User-Level QoS and Traffic Engineering for 3G Wireless 1xEV-DO Systems

Sem Borst, Ken Clarkson, John Graybeal, Harish Viswanathan, and Phil Whiting

Bell Laboratories, Lucent Technologies
600 Mountain Avenue, Murray Hill, NJ 07974

Abstract -- **3G wireless systems such as 3G-1X, 1xEV-DO and 1xEV-DV provide support for a variety of high-speed data applications. The success of these services critically relies on the capability to ensure an adequate QoS experience to users at an affordable price. With wireless bandwidth at a premium, traffic engineering and network planning play a vital role in addressing these challenges. We present models and techniques that we have developed for quantifying the QoS perception of 1xEV-DO users generating FTP or Web browsing sessions. We show how user-level QoS measures may be evaluated by means of a Processor-Sharing model which explicitly accounts for the throughput gains from multi-user scheduling. The model provides simple analytical formulas for key performance metrics such as response times, blocking probabilities and throughput. Analytical models are especially useful for network deployment and in-service tuning purposes due to the intrinsic difficulties associated with simulation-based optimization approaches. We discuss the application of our results in the context of Ocelot, which is a Lucent tool for wireless network planning and optimization.**

Keywords -- **High-speed wireless data, user-level QoS, network optimization, blocking performance, throughput performance, cdma2000, traffic engineering, Processor-Sharing model, page response times, Ocelot planning tool.**

Introduction

The highly anticipated introduction of wireless data services over 3rd-generation (3G) wireless networks is expected to raise new challenges in planning, deployment and operation of these networks. IP packets, the pervasive mode of data communication in wired networks, is likely to carry over to wireless data networks, bringing with it unique problems of scheduling packet transmissions under the widely varying channel conditions typical of most real environments. Channel-aware and traffic-aware scheduling, see e.g. [AKRSVW2000, BBGPSV2000, BV2001, BW2002, JKKS2002, JPP2000], that exploits the delay tolerance of data is a key source of performance enhancement for wireless data networks. Higher layers of flow control (hybrid ARQ, TCP/IP), the presence of fewer users, who have more diverse, bursty and less predictable traffic behavior than voice users, and the need to support various types of QoS requirements are additional factors that render the control and prediction of data performance significantly more complex in these systems. Hence, optimal planning of these networks is considerably more complex than conventional voice networks, and requires development of novel techniques. The challenge is particularly acute during application to actual customer networks, which deviate significantly from the idealized "flat earth, uniform traffic density, and hexagonal cell geometry" networks often used for theoretical analyses.

Planning wireless networks with the Ocelot tool

Ocelot is a predictive optimization tool developed at Bell Labs to enhance the performance of wireless networks. It originated more than four years ago with 2nd-generation (2G) voice-only wireless networks (e.g., IS-95, GSM, and IS-54/136) as the intended application. Ocelot solves an optimization problem by adjusting various base station parameters (e.g., antenna azimuth and down-tilt angles, and sector power levels) in order to maximize the coverage and capacity of the network, subject to user-specified QoS constraints. Ocelot has been successfully applied during the network design stage, as well as to operational networks for post-deployment optimization. In addition to a significant reduction in manual planning, substantial gains in performance have also been obtained through Ocelot optimization in more than 100 metro markets worldwide.

For 2G systems there is only a single service (voice) whose primary QoS measure is blocking. With the advent of 3G networks, however, the complexity has grown substantially. Now network planning must also contend with data services, with their more complex QoS requirements. Over the last two years, several novel features have been introduced into Ocelot specifically to address data services within 3G systems. Those features specific to combined voice plus data networks such as 3G-1X and UMTS will be described elsewhere. This article specifically focuses on the efforts at Bell Labs to enhance Ocelot for 3G data-optimized networks such as 1xEV-DO [TE]. For application to 1xEV-DO, it was necessary to select appropriate traffic models for FTP, Web browsing, and other common data applications, as well as develop analytical models to predict scheduler performance and the QoS for data users. Given the complexity of the goal, several simplifying assumptions and approximations are necessary to obtain a tractable formulation, and due care must be exercised to retain the essential features necessary for realistic modeling of these networks. These aspects are described in detail in the rest of this paper.

The simplified flow chart below illustrates the salient modules of Ocelot and their inter-relations.

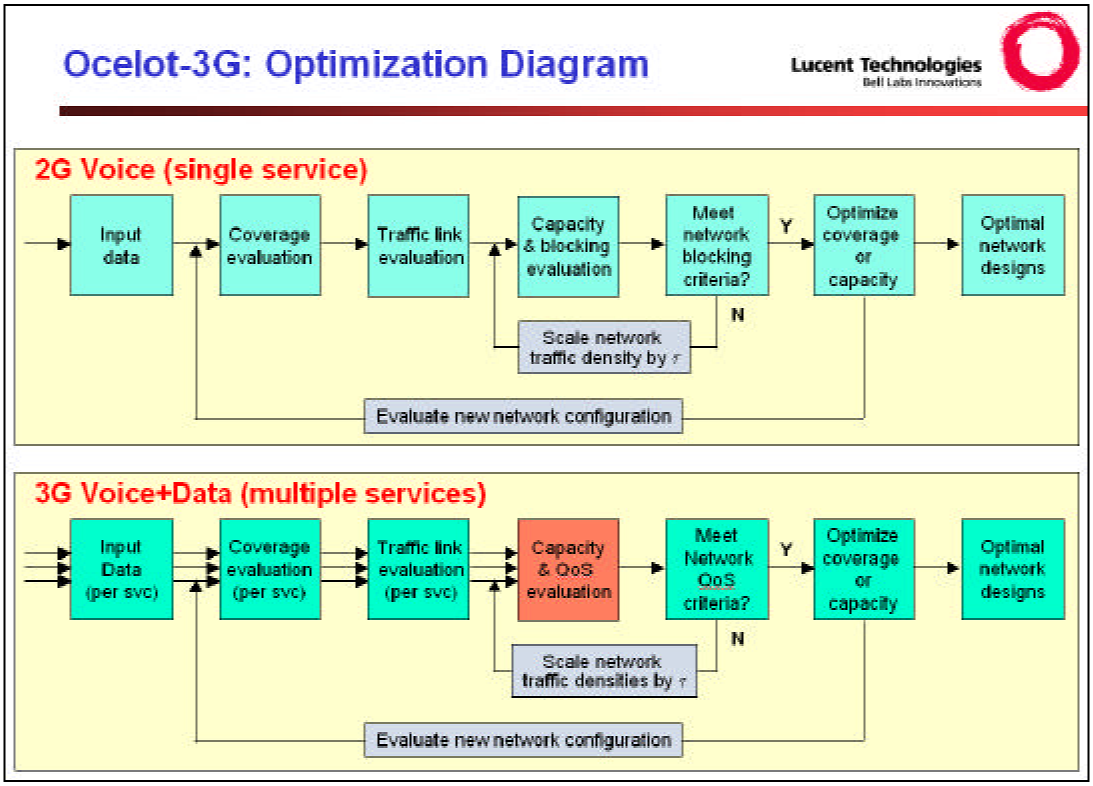
Ocelot-3G: Optimization Diagram
Lucent Technologies
Bell Labs Innovations
2G Voice (single service)
Input data
Coverage evaluation
Traffic link evaluation
Capacity & blocking evaluation
Meet network blocking criteria?
Y
Optimize coverage or capacity
Optimal network designs
N
Scale network traffic density by γ
Evaluate new network configuration
3G Voice+Data (multiple services)
Input Data (per svc)
Coverage evaluation (per svc)
Traffic link evaluation (per svc)
Capacity & QoS evaluation
Meet Network QoS criteria?
Y
Optimize coverage or capacity
Optimal network designs
N
Scale network traffic densities by γ
Evaluate new network configuration

