TCP-Aware Resource Allocation in CDMA Networks

Majid Ghaderi
University of Waterloo
mghaderi@cs.uwaterloo.ca

Ashwin Sridharan
Sprint ATL
ashwin.sridharan@sprint.com

Hui Zang
Sprint ATL
hui.zang@sprint.com

Don Towsley
University of Massachusetts, Amherst
towsley@cs.umass.edu

Rene Cruz
University of California, San Diego
rcruz@ucsd.edu

ABSTRACT

TCP is the dominant transport protocol over both wired and wireless links. It is however, well known that TCP is not suitable for wireless networks and several solutions have been proposed to rectify this shortcoming. In this work, we explore cross-layer optimization of the *rate adaptation* feature of cellular networks to optimize throughput of a single long-lived TCP session. Modern cellular networks incorporate RF technology that allows them to dynamically vary the wireless channel rate in response to user demand and channel conditions. However, the set of data rates as well as the scheduler's rate adaptation policy are typically chosen to optimize throughput for *inelastic* applications. In order to optimize such a system for TCP, we propose a *two state* TCP-aware scheduler that switches between two channel rates as a function of the TCP sending rate. We develop a fluid model of the steady-state behavior of a TCP session in such a system and derive analytical expressions for TCP throughput that explicitly account for rate variability as well as the dependency between the scheduler and TCP. Using the model we choose RF layer parameters that, in conjunction with the TCP-aware scheduler, improve long-term throughput of a single TCP flow by $15-25\%$. We also compare our analytical results against those obtained from *ns-2* simulations and confirm that our model indeed closely approximates TCP behavior in such an environment.

Categories and Subject Descriptors: C.2.2 [Computer Systems Organization]: Computer-Communication Networks

General Terms: Design, Performance.

Keywords: Wireless networks, TCP throughput, Rate adaptation, Cross-layer optimization.

1. INTRODUCTION

Modern digital communication technologies combined with powerful mobile processors now allow wireless channel schedulers in cellular networks to rapidly change the allocated channel resources in response to channel conditions as well as user demands. This is achieved by controlling various parameters (and combinations thereof) such as the coding rate, spreading factor, modulation scheme[1] and link layer re-transmission rate. For ease of exposition, we shall refer to these jointly as *RF Control Variables*. These variables essentially trade-off data rates for improved *frame error rates* (FER) and vice versa.

Cellular networks typically specify various combinations of the RF control variables that result in a set of *allowed* data rates and corresponding FER. The RF scheduler dynamically assigns rates from this allowed set based on its rate adaptation policy. For example, in the CDMA2000 1xRTT network [2], the scheduler can dynamically transition between four different data rates during a mobile's session in response to buffer content and channel conditions by varying the spreading factor through the *Walsh code*[2] length. A shorter *Walsh* code lowers the spreading factor, which results in higher data rates but at the cost of lower SINR (Signal to Interference and Noise Ratio) and hence higher frame error rates. Channel schedulers in modern third generation cellular networks such as W-CDMA and 1xEV-DO, can allocate from ten different transmission rates in each time slot to each user.

In practice, the above mentioned factors that decide the set of allowed data rates as well as the scheduler's rate adaptation policy are chosen to optimize the *raw* physical layer *goodput* of a user. The set of data rates is obtained by choosing a combination of the RF control variables that produces the highest channel data rate under a particular channel condition[3] for a target Frame Error Rate (FER). Similarly, scheduling policies typically involve the assignment of the highest possible data rate allowed (for a given channel condition) from the set of given data rates that can clear the buffer backlog[4](See [24] for a detailed characterization of this behavior). While ideal for *inelastic* constant rate applications, this methodology for resource allocation can produce sub-

[1]Signal power is typically used to offset noise and interference rather that directly increase data rate.

[2]In CDMA systems, the Walsh code is an orthogonal code used to identify each user and mitigate interference from other users.

[3]A channel condition is defined by a particular range of SINR values and is a function of fading, interference, *etc.*

[4]The CDMA2000 1xRTT is an example of a commercial system with such features.

MobiCom'06, September 23–26, 2006, Los Angeles, California, USA.

optimal performance of *elastic* applications and protocols, in particular TCP, that adapt their rate in response to feedback from the receiver.

As is well known, TCP, by far the most dominant transport protocol, uses the additive increase multiplicative decrease (AIMD) algorithm that gradually increases its transmission rate based on receiver feedback and rapidly throttles back when it perceives losses (either due to congestion or channel errors). Given this complex relation between TCP throughput and the channel transmission and loss rate, the same trade-off in channel capacity and frame errors that works for inelastic traffic may yield data rates and FERs that degrade TCP throughput. Similarly, a scheduler's rate adaptation policy that always aims to clear buffer backlog can be sub-optimal for TCP. For example, when the TCP source has a small window and is ramping up its rate, it is very sensitive to losses but not to the assigned channel rate. In such a state if the RF scheduler allocates a high channel rate at the expense of a higher FER (perhaps due to a sudden accumulation of buffer backlog as a result of jitter in the network), the TCP source cannot fully utilize the high rate and in fact may drop its window or time-out due to the channel errors. Conversely, for larger windows (higher TCP sending rates), it may be advisable to allocate higher channel rates even at the expense of higher bit error rates, since low channel rates will inevitably result in packet losses due to congestion. On a related note, a previous study found that sharp bandwidth oscillations induced by rate adaptation of the RF scheduler in CDMA networks that are agnostic to TCP result in throughput degradation [14].

