{k}; \lambda_r + \nu_j) + \right.$$
$$\left. + F_{np}(k_n) \sum_{i=1}^{M} \delta_i \frac{\lambda_r \nu_j}{(\sigma_i + \lambda_r)(\nu_j + \lambda_r)} f^*_{T_S}(\mathbf{k}; \nu_j + \lambda_r) + \left. \frac{\partial f^*_{T_S}(\mathbf{k}; s)}{\partial s} \right|_{s=\nu_j} \right.$$
$$+ \left(\frac{\nu_j}{\lambda_r(\lambda_r + \nu_j)} - \frac{\lambda_r}{\nu_j(\lambda_r + \nu_j)} - D \right) f^*_{T_S}(\mathbf{k}; \nu_j) + \left(\frac{\lambda_r D}{\nu_j + \lambda_r} + \frac{\lambda_r}{\nu_j(\nu_j + \lambda_r)} \right)$$
$$\left. + \sum_{i=1}^{M} \delta_i \frac{\lambda_r \nu_j}{(\sigma_i + \lambda_r)(\nu_j + \lambda_r)} (1 - f^*_{T_S}(\mathbf{k}; \nu_j + \lambda_r)) \right\} \mathbf{V}^{-1}\mathbf{1}. \quad (16)$$

We obtain the perceived cost by combining the per-byte resource cost and the delay using W, the expected document size, and α_t, the value of a user's time. So, the perceived cost is

$$c^{(n)}(\mathbf{k}) = W c_r^{(n)}(\mathbf{k}) + \alpha_t c_d^{(n)}(\mathbf{k}). \quad (17)$$

C. Validation

The accuracy of the analytic model was verified by simulation. The Java-based simulator that we implemented modelled user mobility between the WLAN and CELL networks and real simulation of the WLAN queue.

We know that web document access frequency follows a Zipf-like distribution, and that there is a weak correlation between document size and access frequency [20]. Therefore, for the remainder of this discussion we use a specific Zipf-like cumulative distribution function

$$F_p(k_n) = \begin{cases} \frac{\ln(k_n+1)}{\ln(K+1)}(1-\epsilon) & 0 < k_n < K \\ 1 - \epsilon & k_n \geq K \end{cases} \quad (18)$$

where ϵ, assumed to be very small, is the probability that the user does not choose any of the K documents. We point out that the above analysis holds regardless of the distribution used.

Figure 3 shows the analytical and simulated perceived cost for different document size distributions, plotted with 95% confidence intervals. Each data point shown represents the results from ten simulation runs. All of the mobility models used Erlang-distributed residence times in the WLAN and CELL, with mean residence times of $t_{res,w}$ and $t_{res,c}$ respectively. The parameters used were $\alpha_w = \$0.1/MB$, $\alpha_c = \$0.01/KB$, $\alpha_t = \$20/h$, $\beta_w = 10Mbps$, $\beta_c = 100kbps$, $W = 50KB$, $m_w = 4$, $m_c = 4$, $t_{res,w} = 60s$, $t_{res,c} = 60s$, $\lambda_r = 1/20s^{-1}$,

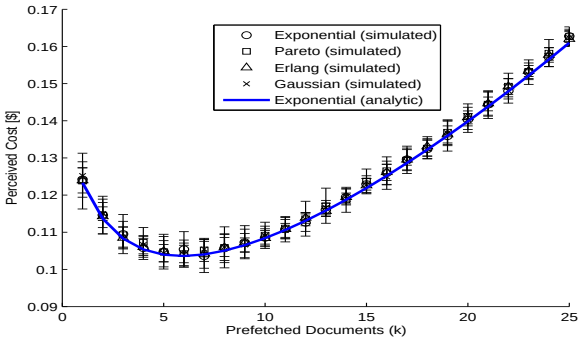

Fig. 3. The effect of document size distribution on perceived cost

$K = 50$, and $N = 10$. Unless otherwise specified, these parameters are used in later discussions. In the analytic model, we had assumed exponentially distributed document sizes. However, as the simulation results show in Figure 3, the resulting perceived cost does not change significantly even document sizes have Erlang, truncated Pareto, or Gaussian distributions. Likely, this is because when few documents are prefetched, there is little backlog in the WLAN, so the delay caused by queueing is insignificant and document size distribution has little effect. On the other hand, by the Central Limit Theroem we see that when many documents are prefetched, the distribution of the total size of a batch of documents approaches Gaussian in shape, regardless of distribution.