Figure 1. Ocelot optimization flow chart

Observe that the major difference between the 2G (Voice) and 3G (Voice plus Data) versions is the addition of data-specific QoS evaluations in the latter, while the former consists mainly of blocking evaluation based on standard circuit-switched models. The input data to Ocelot includes terrain and propagation information, base station locations, traffic density proportions within each elementary region of the covered area, in addition to traffic characteristics and QoS requirements of the different traffic classes served by the network. Ocelot then proceeds iteratively by adjusting the antenna parameters at the base stations to optimize coverage or capacity for the current traffic density, followed by global scaling of the traffic (for its network capacity determination) followed by re-optimization, and so on. At the current stage, this goal has been achieved only in part for the 1xEV-DO version of Ocelot, in the sense that the tool currently performs the QoS evaluations for fixed traffic densities, and does not optimize the parameters. Future work will complete the loop, and is expected to provide important insights into the differences between planning voice and data networks.

Modeling 1xEV-DO traffic and QoS

From the above description of network planning and Ocelot, it is clear that the major additional elements required for planning 1xEV-DO networks are traffic and QoS models for 1xEV-DO connections. Most of the remaining part of this paper is hence devoted to these topics. It is well-known that, unlike voice, data traffic is very diverse and complex. Different applications may not only have drastically different traffic characteristics, but may also extremely diverse QoS requirements. In order to describe the most fundamental differences, it is convenient to make a broad distinction between *streaming traffic* and *elastic traffic*. Streaming traffic is produced by audio and video applications for both real-time communication and reproduction of stored sequences (or ‘traces') [3GPP2A]. Elastic traffic, on the other hand, results from the transfer of digital documents such as Web pages, files and e-mails, where the transmission rate is adaptable depending on the levels of congestion in the network.

For streaming traffic, small packet-level delay and low loss are crucial QoS requirements. For elastic traffic, on the other hand, it is not so much the delay of individual packets that is important, but the total transfer delay of the document that determines the QoS as perceived by the users. In this paper, we focus on the category of elastic traffic. A substantial majority of web-browsing and file transfer (FTP) traffic fall in this category, which is expected to be the dominant mode of data service usage in the near future. In the conlusion section, we will briefly comment on extensions to handle streaming traffic, whic h requires distinct modeling and analysis methods.

The analysis presented in this paper specifically models the QoS performance offered to elastic users in a network of cells that use the 1xEV-DO air-interface standard [TE]. The common scheduling mechanism implemented in the 1xEV-DO system is the so-called *Proportional Fair* algorithm. The actual implementation of this algorithm is quite complex, but we show that it is possible to accurately capture the essential features of the system using a modified version of a standard queuing model known as *Processor Sharing*. The Processor-Sharing model has a number of advantageous features including analytical tractability, computational simplicity and considerable robustness (almost complete insensitivity) to parameters that describe traffic and channel statistics. For example, important QoS measures such as blocking probability and throughput depend only on the mean traffic load of each cell, and not on the finer details of the traffic statistics of individual users. Similarly, the mean transfer delay experienced by individual users depends only on their mean service times. These features make the Processor-Sharing model particularly suitable for use in a network design tool like Ocelot as part of the QoS evaluation module. We refer to [BKQRW2002] for further details.

The rest of this paper is organized as follows. We begin with a preliminary explanation of the 1xEV-DO standard and its detailed features such as the channel reports, scheduling algorith and incremental redundancy. We then discuss specific traffic models for Web browsing and FTP sessions. Next, we present the Processor-Sharing model and explain how it models the Proportional Fair algorithm performance. This is followed by a section on the application to Ocelot and a section with numerical results that compare the predictions of the Processor-Sharing model with simulations. Finally, we conclude with a discussion of future extensions of this work.

	Data Rates (kbps)											
	38.4	76.8	153.6	307.2	614.4	307.2	614.4	1228.8	921.6	1843.2	1228.8	2457.6
Code Rate	1/5	1/5	1/5	1/5	1/3	1/3	1/3	1/3	1/3	1/3	1/3	1/3
Modulation Type	QPSK	QPSK	QPSK	QPSK	QPSK	QPSK	QPSK	QPSK	8PSK	8PSK	16QAM	16QAM
PN Chips / Bit	32	16	8	4	2	4	2	1	1.33	0.67	1	0.5
Encoder Packet Duration (ms)	26.67	13.33	6.66	3.33	1.67	6.66	3.33	1.67	3.33	1.67	3.33	1.67

Table 1. 1xEV-DO forward link data rate configurations

1xEV-DO system description

1xEV-DO is part of a 3rd-generation cdma2000 family of standards that is derived from Qualcomm's High Data rate (HDR) system [TE]. This system is designed to operate in a 1.25 Mhz spectrum. It is bandwidth-compatible with the IS-95 and 3G-1X systems and thus can be deployed with the same frequency plan. The air-interface on the forward link is however significantly different from that of 3G-1X. The system is highly optimized for packet data and supports delay-tolerant Internet applications such Web browsing, FTP, and e-mail. In contrast to traditional CDMA, users are time division multiplexed with short slot durations (1.67 ms), making it possible to transmit at a peak rate of 2.4 Mbps. Users can be scheduled for transmission in any slot as there is no need for separate channel setup and tear down. Each slot carries pilot and medium access control bits that indicate the user identity for that slot. Control signals are also time division multiplexed with the data traffic.

Fast channel condition feedback in the form of data rate control (DRC) bits on the reverse link is employed to control the transmission rate to the user in each slot, see Table 2. Variable-rate transmission is achieved through the use of adaptive coding and modulation schemes. Turbo codes with puncturing, and QPSK, 8 PSK and 16 QAM modulation schemes are used to achieve a rate variation from 38 Kbps to 2.4 Mbps as shown in Table 1. The fast feedback and short slot durations also allow the base station to schedule transmission to users when their channel fading conditions are most favorable. This enhances the throughput of the HDR system over power-controlled CDMA systems and is usually referred to as *multi-user diversity*. Incremental redundancy is another innovative feature of the system. Some of the code blocks are partitioned into a number of self decodable parts and transmitted over multiple time slots, with the subsequent slot transmissions occurring only when necessary, depending on the positive or negative acknowledgment from the receiver. This further improves the throughput of the system. These important enhancements over traditional CDMA systems make 1xEV-DO a promising candidate for widespread deployment in the market place for packet data applications.