The above discussion clearly motivates the case for a "TCP-aware" scheduler as a means to improve throughput of a TCP session on a wireless channel[5]. In order to achieve this objective, such a scheduler should be able to: a) choose control variables such as coding rate etc. that yields the optimal set of data rates (and corresponding frame error rates) from the perspective of *TCP throughput* and b) adapt its rate in a manner that is cognizant of TCP dynamics. The proposal of a simple scheduler that captures the above mentioned properties and analysis of its performance forms the main objective of this paper. *Specifically, we propose a two-rate wireless channel scheduler that changes its rate in response to the TCP sending rate and build an analytical model to compute the bulk TCP throughput of a session in such an environment. We then show how the model can be used to optimize control variables like coding rate to obtain the set of data rates that maximizes TCP throughput.*

Several previous studies have explored the subject of cross-layer optimization of error coding rate, signal power *etc.* to maximize TCP throughput in CDMA networks. However, none of them have considered TCP-aware rate adaptation and related RF layer optimization. We dwell in more detail on related work in Section 2.

Our contribution can be summarized as follows:

1. We propose a simple two-state wireless channel scheduler that adapts its state in response to TCP sending rate. Each scheduler state results in a different transmission rate, round trip time and FER. This system is used to study the benefits of cross-layer optimization of the dynamic rate adaptation feature of modern cellular networks with respect to TCP throughput. We wish to state that in the current work, the proposed scheduler operates on a *single* TCP session. Extensions to multiple users and flows is discussed in Section 6 as part of future work.

2. We develop analytical expressions for the steady state throughput of a long-lived TCP session in such an environment. Our model explicitly captures the dependency of the scheduler on TCP sending rate as well the impact of the presence of two distinct rates and frame error rates on TCP.

3. Our analytical model includes both types of TCP configurations. One, when the TCP window size is allowed to grow large enough so that the session experiences both channel and congestion losses, and two, when the maximum window size is constrained by the receiver advertised window size and hence TCP experiences only channel related losses.

4. We demonstrate how these analytical expressions can be utilized for the selection of RF control variables that maximize TCP throughput. For example, we identify the optimal coding rates to be used in each of the two states when coding rate is used to control data rate and FER. The model is also applied to determine the optimal spreading factors, which is representative of rate control in current CDMA networks. Our studies show that throughput improvements of the order of $15-25\%$ can be obtained for a single TCP session by optimization of the rate adaptation feature.

The remainder of the paper is structured as follows. Section 2 discusses the related work in more depth and contrasts it with our current work. Section 3 presents our system model of a TCP-aware RF channel scheduler. Section 4 presents our TCP model that captures the correlation between the TCP session and the scheduler. We derive analytical expressions for TCP throughput that explicitly accounts for the presence of two channel states. Section 5 validates the accuracy of the model against *ns-2* simulations. We also demonstrate the utility of this model through numerical optimization of the channel rates to maximize TCP throughput. Our conclusions as well as future work including extensions to multi-session (and user) scenarios are discussed in Section 6.

2. RELATED WORK

Numerous approaches have been proposed in the literature to optimize TCP performance in wireless networks. These approaches can be broadly categorized as either *TCP enhancement* approaches or *link-layer* optimization approaches.

The approach presented in this paper belongs to the latter category and is complementary to TCP enhancement approaches. Hence, notwithstanding their importance, we only briefly mention some of the works in this category. TCP enhancement consists of approaches that either introduce end-to-end TCP modifications or *split* the TCP connection with the help of an intelligent agent. A few examples of the former are TCP Westwood [10], TCP-Freeze [19] and the Eifel timer [23]. Examples of the latter are Snoop [6] and the ACK and Window-regulator [12, 13]. We refer the reader to [8, 17] for a more detailed survey.

[5]Such an approach falls in the realm of RF cross-layer optimization.

The framework presented in this paper is a link-layer optimization approach that, rather than modify TCP to adapt to RF dynamics, adapts the RF layer to TCP dynamics. In this view, our work is closer in philosophy to previous literature that optimizes link layer parameters like Forward Error Correction (FEC) (or coding rate), Automatic Repeat reQuest (ARQ) and RF scheduling to improve TCP throughput.

References [7, 22] analyzed the trade-off between TCP throughput and the amount of FEC added by the link layer. They showed that there exists a coding rate that maximizes TCP throughput though they only consider channel error losses. Reference [9] also conducted a similar study but included the impact of signal power and ARQ as well. Baccelli *et al.* [3] developed an analytical model of TCP that includes the impact of congestion losses due to a finite capacity channel. They used this model to study the impact of both coding rate and processing gain on TCP throughput. See also [15, 28, 16] for other cross-layer optimization techniques proposed to improve TCP throughput. All these previous studies however consider a static scenario with only a *single* coding rate and cannot be used to analyze dynamic rate variations.

Adaptive coding on the fly has been studied by the authors of [22, 20] via simulations. In both cases, however, the scheduler behavior is agnostic to TCP *state* and each coding rate is chosen based on expressions for the *long term* throughput of TCP in wireline networks. This ignores the correlation over short time scales between TCP and the scheduler.