D. Discussion of the Stable and Optimal Strategies

We first discuss the resource cost and delay separately to gain insight into how resource cost and delay affect the perceived cost. Figure 4 shows how the optimal and stable prefetching strategies, k_r^* and k_r^s respectively, are affected *when resource cost alone* is considered. When the WLAN is very cheap with respect to the CELL, $\frac{\alpha_w}{\alpha_c}$ is very small, and each user can reduce its own resource cost by prefetching many documents. However, in doing so, each user increases the resource cost for all other users. For all users in the network to minimize resource cost for everyone, the optimal prefetching strategy k_r^*, as shown in Figure 4, is to prefetch far less than that the stable strategy k_r^s. When the ratio $\frac{\alpha_w}{\alpha_c}$ is higher, the optimal and stable strategies are very close. This suggests that if only resource cost is considered, the pricing ratio $\frac{\alpha_w}{\alpha_c}$ should not be too small.

Figure 5 shows how the number of users affects the optimal and stable prefetching strategies, k_d^* and k_d^s respectively, *when delay alone* is considered. For small numbers of users, the stable strategy is to prefetch as many documents as possible (in this case $K = 50$, the maximum number of documents), which is significantly greater than the optimal strategy. This problem, when delay alone is considered, arises when a flat-rate pricing scheme is used. This result suggests that a flat-rate pricing scheme results in significant suboptimality when users are selfish.

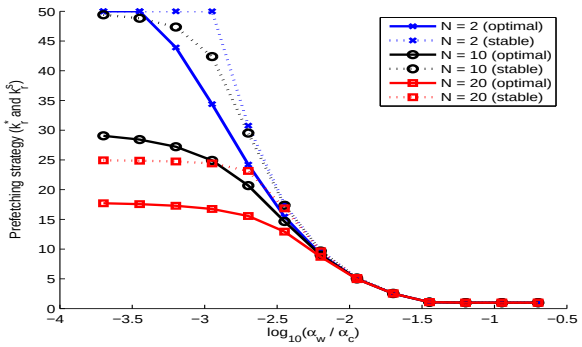

Fig. 4. The effects of $\frac{\alpha_w}{\alpha_c}$ on prefetching, considering *resource cost only*

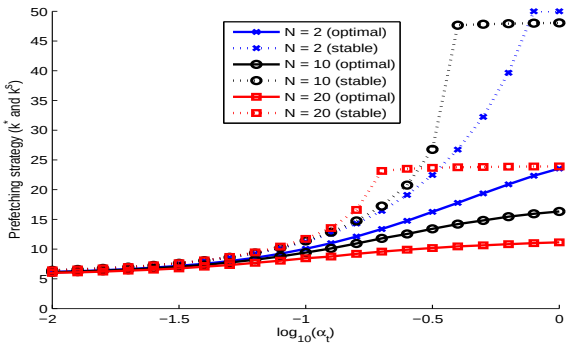

Fig. 6. The effects of α_t on prefetching, considering perceived cost

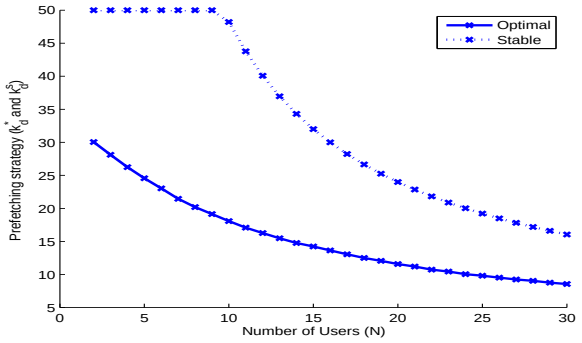

Fig. 5. The effects of the number of WLAN users on prefetching, considering *delay only*

Figure 6 show the plots of the stable and network optimal prefetching strategies, k^* and k^s respectively, when resource cost and delay are combined into perceived cost. In this figure, as the value of time α_t is increased, the tendency for suboptimality increases. For small α_t, the behavior of perceived cost follows that of resource cost, but for large α_t, the behavior of perceived cost follows that of delay. In practice, α_t tends to be small (e.g. \$20 per hour is equivalent to \$0.0056 per second), and so resource cost typically dominates the perceived cost.

IV. Optimal Pricing

In the following section, we discuss how the pricing ratio in a hybrid pricing scheme can be optimized.