SNR threshold (dB)	-12.5	-9.5	-8.5	-6.5	-5.7	-4	-1	1.3	3	7.2	9.5
Rates (kbps)	38.4	76.8	102.6	153.6	204.8	307.2	614.4	921.6	1228.8	1843.2	2457.6

Table 2. 1xEV-DO data rate table

Proportional Fair algorithm

As mentioned above, the fast feedback information and short slot duration in 1xEV-DO allow the base station to schedule transmissions to users when their channel conditions are favorable. The so-called Proportional Fair scheduling algorithm [BBGPSV2000, JPP2000] is specifically designed to achieve this objective. The key feature is to select users for transmission when their channel conditions are near-optimal in a relative sense, so as to optimize the throughput performance, while ensuring some degree of fairness among the various users. The Proportional Fair algorithm is the default scheduling mechanism implemented in current product releases of the 1xEV-DO system.

The Proportional Fair algorithm operates as follows. In each slot t, it selects user i with the maximum value of theratio $DRC_i(t)/R_i(t)$, where $DRC_i(t)$ is the instantaneous rate estimate for user i in slot t and $R_i(t)$ is the exponentially smoothed throughput of user i in slot t. Thus the time slot is not necessarily assigned to the user with the highest absolute rate, but is assigned to the user with the maximum value relative to the averaged throughput. The value of $R_i(t)$ is updated in each slot according to

$$R_i(t+1) = (1-1/T)*R_i(t) + (1/T)*Y_i(t)*DRC_i(t),$$

where $Y_i(t)$ is a 0-1 variable indicating whether or not user i is selected in slot t. The time constant T may be interpreted as the length of the interval over which the throughput is averaged. A typical value for T is 1000 slots, corresponding to 1.67 seconds.

Under certain statistical assumptions, it may be shown that the Proportional Fair algorithm maximizes the sum of the logs of the averaged throughputs of the various users. In other words, the throughput of no single user can be improved without reducing the throughputs of the other users by a greater total percentage, which property is referred to as “Proportional Fairness”.

As mentioned in Section 1, the 1xEV-DO system is expected to support a variety of data applications, such as transferring files, Web browsing, downloading e-mails and possibly streaming services as well. In the present paper we will focus on Web browsing, which is likely to be one of the dominant traffic sources.

A Web browsing session typically consists of several page requests, interspersed by `think times' between successive downloads. A page usually contains several embedded objects, each of which is segmented into packets that are then carried from the Web server to the user through a common TCP connection. Depending on the HTTP version used, the TCP connections associated with the various embedded objects are either set up in a serialized manner, or partly in parallel, with a certain maximum number of simultaneous TCP transfers.

In the present paper, we treat pages as the basic entities which are transmitted across the air-interface, and do not model packet-scale details. This simplification is sensible when the wireless link is the main bottleneck on the end-to-end path from the Web server to the user. In that case, the queue at the base station should rarely be starved during a page download, so that it is reasonable to lump individual packets into a single burst.

Processor-Sharing model

Homogeneous users

In this section we describe how the performance of the 1xEV-DO system as perceived by Web browsing users may be evaluated by means of a Processor-Sharing model. We refer to [BKQRW2002] for further details. For convenience, we first focus on a scenario with a static population of n active users with statistically identical (but not necessarily independent) channel processes. By symmetry considerations, it then follows that each of the n users receives a fraction $1/n$ of the time slots. Since the slot duration of 1.67 ms is relatively short compared to the time scale of interest for user-perceived performance, it is then natural to assume that each of the n users is continuously served at a fraction $1/n$ of the aggregate transmission rate. The latter idealization is reminiscent of the typical use of the Processor-Sharing paradigm as a convenient abstraction of Round-Robin scheduling. What is different, however, is that the aggregate transmission rate is not a fixed quantity, but is determined by the channel-aware actions of the Proportional Fair algorithm and thus depends on the number of active users n. As described earlier, the Proportional Fair algorithm basically selects in each slot the user with the maximum relative rate, i.e., the highest instantaneous rate, normalized by the smoothed throughput. Since the channel processes of the users are assumed to be statistically identical, we conclude that the smoothed throughputs of the various users should be identically distributed as well (though not necessarily independent). In addition, the smoothed throughputs should not show much fluctuation over time when the time constant T in the exponential smoothing is sufficiently large. We refer to [KW2002] for a rigorous justification of these claims. When combined, these two observations imply that the smoothed throughputs of the various users show only little variation around some common constant.

Consequently, the Proportional Fair algorithm effectively selects the user with the highest instantaneous rate in each slot. Hence, the aggregate expected transmission rate with *n* users is $H(n) = E\{\max\{DRC_1,..., DRC_n\}\}$, with $DRC_1,..., DRC_n$ representing the instantaneous rates of the various users. Since the channel processes are assumed to be statistically identical, we may write $DRC_i = R \times X_i, \quad i = 1,..., n$, and thus *H(n) = H(1) G(n)*, where $G(n) = E\{\max\{X_1,..., X_n\}\}$, where *H(1) = R* is the time-average rate and $X_1,..., X_n$ represent the fluctuations in the instantaneous rates of the various users around the time-average value. For example, if the users have independent Rayleigh fading channels and the instantaneous rate is linear in the instantaneous SNR (signal-to-noise ratio), then $X_1,..., X_n$ are independent, exponentially distributed random variables with unit mean. In that case a straightforward computation yields $G(n) = \sum_{m=1}^{n} 1/m$. Note that *G(n)* then behaves like log(*n*) as *n* tends to infinity. In the actual 1xEV-DO system, the instantaneous rate is selected from a finite set of discrete values according to Table 2. The function *G(n)* must then be evaluated numerically, and will saturate at a finite asymptote as *n* approaches infinity. We will refer to *G(n)* as the *gain factor*, since it represents the throughput gains that the Proportional Fair algorithm achieves from channel-aware scheduling, relative to the time-average rate *R*.

We now turn to a dynamic configuration of users governed by the arrival and service completion of page requests during Web browsing sessions. If the backlog periods are relatively long, i.e., if the number of active users varies relatively slowly compared to the time scale on which the Proportional Fair algorithm operates, then it is plausible to assume a separation of time scales, where each user is continuously served at a rate *H(n) / n* (in bits/second), i.e., a fraction *G(n)/n* of its time-average rate, whenever there are *n* active users.

We assume that the Web browsing sessions are initiated according to a Poisson process of rate l, and entail a generally distributed number of page requests with finite mean *M*. The page sizes (in bits) are assumed to be generally distributed with finite mean t. In particular, the page sizes are allowed to have a long-tailed distribution with possibly infinite variance, as long as the mean is finite. The `think times' can have an arbitrary distribution with a finite mean. We assume that at most *K* transfers are supported simultaneously. Page requests which are submitted when there are already *K* transfers in progress are blocked.

The total offered traffic (in bits/second) for the Web browsing users is given by $s = l * t * M$, where l is the session arrival rate (per second), t is the mean page size (in bits) and *M* is the mean number of page requests per session. For later purposes, it is convenient to also define the normalized load $r = k * s$, where $k = 1/R$ is the conversion factor from bits to seconds, with *R* = *H(1)* denoting the time-average rate (in bits/second). Note that r is a dimensionless quantity.