Chan *et al.* [13] proposed a flow-level scheduler, the Short Flow Priority (SFP) scheduler, to optimize the performance for short-lived TCP flows and showed through simulations that it performs better that the combination of Proportional Fair (PF) for user-level scheduling and FIFO for flow-level scheduling. Reference [21] proposed a TCP rate aware multi-user EV-DO scheduler to minimize time-outs that is shown to perform well through simulations. Neither, however considers the impact of optimization of RF level parameters on TCP throughput or the correlation between the TCP rate and the scheduler's state. Mattar *et al.* [24] performed a detailed characterization of TCP behavior on CDMA2000 1xRTT channels and the role of key factors like the RLP (Radio Link Protocol) layer and channel conditions. They identified the strong correlation between the TCP sending rate and the channel rates of the CDMA2000 scheduler, as a key characteristic of such networks.

References [1] and [12] modeled TCP in the presence of variable round trip times and packet losses on wireless links. However, they assume that the variability is independent of TCP dynamics which is at odds with the environment considered here as well as the commercial environment studied in [24]. Finally, the authors of [27] studied optimization of transmission power to maximize TCP throughput. They explicitly considered TCP dynamics in the selection of the transmission power level. However the resulting solutions were quite complex requiring detailed TCP state knowledge. Furthermore, our focus is also different from this work since we study the impact of rate adaptation which was not considered.

3. SYSTEM MODEL

The next two sections of the paper are devoted to presentation of the system model. In this section we propose a TCP-aware RF scheduler that allocates resources based on the TCP sending rate. For ease of exposition, we use the CDMA2000 1xRTT system as an example of a practical system to motivate our proposed RF scheduler, although our explanations also apply equally to other wireless systems that dynamically adapt wireless channel rate in response to user sending rate.

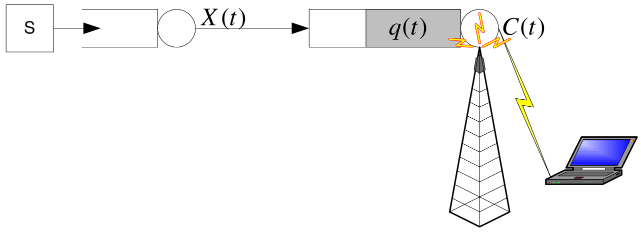


Figure 1: Illustration of a cellular hop

3.1 The CDMA2000 1xRTT System

Figure 1 depicts the wireless hop in a typical cellular network. It comprises of a base station, mobile devices, and a buffer at the base station for each user. The RF scheduling algorithm resides at the base station (or in the case of CDMA2000 1xRTT, the Base Station Controller) and determines the rate allocated to each mobile session. For the purposes of this work, we focus on one such mobile session that involves a TCP bulk transfer on the downlink.

According to CDMA2000 standards [2], whenever the buffer level for a particular user at the base station exceeds a threshold the scheduler dynamically increases its rate to a pre-specified higher rate by assigning a *supplemental channel*. The higher rate supplemental channel is achieved by reducing the Walsh code length, which reduces the spreading factor thus increasing data rate[6]. However, the high rate comes at the expense of increased interference (due to smaller code length) which results in higher frame error rates. For example, in CDMA2000 1xRTT, the *fundamental channel* has a data rate of 9.6 Kbps and a target FER of $1\% - 2\%$. In comparison the supplemental channel offers rates ranging from 19.2 Kbps to 153.6 Kbps but with a higher target FER of $5\% - 10\%$.

3.2 TCP-Aware RF Scheduler: Proposed System Model

The TCP-aware RF scheduler we consider resides at base station. It is quite similar in operation to the 1xRTT scheduler described above, with the exception that it assigns a channel rate based on the *user's TCP sending rate*[7] rather than buffer content. Specifically, whenever the TCP sending rate exceeds (or drops below) the current channel rate, the base station increases (decreases) the channel rate (accompanied by the corresponding trade-off with FER, signal power, *etc.*). If the user exceeds the maximum possible channel rate, the session experiences packet loss due to dropped packets. For analytical tractability, we assume that the scheduler can switch between at most *two* rates. Extend-

[6]This is also accompanied by a slight increase in signal power to combat fading, *etc.*

[7]This of course implies the assumption that the scheduler can measure TCP sending rate, which is defined in the next section.

ing the model to three or more rates is ongoing work. Also, we note that the scheduler considered in this work is applicable to a single TCP session. The issue of multiple users (sessions) is discussed in Section 6.

We present the operations of the scheduler as well as the assumptions regarding the system in more detail below:

1. The RF scheduler decides which of two channel rates C_0 and C_1 are to be assigned based upon the user's TCP sending rate. We assume $C_0 \leq C_1$. If the TCP sending rate is below C_0, the scheduler assigns a channel rate of C_0, otherwise it assigns C_1[8]. In other words:

$$C(t) = \begin{cases} C_0 & \text{if } X(t) \leq C_0 \\ C_1 & \text{otherwise} \end{cases}$$

where $X(t)$ is the TCP sending rate at time t and $C(t)$ is the assigned channel capacity. The motivation for such a scheduler was presented in Section 1. Intuitively, when the TCP sending rate is small, a lower frame error rate is essential (to avoid time-outs *etc.*). The scheduler can exploit the knowledge of the low TCP sending rate to trade-off channel capacity for channel integrity. At higher TCP sending rates, it is more appropriate to assign a larger channel capacity at the expense of a higher FER. This is because, even though packet loss probability due to channel errors increases, a larger channel capacity prevents packet loss due to congestion, which would have happened with probability *one* were capacity not increased, allowing TCP to transmit at high rates for a longer time.