A. Optimizing the Pricing Ratio

Suppose we consider a scenario where α_t is fixed at \$20 per hour. Using the perceived cost, we can calculate that the stable prefetching strategy is $k^s = 6.1885$, which is the actual strategy that selfish users would use. However, the optimal resource cost strategy is $k_r^* = 5.6130$, meaning that the users are paying more than necessary, and the optimal delay strategy is $k_d^* = 11.5736$, meaning that the users are also waiting longer than necessary for requests.

In practice, it is easy to manipulate the pricing but it is difficult to change the bandwidth given the hardware in the network. Therefore, the goal is to manipulate the pricing such that the stable prefetching strategy of the perceived cost coincides with the optimal strategies of the resource cost and delay. In this way, the system performs optimally, and no user would opt for a different prefetching strategy.

We propose a hybrid pricing scheme where the service provider charges a fixed monthly fee α_F in addition to the per-byte costs α_w and α_c. The fee α_F is set by the service provider to make the wireless network profitable, and does not affect the prefetching strategies of the users. In the extreme case when α_w and α_c are zero, the network uses flat-rate pricing, where delay alone is considered for prefetching decisions. As we saw in Figure 5, in this case the network can become highly suboptimal.

We assume that α_F is reasonably set by the service provider, and thus does not affect the user's per-document perceived cost. We assume that all users subscribe to the network regardless of α_F, since issues of network participation are beyond the scope of this discussion.

Since the expected cost expressions are linear with respect to α_w and α_c, we are only interested in the *ratio* between these two costs. Therefore, we fix α_c and vary α_w. We expect that as $\frac{\alpha_w}{\alpha_c}$ decreases, there is increased benefits to prefetching. Now, when α_w is decreased and α_c is fixed, the service provider would increase α_F to recuperate lost profits. The goal of this optimization is to manage the suboptimal behavior of selfish users such that users can achieve the best possible perceived cost by manipulating the ratio $\frac{\alpha_w}{\alpha_c}$. The exact values of α_w, α_c, and α_F depend on the perceived utility of data and the cost for the service provider to run a wireless network, and is beyond the scope of this paper.

Figure 7 shows the stable and optimal strategies as the pricing ratio is changed. We see that as $\frac{\alpha_w}{\alpha_c}$ changes, not only does the degree of suboptimality change, but we can also manipulate k^s.

We are interested in finding the *best achievable perceived cost* for selfish users. That is, the perceived cost when users all use the stable prefetching strategy k^s. Fixing α_c, we can find the optimal WLAN price α_w^* by solving

$$\alpha_w^* = arg\ min_{\alpha_w} \left\{ c^{(n)}(k^s(\alpha_w); \alpha_w) \right\} \qquad (19)$$

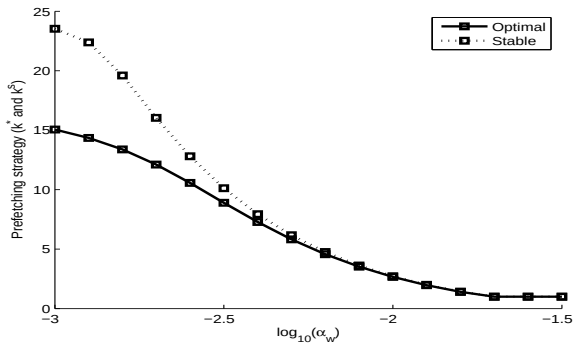

Fig. 7. The effects of the pricing ratio on prefetching, considering perceived cost

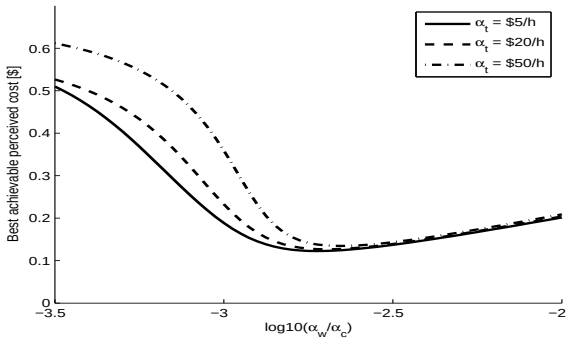

Fig. 8. Optimization of perceived cost

where $k^s(\alpha_w)$ is the solution to (2) using α_w. While intuitively one would expect the best achievable perceived cost to always decrease as α_w decreases, it is not the case, as shown in Figure 8. From this figure, we can see that to a certain degree, reducing α_w while fixing α_c decreases the perceived cost for the user. However, as we continue to decrease α_w, the best achievable perceived cost increases rapidly. This is likely because a lower $\frac{\alpha_w}{\alpha_c}$ encourages more prefetching. When users prefetch too many documents, the WLAN becomes too heavily loaded and the WLAN queueing delay increases, prefetching becomes less effective for all users, and so the best achievable perceived cost increases.