The above description amounts to a so-called Processor-Sharing model with varying service rate. It follows from standard results [Coh79, Kel79] that the distribution of the number of active users is given by

$$P\{N=n\}=J(K)^{-1}\frac{r^n}{f(n)},$$

where $f(n)=\prod_{m=1}^{n}G(m)$ and *J(K)* is a normalization constant. In particular, the mean number of active users is given by

$$E\{N\}=J(K)^{-1}\sum_{n=1}^{K}\frac{nr^n}{f(n)},$$

and the blocking probability is given by

$$L=P\{N=K\}=J(K)^{-1}\frac{r^K}{f(K)},$$

so that the throughput is $(1-L)*s$. Using Little's law, we obtain that the mean response time for a page is given by

$$E\{S\}=\frac{E\{N\}}{l*(1-L)},$$

The above formula reflects the celebrated insensitivity property of the Processor-Sharing discipline, which shows that the mean response time only depends on the page size distribution through its mean. In fact, it may be shown that the expected conditional expected response time is

$$E\{S\mid B=b\}=\frac{b}{E\{B\}}E\{S\}.$$

Thus, the expected response time incurred by a user is proportional to the actual page size, with factor of proportionality $D(K)=E\{S\}/E\{B\}$. This property embodies a certain `fairness principle', which implies that users requesting larger pages tend to experience larger response times. We will refer to the coefficient *D(K)* as the `stretch factor'.

Symmetric scenarios

For convenience, we assumed in the above formulation that the channel processes of all the users are statistically identical. In practice, the channel characteristics of the users will be radically different due to spatial diversity. We now extend the above-described Processor-Sharing model to such heterogeneous scenarios. We first consider a scenario where the channel processes are partially symmetric, in the sense that the users may have heterogeneous time-average rates, but that the relative fluctuations in the rates around the respective time-average values are still statistically identical. In other words, the instantaneous rates of the various users scale linearly with the time-average rates, i.e., $DRC_i = R_i \times X_i$, where R_i the time-average rate of user *i*, and $X_1,...,X_n$ represent the statistically identical fluctuations in the instantaneous rates of the various users around their respective time-average values. Now observe that the smoothed throughputs of the various users as maintained by the Proportional Fair algorithm will scale linearly with the time-average rates as well. As a result, each of the users will still receive a fraction *1/n* of the time slots when there are n users active, see also for instance [Hol2001]. In addition, it may be verified that when served, the expected rate of each user *i* is *G(n)* times its time-average rate R_i. As before, we may thus assume that each user *i* is continuously served at a fraction *G(n)/n* of its time-average rate R_i, except that the time-average rates are now no longer identical but may vary across users. However, the above-described Processor-Sharing model still applies, provided the service requirement of a user is normalized by its time-average rate. Accordingly, we now need to compute the conversion factor as $k = E\{1/R\}$, where *R* is a random variable representing the time-average rate of an arbitrary user (the randomness reflecting the spatial diversity). Note that the above formula reduces to $k = 1/R$ when the channel processes are statistically identical so that all the users have the same mean rate *R*. With this modification, the above expressions for the various performance metrics continue to hold. In particular, the mean response time will remain proportional to the mean service requirement of a user. Note however that the mean service requirement now encapsulates both the mean page size and the inverse of the time-average rate, so that the mean response times will now obviously vary across users and be proportional to the inverse of their time-average rates. In contrast, the blocking probability does not depend on the identity of the user.

Heterogeneous users

In the above treatment, we allowed the users to have heterogeneous time-average rates, but still assumed the channel processes to be partially symmetric, in the sense that the relative rate variations of the various users around the time-average values are statistically identical. The latter assumption is roughly valid when the users for example have Rayleigh fading channels and the rate is approximately linear in the SNR. This approximation is reasonable when the SNR is not too high. In practice, when the time-average rates are heterogeneous due to differences in the underlying time-average SNR, the relative rate variations will usually not be exactly identical in distribution. Typically, the relative rate fluctuations will decrease with increasing SNR due to the concavity and the truncation of the transmission rate at higher SNR values. As a result, the gain factor *G(n)* is no longer independent of the time-average rates of the users. As an approximation, we will compute the gain function assuming a common SNR for all the users, equal to the average SNR on a log scale. Observe that the `true' average SNR is likely to be lower due to the fact that the actual user population will tend to be biased towards low-SNR users which experience longer transfer delays. As a result, the approximation will tend to overestimate the value of the *absolute rate* $H(n)$. However, the approximation will tend to underestimate the value of the *relative gain factor* $G(n) = H(n) / H(1)$ due to the fact that the relative gains from scheduling tend to be lower for high-SNR users as explained above. Consequently, we expect the resulting approximations for the response times, the blocking probabilities and the throughput to be conservative. We will examine these issues in greater depth when we discuss the numerical experiments in the next section.

Computation of gain factor

In order to compute the gain function for a common time-average SNR, it remains to characterize the distribution of the instantaneous rate given the time-average SNR. In view of the mapping of Table 2, we thus need to specify the distribution of the instantaneous SNR estimate. We will assume that the instantaneous SNR estimate is exponentially distributed, corresponding to one-path Rayleigh fading. In the case of two-path Rayleigh fading, the instantaneous SNR could be assumed to have a chi-squared distribution with two degrees of freedom. This would reduce the variation in the instantaneous SNR, and negatively affect the gain function.

The above procedure is reasonable in low-mobility scenarios with correspondingly low Doppler frequencies. In high-mobility scenarios, it is necessary to account for the fact that the instantaneous SNR estimate will typically be lowered due to the delayed feedback. Specifically, the DRC information fed back from the mobile is not available at the base station until several frames after the time the DRC was actually determined by the mobile because of processing and transmission delays. Since the channel is time-varying the channel conditions at the time of transmission could become significantly different from that at the time when the DRC was determined at the mobile, depending on the speed of the mobile which in turn determines the time correlation of the fading channel. Thus it is possible to enhance the performance of the system by employing a predictor at the mobile to predict the channel conditions at the time when the DRC is actually used by the base station. When such a prediction scheme is used by the mobile, it becomes necessary to model the predictor and include its effects in the calculation of the gain factor in order to compute the throughput and delay performance accurately.

Application in Ocelot

The model of the previous section requires as input the traffic offered to each sector, the conversion factor k for that sector, and the mean rate. These are readily found using the cumulative distribution functions (CDFs) of the SNR and traffic for each sector. So all that is required for incorporation of the 1xEV-DO model into Ocelot is the determination, for each sector, of such CDF's, which are derived from radio link estimates, and from Ocelot's model of wireless traffic.

Ocelot models the distribution of wireless traffic as a mesh (plane graph), and for the purposes here, it is enough to consider the vertices of that graph. Each vertex represents a potential location for wireless users, and so the distribution of the vertices represents the distribution of wireless traffic. For additional flexibility in the representation, each vertex is annotated with a value that represents an estimate of the Erlangs of 1xEV-DO users at its location. In addition, for each location and each sector, the path loss to the location from the sector has been calculated, and many other quantities.