2. The packet error probability is implicitly assumed to be a function of the assigned rate and denoted by $p_0(p_1)$ when the assigned rate is $C_0(C_1)$. This is an important feature representative of current wireless systems where an increase in channel rate typically comes at the cost of increased packet error probability. For simplicity, we shall refer to (p_i, C_i) together as a *state* or *mode*. We shall dwell in more detail on the relation between p and C in Section 5.

3. We assume the presence of power control to primarily combat fast fading and interference effects. This is true in current systems where fast closed loop power control tracks a specified target SINR (or equivalently target FER).

4. We assume no (or a very small) buffer at the base station. Hence, TCP experiences congestion if its sending rate exceeds the maximum channel rate (C_1).

We note that there are several other features of TCP, for example, timeout values, window size, sending state, *etc.* which could potentially be utilized to improve upon our proposed scheduler. Incorporation of such features however, would make the scheduler complex to implement as well as to study. Our aim here is to propose a system that involves minimal modifications to schedulers used in current technologies like CDMA2000 1xRTT, EV-DO, *etc.* and can be studied analytically. From this perspective, we believe that our proposal to incorporate knowledge of only TCP sending rate satisfies both goals.

[8]In practice a higher rate allocation may not always be possible due to heavy congestion scenarios or deep fading but is typically feasible in low load scenarios. This phenomenon can be easily and naturally incorporated in our model by modeling the *denial* of a higher channel as a random variable. We omit details due to space constraints.

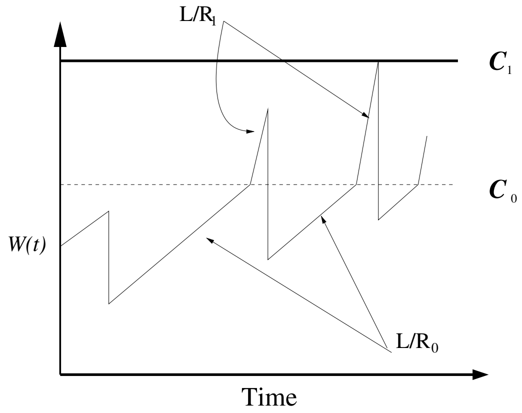


Figure 2: TCP window size evolution over a variable rate channel.

Another issue worth mentioning is that modern cellular systems have four or more *modes*, *i.e.*, they can support up to four or more different channel rates. However, obtaining succinct analytical expressions even for three is quite difficult. Hence, as a starting point we study two modes to demonstrate the impact of selecting $(C0, C1)$ on TCP performance.

Finally, we emphasize that at this stage, no specific assumptions have been made regarding how the two channel rates are achieved nor how they result in the specific channel error probabilities. Indeed the specific relation is not required in the TCP model and only the actual variables (p_i, C_i) are required. The channel rates and packet error probabilities are a function of the underlying technology that is used, *e.g.*, adaptive modulation, spreading, *etc.*. This issue is addressed in detail in Section 5.

4. RATE MODEL FOR VARIABLE CHANNEL

In the previous section, we presented the system model for a simple TCP-aware RF scheduler. Such a system results in two distinct operating regimes with different channel capacities, round trip times and packet error probabilities. In this section, we present our TCP model for such a system.

4.1 TCP Model

Existing TCP models typically assume that the Round Trip Time (RTT) and packet loss statistics are independent of TCP dynamics in throughput calculations. However, in the wireless environment considered in this work, there exists a strong correlation between the scheduler and TCP sending rate. In particular, the channel capacity C, which affects RTT, as well as packet loss probability p *are functions of the TCP sending rate*. As an illustration, Fig. 2 depicts the evolution of window size of TCP in steady-state when serviced by the proposed TCP-aware RF scheduler. The scheduler assigns rates based on the TCP sending rate and as can be seen, this in turn affects the window *growth* rates. In Section 5 we show that ignoring this dependency can result in large errors in throughput prediction.

In order to tackle the impact of the proposed RF scheduler on TCP throughput, we use the model developed by Baccelli *et al.* [3] for a *single fixed rate* as a starting point and develop a model that accounts for the two rate regime. We present a more detailed explanation of our model below:

1. We assume that the TCP version is TCP Reno and model the TCP window growth in steady state as a fluid process where the window size grows linearly in the absence of loss.

2. The sender is assumed to always have data to send and, for analytical tractability, we ignore time-outs and slow start.

3. Let $W(t)$ denote the window size of TCP at time t and $R(t)$ the round trip time at time t. In the absence of a buffer, if the scheduler is in mode $i = 0, 1$ at time t, we approximate the round trip time $R(t)$ as:
$$R(t) = R_i = a + L/C_i$$
where a is the propagation delay, L is the packet length and C_i is the channel capacity in mode i.

4. Let $X(t)$ denote the instantaneous TCP sending rate at time t in bits/sec. Then $X(t) = \frac{W(t)}{R(t)}$.

5. In congestion avoidance mode, the TCP window size increases by roughly one packet (L bits) every R_i seconds in mode i when there is no packet loss. We approximate this in our fluid model with a linear growth rate of L/R_i. Consequently, the sending rate grows at a linear rate of L/R_i^2 bits/sec^2 in the absence of loss. To see this note that the rate of increase of $X(t)$ is given by:
$$\frac{X(t+R_i) - X(t)}{R_i} = \frac{W(t+R_i) - W(t)}{R_i^2} = \frac{L}{R_i^2}\,.$$

6. As in [26, 3], we assume that the channel losses can be modeled by an inhomogeneous Poisson process with rate $p_i X(t)$, $i = 0, 1$ at time t.