Figure 8 also shows the effect of α_t, the value of time, on the best achievable perceived cost. As the value of time increases, the weight of delay cost increases. Since the prefetching strategy when only delay is considered is significantly higher than that when only resource cost is considered, more prefetching is encouraged. However, when only delay is considered, the suboptimality is significantly greater, and so the best achievable perceived cost is increased.

V. Conclusions

Speculative prefetching has been shown to be an effective technique for reducing resource cost and delay in heterogeneous wireless networks. In modern WLANs, there is little centralized management, so it is important that we find methods to control the effects of selfish users.

In this paper, we studied the optimal pricing for a two-tier heterogeneous network with prefetching and selfish users. Using an analytic model to quantify the expected perceive cost associated with the number of documents a user prefetches, we demonstrated the effects of variables such as pricing, the number of WLAN users, and the value of time on the stable and optimal prefetching strategies. We showed that the pricing ratio can be manipulated to optimize the best achievable perceived cost for users, such that the network is in a Nash Equilibrium. Finally, there are many mechanisms such as trust and courtesy that govern human interaction. The assumption that users behave only in their own best interest provides a worse case analysis of selfish behavior.

References

[1] Z. Jiang and L. Kleinrock, "An adaptive network prefetch scheme," *Journal on Selected Areas in Communications*, vol. 16, no. 3, pp. 358–368, 1998.

[2] M. Angermann, "Analysis of speculative prefetching," *ACM SIGMOBILE*, vol. 6, no. 2, pp. 13–17, 2002.

[3] V. N. Padmanabhan and J. C. Mogul, "Using predictive prefetching to improve world wide web latency," *SIGCOMM Computer Communications Review*, vol. 26, pp. 1193–1205, 1996.

[4] S. Gitzenis and N. Bambos, "Power-controlled data prefetching/caching in wireless packet networks," *INFOCOM*, vol. 3, pp. 1405–1414, 2002.

[5] Z. Jiang and L. Kleinrock, "Web prefetching in a mobile environment," *IEEE Personal Communications*, vol. 5, no. 5, pp. 25–34, 1998.

[6] L. Yin and G. Cao, "Adaptive power-aware prefetch in wireless networks," *Transactions on Wireless Communications*, vol. 3, no. 5, pp. 1648–1658, 2004.

[7] N. Tuah, M. Kumar, and S. Venkatesh, "Resource-aware speculative prefetching in wireless networks," *Wireless Networks*, vol. 9, pp. 61–72, 2003.

[8] M. Crovella and P. Barford, "The network effects of prefetching," *INFOCOM*, vol. 3, pp. 1232–1239, 1998.

[9] "http://www.mozilla.org/projects/netlib/."

[10] "http://www.google.com/help/features.html."

[11] S. Drew and B. Liang, "Mobility-aware web prefetching over heterogeneous wireless networks," *Proceedings of the 15th IEEE PIMRC*, pp. 687–691, 2004.

[12] B. Liang, S. Drew, and D. Wang, "Performance of multiuser network-aware prefetching in heterogeneous wireless systems," *in press, to appear in ACM/Springer Wireless Networks*, 2007.

[13] J. Nash, "Non-cooperative games," *Annals of Mathematics*, vol. 54, no. 2, pp. 286–295, 1951.

[14] G. Latouche and V. Ramaswami, *Introduction to matrix analytic methods in stochastic modeling*. SIAM, 1999.

[15] A. H. Zahran, B. Liang, and A. Saleh, "Modeling and performance analysis of beyond 3g integrated wireless networks," *Proceedings of the IEEE International Conference on Communications (ICC)*, 2006.

[16] A. Bestavros, "Speculative data dissemination and service to reduce nserver load, network traffic and service time for distributed information systems," *Proceedings of the International Conference on Data Engineering*, pp. 180–187, 1996.

[17] R. T. Deacon and J. Sonstelie, "Rationing by waiting and the value of time: Results from a natural experiment," *Journal of Political Economy*, pp. 627–647, 1985.

[18] D. Gross and C. Harris, *Fundamentals of Queueing Theory*, 3rd ed. John Wiley & Sons, Inc., 1998.

[19] A. Papoulis and S. U. Pillai, *Probability, random variables, and stochastic processes*, 4th ed. McGraw Hill, 2002.

[20] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: evidence and implications," *INFOCOM*, vol. 1, pp. 126–134, 1999.