In order to determine the contribution of a given location to a sector's 1xEV-DO load, Ocelot uses a simple model of shadow (slow) fading. Ocelot uses the common model of shadow fading as a log-normally distributed random variable. Together with the path loss estimates and sector power levels, this shadow fading model implies, at a given location, a probability for each sector that the sector has the maximum SNR and therefore will be serving 1xEV-DO traffic for the location. (Note that this calculation is distinct from the later calculations of the 1xEV-DO model, incorparating the effect of fast (Rayleigh) fading.) Currently, the Ocelot calculation is conservative regarding these probabilities: the SNR for a sector, conditioned on the sector having the maximum SNR, is likely to be higher than the unconditioned SNR for the sector, but Ocelot's calculation ignores this conditioning. These probabilities, together with the SNR estimates for each location/sector pair, and Erlang values for each location, constitute the CDF's needed to compute the key quantities for modeling 1xEV-DO.

It is worth remarking that currently Ocelot's estimates of the probability that a given sector will have the maximum SNR are approximations, using a scheme due to Ocelot co-creator John Hobby. This is motivated by the need for computational speed. Based on the lognormal approximation, there is an SNR value X such that the expected number of sectors above that value is one half. Ocelot computes that value, and approximates the probability that a sector is maximum according to the probability that the sector SNR is above X. Note that if a sector does have SNR above X, then with probability at least one half, the sector does have the maximum value.

Numerical results

Our results are obtained using a trace-driven simulation of a single 1xEV-DO base station/sector, each trace value determining a user channel at a given time slot. The traces themselves are emulations of one-path Rayleigh fading channels using a well-known oscillator model, originally due to [Jak74], pages 65-76. (Other channel models can equally well be examined, e.g., Ricean, two-path Rayleigh fading, etc). Low fading frequencies (~5Hz) are used throughout and it is assumed that there is perfect channel prediction. This is an unrealistic assumption at higher fading frequencies where both simulation and experimental trials show reduced gains arising from degradation in the performance of the channel predictor.

The time constant in the Proportional Fair algorithm is taken to be 1000 slots or roughly 1.67 seconds. The file sizes used in the simulations lead to response times much longer than this in the vast majority of cases, and so the convergence time of the Proportional Fair algorithm can be safely neglected. Indeed, a series of preliminary experiments were conducted which show that file sizes as short as 12.5 kbytes on the average would still give reasonably accurate comparisons with simulation. These further reflected the anticipated degradation in performance of the Proportional Fair algorithm when the file size was reduced still further. (Such performance impacts may be limited by careful choice of the initial value of the throughput estimates used by the Proportional Fair scheduler.)

Our first results are for the base station/sector supporting a population of mobiles conducting Web browsing sessions. We use a mean-SNR cumulative distribution function taken from [BBGPSV2000] which may be regarded as typical. (In practice the analysis is conducted over Ocelot supplied mean-SNR CDF's for the cell/sectors in the service area which is being planned out.)

The Web browsing model consists of a (geometrically distributed) number of page request (M = 20 pages on the average) with a constant page size t = 40 kbytes. The interval between page requests corresponds to a user think time taken to be exponential with mean 40 seconds. These numbers match the parameter values specified for the HTTP traffic model in [3GPP2B]. (Neither the geometric assumption nor the assumptions for the think time are needed for the analytical model.) Coupled with the distribution function F, these assumptions determine the average load (in seconds of required transmission time) per user. The scheduler was limited to a maximum of K = 15 simultaneous page requests and other page requests were supposed to be blocked and cleared. (In the simulation the page blocking was estimated by measuring the time congestion, as opposed to counting the page losses. Time and page congestion are equal provided the arrival process is Poisson.)

Figure 2 shows the mean time to download a page versus the arrival rate l of Web browsing sessions. Each simulation point was produced for an interval corresponding to 10 million slots or roughly 5 hours of system time. The dotted curve gives simulation results for the case where each page has a constant size. The dot-dashed curves are corresponding results in which the file distribution was changed to be Pareto with exponent a = 3 (and hence finite variance) with the same mean as before.