7. If the TCP sender is not constrained by the receiver window, then TCP experiences congestion with probability one when its sending rate, $X(t)$, exceeds C_1. However, if TCP is constrained by the receiver window size, the sending rate stops increasing once it reaches the receiver advertised window and it experiences only channel related losses.

We need to approach the problem differently depending on whether the two modes (channel rates) satisfy either $C_1 \le 2C_0$ or $C_1 > 2C_0$. For the important case[9] of $C_1 \le 2C_0$ we derive expressions for mean TCP throughput for both, the unconstrained rate case as well as the constrained rate case in Sections 4.2 and 4.3 respectively. We have also obtained expressions for the TCP throughput for the case when $C_1 > 2C_0$ but omit it here due to lack of space as well as the fact that most performance improvements are obtained in the region $C_1 \le 2C_0$.

For ease of exposition, we define some notation.

[9] In practice, most wireless rates obey the relationship $C_1 = 2C_0$.

1. Let $f_i(x,t)$ denote the density function of rate $X(t)$ at time t in mode i. Then,
$$f_i(x,t)\Delta x = \mathbb{P}\left\{x \le X(t) \le x + \Delta x, C(t) = C_i\right\},$$
where $C(t)$ is the instantaneous channel rate at time t. It is clear from the above definition that
$$f(x,t) = f_0(x,t) + f_1(x,t)\,.$$

2. We also define the terms:
$$\delta_i = \frac{L}{R_i^2}, \quad \gamma_i = \frac{p_i}{L}, \quad \text{for } i = 0, 1\,.$$

4.2 Case I: The Rate Unconstrained Case

Recall that in the *rate unconstrained* case, the sender side TCP window size is not limited by the receiver. Hence, TCP experiences congestion loss, with probability one, whenever its sending rate $X(t)$ exceeds C_1.

Before proceeding with the analysis, it is worthwhile discussing an important aspect of the variable rate environment that directly affects the analysis, namely the behavior of the TCP sending rate at the *channel transition points.* Let at some time t^-, the TCP sending rate increase to $X(t^-) = C_0$. The corresponding window size is given by $W(t^-) = C_0 \cdot R_0$. As per the proposed policy, the scheduler would then assign a rate of C_1 to the TCP session at time t^+ resulting in a new RTT of R_1. Since TCP is a window-based protocol, the *window size* will be continuous at the rate transition point. Specifically, we have $W(t^+) = W(t^-) = C_0 R_0$. Consequently, the *new* sending rate would be given by
$$X(t^+) = \frac{W(t^+)}{R_1} = \frac{R_0 \cdot C_0}{R_1} = \frac{R_0}{R_1} X(t^-)\,.$$

In other words, the TCP sending rate experiences a discontinuous jump by a factor of $g = R_0/R_1$ when the channel capacity transitions from C_0 to C_1. Similar arguments can be used to show that if $X(t) \ge C_0$ and TCP experiences a loss, the sending rate drops by a factor $1/2g$. This aspect of the TCP sending rate must be accounted for in the analysis.

For purposes of analysis, we partition the range of the sending rate $X(t)$ into four different regions as shown in Fig. 3. The discontinuity in $X(t)$ when the channel rate transitions from C_0 to C_1 is clearly visible in the figure. An interesting observation worth mentioning is that because of the discontinuity, the sample path of $X(t)$ never resides in the region (C_0, gC_0). This can be contrasted with the *window size* evolution in Fig. 2 which has no discontinuities.

We use techniques from fluid analysis to develop forward differential equations for the probability density function $f_i(x,t)$. We briefly outline, as an example, the derivation of the forward difference equation for the first region $0 \le X(t) \le C_0/2$. Let at some time $t + \Delta t$, the sending rate $X(t+\Delta t)$ attains a value, say x. Then only one of *two* possible events could have occurred at time t. One, at time t the TCP sending rate is at value $x - (\frac{L}{R_0^2})\Delta t$ and it experiences no loss during the interval $(t, t+\Delta t]$, consequently growing by an amount $(\frac{L}{R_0^2})\Delta t = \delta_0 \Delta t$. Two, at time t, $X(t) = 2x$ and the sender experiences a loss event, resulting in the rate dropping to $2x/2 = x$. Combining these two disjoint events yields the following forward difference equation:
$$\begin{aligned} f_0(x, t+\Delta t)(\delta_0 \Delta t) = f_0(x - \delta_0 \Delta t, t)(1 - \gamma_0 x \Delta t)\delta_0 \Delta t \\ + f_0(x,t) 2\gamma_0 x (2\delta_0 \Delta t) \end{aligned}$$

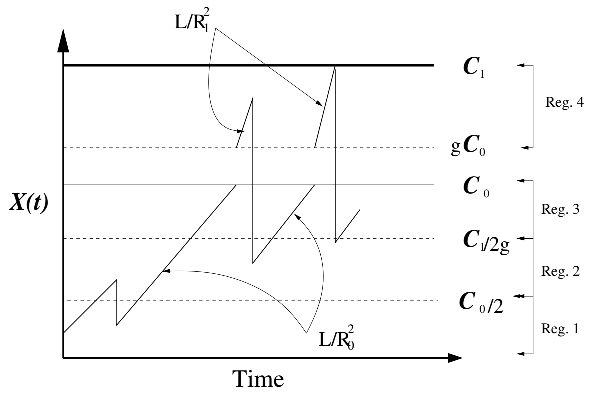


Figure 3: Regimes for $C_1 \leq 2C_0$

Similar equations can be written for the other three regions and we omit the details due to lack of space and refer the interested reader to [18]. Letting $\Delta t \to 0$, we can obtain a set of partial differential equations for $f_i(x,t)$. Under the assumption that $X(t)$ has a steady-state distribution in the limit $t \to \infty$, we obtain the following system of differential equations for the distribution $f_i(x)$, where we denote $\frac{d}{dx} f_i(x)$ by $\dot{f}_i(x)$:

1. $0 < x < C_0/2$

$$\begin{aligned} \delta_0 \dot{f}_0(x) &= -\gamma_0 x f_0(x) + 4\gamma_0 x f_0(2x) \\ f_1(x) &= 0 \end{aligned} \tag{1}$$

2. $C_0/2 < x < C_1/2g$

$$\begin{aligned} \delta_0 \dot{f}_0(x) &= -\gamma_0 x f_0(x) + 4g\gamma_1 x f_1(2gx) \\ f_1(x) &= 0 \end{aligned} \tag{2}$$

3. $C_1/2g < x < C_0$

$$\begin{aligned} \delta_0 \dot{f}_0(x) &= -\gamma_0 x f_0(x) \\ f_1(x) &= 0 \end{aligned} \tag{3}$$

4. $gC_0 < x < C_1$

$$\begin{aligned} f_0(x) &= 0 \\ \delta_1 \dot{f}_1(x) &= -\gamma_1 x f_1(x)\,. \end{aligned} \tag{4}$$

Equally important, we also obtain the following boundary conditions:

1. At $x = \frac{C_0}{2}$,

$$f_0((C_0/2)^+) = f_0((C_0/2)^-)\,.$$

2. At $x = \frac{C_1}{2g}$,

$$\frac{1}{R_0^2} \cdot f_0((C_1/2g)^+) = \frac{1}{R_0^2} \cdot f_0((C_1/2g)^-) + \frac{1}{R_1^2} f_1(C_1^-)\,.$$

3. At $x = C_0$,

$$\frac{1}{R_1^2} \cdot f_1((gC_0)^+) = \frac{1}{R_0^2} \cdot f_0(C_0^-)\,.$$

The above differential equations could be solved numerically in order to obtain the rate distribution $\{f_i(t)\}$. However, the actual quantity of interest for optimization is usually the mean TCP throughput, and fortunately analytical expressions for it can be obtained, far more easily by the use of Mellin transforms. They were previously used in similar settings by the authors of [4, 5, 3] and we shall follow their general approach.

Let $\widehat{f}(u)$, $u \geq 1$ be the Mellin transform of some probability distribution function $f(x)$ defined by

$$\widehat{f}(u) = \int_0^\infty f(x) x^{u-1} dx\,. \tag{5}$$

Then the mean of X denoted by $\overline{X}$ is simply given by $\overline{X} = \widehat{f}(2)$. Define ϕ_{ij} as follows

$$\phi_{ij} = \frac{\gamma_i}{\delta_j} = \frac{p_i R_j^2}{L^2}\,. \tag{6}$$

Multiplying Eqn. (1)-Eqn. (4) by x^u, integrating them over the respective limits and summing them yields, after some algebraic manipulations:

$$\begin{aligned} \widehat{f}_0(u) &= \frac{\phi_{00}}{u}(1 - \frac{1}{2^u})\widehat{f}_0(u+2) + \frac{V_0}{u}\psi(u) \\ \widehat{f}_1(u) &= V_0 \Delta(u) \end{aligned} \tag{7}$$

where,

$$\begin{aligned} \psi(u) = (g^2-1)(C_0)^u e^{-\frac{\phi_{11}}{2} g^2 C_0^2} - \Big((\frac{1}{2g})^u g^2 - 1 \Big) \\ \times (C_1)^u e^{-\frac{\phi_{11}}{2} C_1^2} - u\Delta(u) - (\frac{\phi_{10}}{2^u g^{u+1}} - \phi_{11})\Delta(u+2) \end{aligned}$$

and,

$$\Delta(u) = \int_{C_0}^{C_1} e^{-\phi_{11} x^2} x^u du$$

is the incomplete Gamma integral. To obtain Eqn. (7), we have also utilized the boundary conditions as well as the fact that Eqn. (4) can be directly solved to yield $f_1(x) = V_0 e^{-\phi_{11} x^2}$ for $gC_0 \leq x < C_1$.