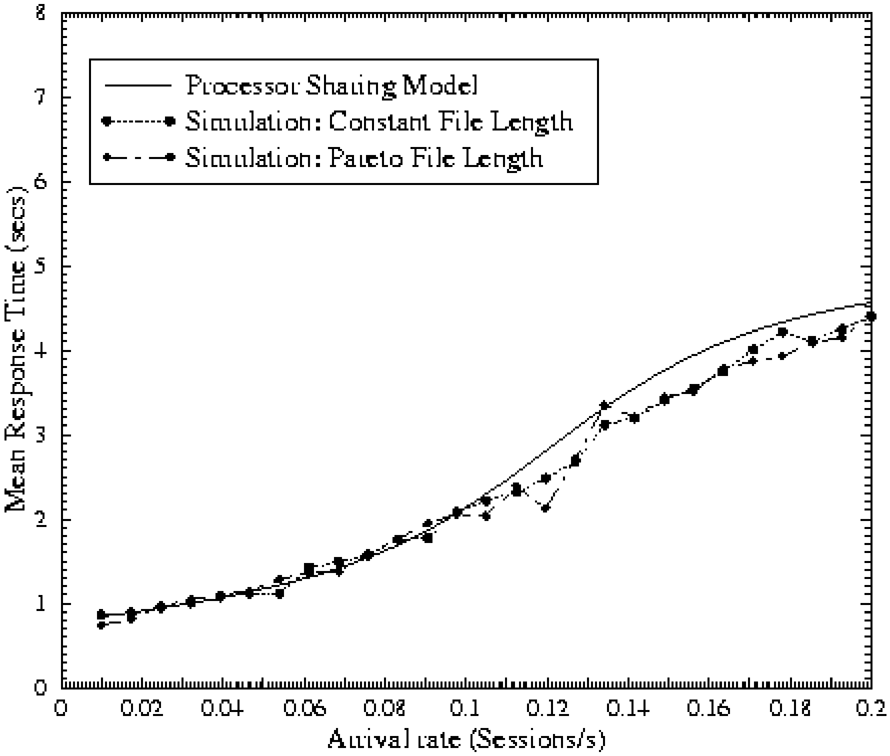

Figure 2. Mean response time for constant and Pareto file size distributions

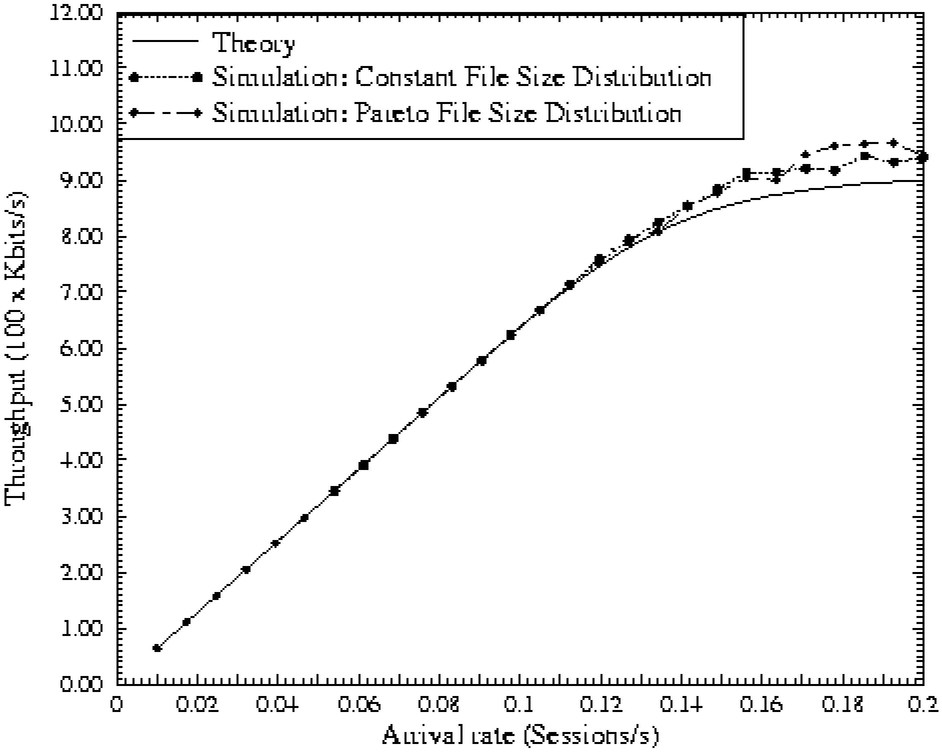

Figure 3. Throughput for constant and Pareto file size distributions

Our results show good agreement with the estimate from the Processor-Sharing model which is shown as a solid curve. The reduced rate of increase of the mean transfer time with the session arrival rate l is a consequence of the limit on the maximum number of simultaneous transfers. Further observe that the insensitivity property is confirmed, as the results for Pareto and constant file size lie close to one another. Figure 3 shows the corresponding system throughputs. These latter values were obtained as the product of the given offered traffic and the estimated page acceptance probabilities.

Distinct CDF's

To examine the validity of the analytical model, results were obtained for several base station/sector CDF's and compared with simulation. Here we present results for two cases from this study. In this case the traffic originates from mobile users which are requesting (single) file transfers. Each file is taken to be constant size, 50 kbytes.

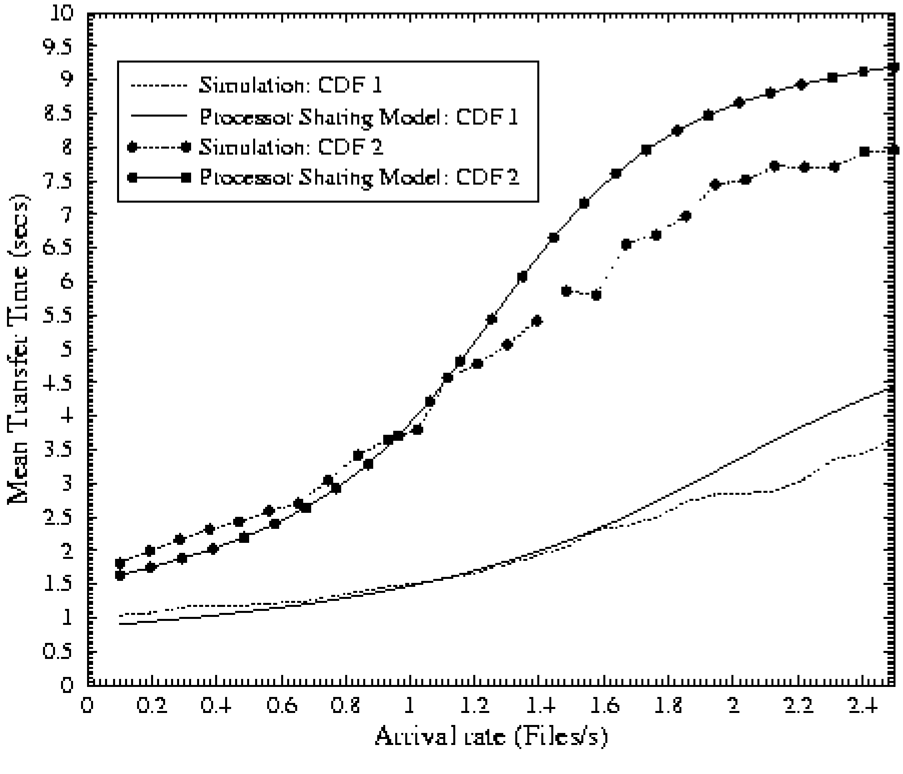

Figure 4. Mean response time for two randomly selected SNR CDF's

Figure 4 depicts the file response times for the two cell CDF's, and shows the marked difference in delay that can arise as a result of differences in user distribution, propagation, network topology etc. (It is this type of performace difference which the planning tool is designed to compensate for by balancing cell/sector loads to QoS targets.) As can be seen, the theoretical and simulation results match up and there is close agreement at low load, but the analytical results overestimate the mean response time somewhat at higher load.

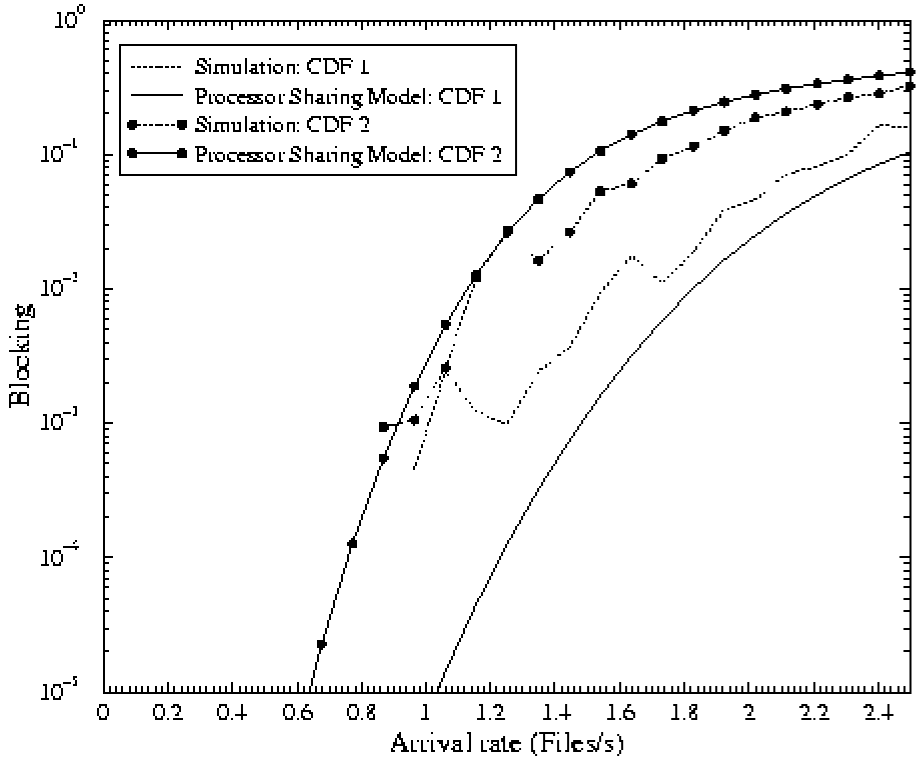

Figure 5. Blocking for two randomly selected SNR CDF's

The corresponding results for blocking are depicted in Figure 5 which also shows a reasonable match between simulation and theory.