By expanding the recursive relation in Eqn. (7), we obtain the following expression for $\widehat{f}_0(u)$:

$$\widehat{f}_0(u) = V_0 \sum_{k \geq 0} (\phi_{00})^k \Pi_k(u) \psi(u+2k), \tag{8}$$

where,

$$\Pi_k(u) = \frac{1}{u+2k} \prod_{i=0}^{k-1} \Big(\frac{1 - 2^{-u-2i}}{u+2i} \Big)\,.$$

It is left to find the unknown constant V_0. Substituting $u = 1$ in Eqn. (5), one easily obtains the normalization condition: $\widehat{f}(1) = \widehat{f}_0(1) + \widehat{f}_1(1) = 1$. Utilizing Eqn. (7) we have,

$$V_0 = \frac{1}{\Delta(1) + \sum_{k \geq 0} (\phi_{00})^k \Pi_k(1) \psi(1+2k)}\,. \tag{9}$$

Finally, the mean TCP throughput is given by $\widehat{f}(2) = \widehat{f}_0(2) + \widehat{f}_1(2)$ which can be expressed as

$$\overline{X} = \frac{\Delta(2) + \sum_{k \geq 0} (\phi_{00})^k \Pi_k(2) \psi(2+2k)}{\Delta(1) + \sum_{k \geq 0} (\phi_{00})^k \Pi_k(1) \psi(1+2k)}\,. \tag{10}$$

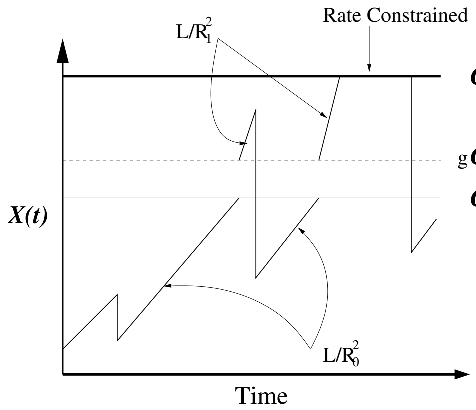


Figure 4: TCP Evolution when maximum advertised window $W_{max} = C_1 \cdot R_1$.

4.3 Case II: The Rate Constrained Case

In the previous sub-section we assumed that sending rate of the source was not constrained by the receiver. Hence the sender side window can grow till it exceeds the maximum bandwidth-delay product of the channel in which case it experiences a congestion loss. Equivalently, it experiences congestion whenever the sending rate exceeds the maximum channel rate. However, often the receiver advertises a window size W_{max} that is smaller than the peak bandwidth-delay product. Consequently, the sender side TCP window stops growing once it reaches this advertised window. Equivalently, the sending rate stops growing once it hits C_1[10]. In such a state, if we assume per-user isolation (which is common in modern cellular systems), the source does not experience congestion and the TCP window will drop only due to channel errors.

This behavior is shown in Fig. 4. From the figure it is clear that a density mass exists at C_1 and the stationary rate density distribution function $f(x)$ has a discontinuity at C_1. We let

$$f_1(C_1) = f_1(x)\mid_{x=C_1} = A\delta(x - C_1) \tag{11}$$

where A is some proportionality constant and $\delta(x)$ is the Dirac Delta function.

The discontinuity does not affect the differential equations. It does however affect the boundary conditions at $C_1/2g$ and C_1^- (the boundary condition at C_0^- remains unchanged). Solving for the first discontinuity by conditioning on $X(t + \Delta t) = C_1/2g$ yields,

$$\frac{1}{R_0^2} f_0((C_1/2g)^+) = \frac{1}{R_0^2} f_0((C_1/2g)^-) + \gamma_1 C_1 A \tag{12}$$

where, recall that $\gamma_1 = p_1/L$. Similarly, conditioning on $X(t + \Delta t) = C_1$, we have

$$f_1(C_1^-) = \frac{R_1^2}{L}\gamma_1 C_1 A\,. \tag{13}$$

Using Eqn. (13) we can re-write the boundary condition at $x = C_1/2g$ (Eqn. (12)) as

$$\frac{1}{R_0^2} f_0((C_1/2g)^+) = \frac{1}{R_0^2} f_0((C_1/2g)^-) + \frac{L}{R_1^2} f_1(C_1^-)\,. \tag{14}$$

This is essentially the same boundary condition as the one for the rate unconstrained case in Section 4.2 (the change is hidden in $f(C_1^-)$). Consequently, we can proceed in exactly the same manner as in Section 4.2 over the region $[0, C_1)$ to derive the mean throughput.

This yields, as before,

$$\widehat{f_0}(u) = V_0 \sum_{k\ge 0} (\phi_{00})^k \Pi_k(u)\psi(u + 2k),$$
$$\text{and } \widehat{f_1}(u) = V_0 \Delta(u)$$

where all the variables retain their original meanings defined in the previous sub-section.

The difference from the unconstrained rate case is that the normalization and throughput relations are now given by

$$\widehat{f_0}(1) + \widehat{f_1}(1) + \int_{C_1^-}^{C_1^+} A\delta(x - C_1)dx = 1$$
$$\text{or } \widehat{f_0}(1) + \widehat{f_1}(1) + A = 1 \tag{15}$$
$$\text{and } \overline{X} = \widehat{f_0}(2) + \widehat{f_1}(2) + \int_{C_1^-}^{C_1^+} xA\delta(x - C_1)dx$$
$$\text{or } \overline{X} = \widehat{f_0}(2) + \widehat{f_1}(2) + C_1 A \tag{16}$$

Through the direct solution of the differential equation in the region $gC_0 < x < C_1$ and the boundary condition at $x = C^-$, we can relate A and V_0 as follows.

$$f_1(C_1^-) = V_0 e^{-\phi_{11} C_1^2/2} = \frac{R_1^2}{L}\gamma_1 C_1 A,$$
$$\text{or} \quad V_0 = \frac{R_1^2}{L}\gamma_1 C_1 e^{\phi_{11} C_1^2/2} \cdot A\,. \tag{17}$$