Limiting throughput

In our results so far, the relative throughput gains from scheduling have been determined by obtaining the mean SNR over the sector CDF on the log (dB) scale. The approximations tend to *underestimate the relative gains* due to scheduling. This is particularly true for higher numbers of competing users. This is reflected in our mean response time results which consistently overestimate at higher loads. However, the same approximation *overestimates the absolute throughputs* as already discussed. A more accurate estimate of absolute throughput can be obtained by averaging the SNR *weighted* with the mean response times of pages/files. According to the Processor-Sharing approximation, these are inversely proportional to $R(g)$, the mean declared rate as a function of the time-average SNR g, which are thus used instead, giving

$$E\{\Gamma\} = \frac{\int g \,/\, R(g)\, p(g)\, dg}{\int p(g) \,/\, R(g)\, dg}\,.$$

Here p denotes the mean SNR density. The solid curve in Figure 6 graphs analytical estimates of the maximum throughput at fixed admission threshold K calculated using this weighting (mean response time weighting of the SNR).

In the above connection, it may be helpful to think of maximum throughput being determined via the following imaginary experiment. Fix K, and start K simultaneous page transfers with users taken at random from the cell SNR CDF F. Each time a page transfer is completed, replace the completed user with another random user, selected according to F, and commence this new users page transfer. The long-run estimate of throughput from this experiment is the maximum possible throughput the system can achieve. This is the case, because pages are blocked independent of the user, and higher throughputs cannot be achieved by blocking additional users without their page sizes and associated mean SNR's being known a priori.

Figure 6 graphs simulations of the throughput in the Web browsing model as a function of K for various session arrival rates. As the graphs show, the analytical throughput bounds do indeed form an envelope for the family of throughput curves. Note also that the maximum throughput increases only slightly with increasing K, reflecting the very small likelihood that the highest rates in Table 2 will be achieved by users with low SNR's. The graphs also show that the throughput flattens out (saturate) as K increases and approaches a second limit, the maximum which can be carried for a given session arrival rate l .

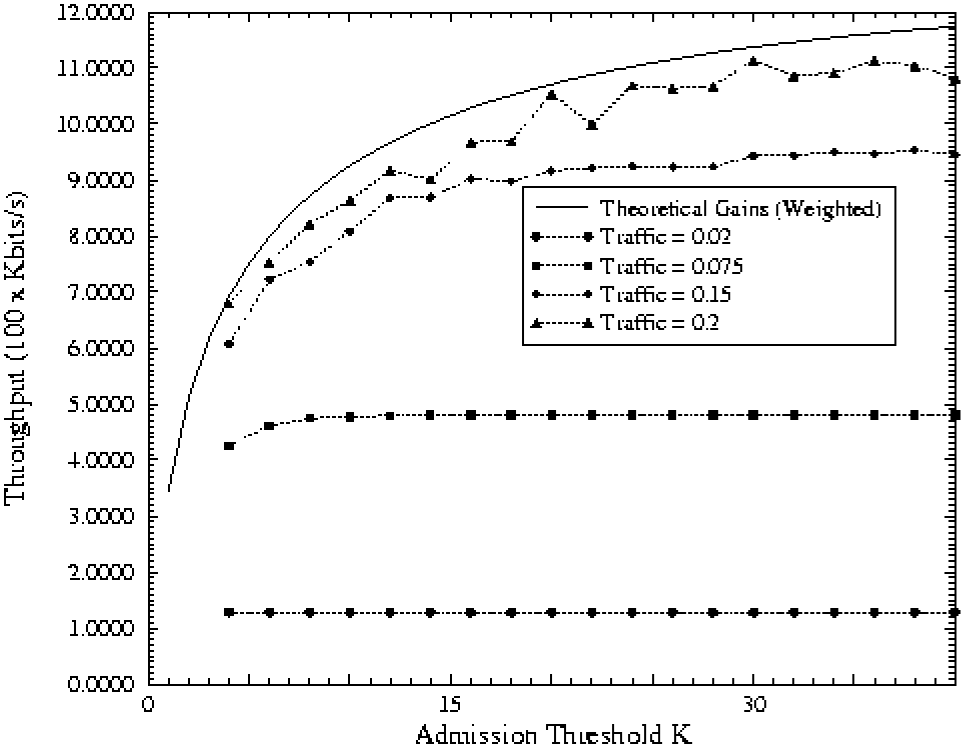

Figure 6. Limiting throughput versus admission threshold K

These results can also be seen from Figure 7 which shows throughput against session arrival rate with an admission threshold of *K* users. The horizontal lines are taken from the envelope curve in Figure 6 The small discrepancies show that response time weighting slightly underestimates the throughput.

Additional numerical results may be found in [BKQRW2002].

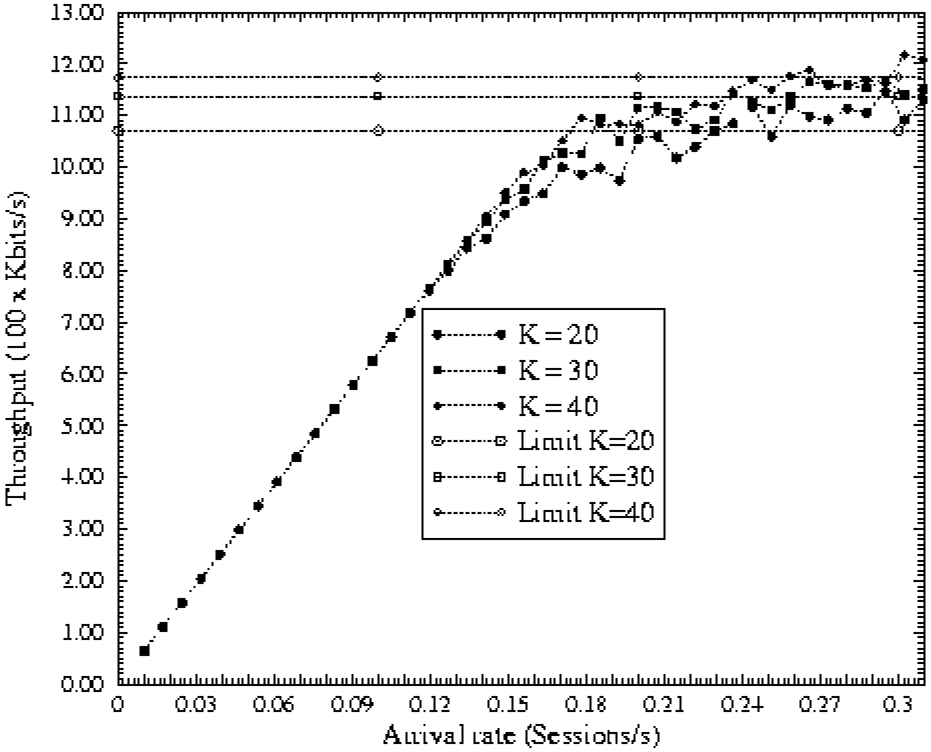

Figure 7. Throughput versus offered traffic lambda

Conclusion

We have described enhancements to Lucent's wireless network planning tool known as Ocelot that enable evaluation and planning of 3rd-generation 1xEV-DO networks. A critical feature of 1xEV-DO is a channel-aware scheduler that uses the Proportional Fair algorithm [BBGPSV2000, JPP2000] to improve system performance. This paper presents novel analytical techniques to evaluate the performance of the Proportional Fair algorithm. While the algorithm itself is quite detailed and complex, we show that it is well approximated by a modified version of a standard queueing model known as Processor Sharing. The approximation permits easy estimation of user-level QoS measures for elastic traffic such as mean file transfer time, mean throughput and blocking for FTP and Web browsing connections. The model is parsimonious in the sense that a single *gain factor* accounts for the scheduling gains that result from individual users' channel variations. Simulations suggest that this parameter can be reasonably accurately estimated from channel statistics even for heterogeneous user populations with different fading characteristics. Another advantageous feature of the proposed model is its insensitivity to the detailed traffic characteristics of individual users. The blocking and throughput depend only on the average system load, while the mean transfer delay for each user depends only on the mean service time he/she receives. These features of the Processor-Sharing model make it particularly suitable for use in a network design tool like Ocelot.

Several enhancements to the Processor-Sharing model are clearly possible, with some being more important than others. One that is particularly relevant is the incorporation of the effects of higher-layer protocols such as TCP and HTTP on data QoS. Preliminary results indicate that these protocols could have significant impact on end users' QoS experience, especially when the system is heavily loaded. For example, additional delays are introduced by page request/page get processes, with the amount of delay depending on Web page statistics and the way HTTP1.1 organizes objects. Pages could arrive in bursts and cause starvation in the buffers at the base station. The TCP three-way hand shake is another source of delay. When combined, these extra delays may cause the round trip time (RTT) to exceed the retransmission time out (RTO) and result in TCP time out. The performance could further degrade if there is packet loss, since this would cause RLP/TCP retransmissions and possibly RLP/TCP time out. Other important modifications to the present analysis include modeling ARQ (incremental redundancy), channel measurement/estimation errors and feedback delays.

Future work will require the computational extension of our model to include the efficient calculation of derivatives. While Ocelot currently models (and optimizes for) UMTS and 3G1X packet data traffic, many extensions and refinements of that work are needed. For example, Ocelot does not presentlyly model streaming data traffic [3GPP2A] and cannot handle multiple priorities. However, in modeling the effect of power amplifier limitations, the power requirements for streaming traffic are sufficiently close to circuit-switched that it is possible that the modeling algorithm currently used could be applied to streaming traffic as well. The validity of such an approach would need to be verified, to assure that, among other things, the effect of the gain due to channel-aware scheduling is relatively small. A more elaborate model of streaming traffic may be needed. For example, methods for the computing the QoS for a large class of streaming data applications under the algorithm proposed in [SR2001] are presented in [KRWB2002]. These methods may be simple enough to incorporate into Ocelot. However, the more general problem of predicting the performance seen by streaming applications under channel-aware scheduling algorithms, such as those considered in [AQS2002], remains a subject for future work.

Acknowledgments

The authors would like to thank their colleagues, John Hobby, Krishnan Kumeran, Lijun Qian, Kavita Ramanan, and Howard Trickey, who were significant contributors to the development of this paper.

References

[TE] TIA/EIA/IS-856. CDMA2000, High Rate Packet Data Air Interface Specification.

[3PPP2A] 3GPP2 S.R0021 (2000). Video Streaming Services - Stage 1.

[3GPP2B] 3GPP2 (nnnn). 1xEV-DV Evaluation Methodology – Addendum (V6).

[AKRSVW2000] D.M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar, P.A. Whiting (2000). CDMA data QoS scheduling on the forward link with variable channel conditions. Technical Memorandum, Bell Laboratories, Lucent Technologies.

[AQS2002] D.M. Andrews, L. Qian, A.L. Stolyar (2002). Proportional Fair and maximum-throughput algorithms with minimum-rate constraints. Technical Memorandum, Bell Laboratories, Lucent Technologies.

[BBGPSV2000] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushayana, A. Viterbi (2000). CDMA/HDR: a bandwidth-efficient high-speed wireless data service for nomadic users. *IEEE Commun. Mag.* **38**, 70-77.

[BV2001] Q. Bi, S. Vitebsky (2001). Performance analysis of 3G-1X EVDO High Data Rate system. Technical Memorandum, Bell Laboratories, Lucent Technologies.

[BKQRW2002] S.C. Borst, K. Kumaran, L. Qian, K. Ramanan, P.A. Whiting (2002). Queueing models for user-level performance of Proportional Fair scheduling. Technical Memorandum, Bell Laboratories, Lucent Technologies.

[BW2002} S.C. Borst, P.A. Whiting (2002). Dynamic rate control algorithms for HDR throughput optimization. *IEEE Trans. Veh. Techn.*, to appear.

[Coh79] J.W. Cohen (1979). The multiple phase service network with generalized processor sharing. *Acta Informatica* **12**, 245-284.

[Hol2001] J.M. Holtzman (2001). Asymptotic analysis of Proportional Fair algorithm. In: *Proc. IEEE PIMR Conf. 2001*, 33-37.

[Jak74] W.C. Jakes (1974). Multipath interference. In: *Microwave Mobile Communcations*. Ed. W.C. Jakes, IEEE Press, Piscataway.

[JPP2000] A. Jalali, R. Padovani, R. Pankaj (2000). Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system. In: *Proc. 50th IEEE Veh. Techn. Conf.*, 1854-1858.

[JKKS2002] N.S. Joshi, S.R. Kadaba, G.N. Kumar, G.S. Sundaram (2002). Differentiating users and services in wireless packet networks. Technical Memorandum, Bell Laboratories, Lucent Technologies.

[KW2002] H.J. Kushner, P.A. Whiting (2002) Convergence of Proportional Fair sharing algorithms under general conditions. Technical Memorandum, Bell Laboratories, Lucent Technologies.

[Kel79] F.P. Kelly (1979). *Reversibility and Stochastic Networks.* Wiley, Chichester.

[Ram2002] K. Ramanan (2002). Capacity regions for wireless systems having QoS constraints.

[KRWB2002] K. Kumaran, K. Ramanan , P.A. Whiting, S.C. Borst (2002). Optimal capacity regions for streaming traffic with QoS constraints. Technical Memorandum, Bell Laboratories, Lucent Technologies.

[SR2001] A.L. Stolyar, K. Ramanan, Largest Weighted Delay First scheduling – large deviations and optimality, *Annals of Applied Probability* **1**, 1-49.