For simplicity, denote $Z = (\frac{R_1^2}{L})\gamma_1 C_1 e^{\phi_{11} C_1^2/2}$. By plugging this relation and the expressions for $\widehat{f_0}(u)$ and $\widehat{f_1}(u)$ in the normalization relation, we derive the following expression for A:

$$A = \frac{1}{1 + Z(\Delta(1) + \sum_{k\ge 0}(\phi_{00})^k \Pi_k(1)\psi(1 + 2k))}\,.$$

The mean throughput can now be computed to be:

$$\overline{X} = \frac{C_1 + Z\,(\Delta(2) + \sum_{k\ge 0}(\phi_{00})^k \Pi_k(2)\psi(2 + 2k))}{1 + Z\,(\Delta(1) + \sum_{k\ge 0}(\phi_{00})^k \Pi_k(1)\psi(1 + 2k))}\,. \tag{18}$$

5. NUMERICAL RESULTS

In this section, we verify the accuracy of our model by comparison against *ns-2* simulations and then demonstrate how the model can be used to optimize resources on a wireless channel. Before evaluating the performance of the model, we must address the issue of how rate adaptation by the RF scheduler affects both channel capacity and packet error probability. Note that our model does not assume any specific dependencies between the various channel rates and

[10] For simplicity, we assume that the receiver advertised window is perfectly sized, that is $W_{max} = C_1 \cdot R_1$. Of course, our model also works when $W_{max} \le C_1 \cdot R_1$, in which case we simply choose a new peak capacity $C_1' = \frac{W_{max}}{R_1}$.

packet error probabilities, rather it assumes these are predetermined input variables.

As mentioned in Section 1, in current CDMA systems rate adaptation is achieved by changing any of three variables: error coding rate, the spreading factor or modulation scheme. In this work we study the impact of the first two variables, though our model is equally applicable to any other means of controlling the data rate. We initially assume that the channel rate and packet error are controlled *solely by changing the coding rate*. This is primarily to provide a simple basis to evaluate the accuracy of the model as well as showcase its utility. Section 5.1 presents the relationship we use to quantify the impact of coding rate on channel capacity and packet error probability. We use this relationship to: a) evaluate the accuracy of our model in Sections 5.2 and 5.4, and b) choose coding rates to maximize TCP throughput in Section 5.3.

Finally, in Section 5.5, we apply the model to scenarios where resource allocation is controlled by changing the *spreading factor*. This is representative of current CDMA networks where spreading factor is the dominant control knob to adapt data rates. We evaluate the gain in TCP throughput and resultant trade-off with energy consumption as a function of the spreading factor.

Another potential control factor in wireless networks is the retransmission mechanism typically deployed at radio link layer to mitigate high frame error rates by retransmitting erroneous frames. Our model can easily incorporate the impact of link-layer retransmissions as a trade-off between channel rate and packet error probability similar to the coding rate studied here, though, it cannot account for the latency introduced by link layer retransmissions. However, extensive measurements conducted over a commercial 1xRTT network by Mattar *et al.* [24], found that the impact of rate changes of the scheduler on TCP sending rate is far more dominant, while there is little or no correlation between TCP round-trip times and the link-layer retransmissions primarily due to large RTTs and very fast re-transmissions. Hence we do not study the impact of link layer re-transmissions in this work.

5.1 Packet Error Probability: The Variable Coding Case

As mentioned previously, we assume that the capacity C_i and packet error probability p_i in mode i are functions of the coding rate ρ_i. Hence, for a given bit error probability, the TCP throughput is a function of the two coding rates, *i.e.*, $\overline{X}(\rho_0, \rho_1)$. The relation between the capacity and coding rate is straightforward and is given by $C_i = \rho_i \cdot C^*$ where C^* is the uncoded channel capacity.

The packet error probability however is strongly dependent on not just the coding rate, but also the coding scheme used. Consequently, one must either choose a specific coding scheme, or resort to bounds on the achievable packet error probability for a given coding rate.

One such bound is the Gilbert-Varshamov [25] bound. This was used in [3] and we also use it as an approximation. The Gilbert-Varshamov bound is a bound on the parameters of a code of length B and information bit length K. It specifies that there exists a minimum Hamming distance d between any two codewords that must satisfy

$$2^B \le 2^K \sum_{j=0}^{d-1} \binom{B}{j} \tag{19}$$

where $d - 1 \le B/2$. Such a code can correct at most $t = \lfloor (d-1)/2 \rfloor$ errors. Hence, the above relation bounds the maximum number of correctable errors for any coding rate.

Let p_e denote the bit error probability for the wireless channel in consideration. Suppose that TCP packets have fixed size of L bits. The TCP packets are broken up into radio blocks of size B bits for transmission over the wireless channel. Each radio block is assumed to have K bits of information and $(B - K)$ bits for coding. Hence the coding rate is $\rho = K/B$. The packet error probability is

$$p = 1 - (1 - p_b)^{\lceil \frac{L}{\rho B} \rceil}, \tag{20}$$

where p_b is the radio block error probability. Using the Gilbert-Varshamov bound, for a given coding rate, we can determine the maximum number of error bits, t, that are correctable using the coding scheme with rate ρ. Then,

$$p_b = \sum_{j=t+1}^{B} \binom{B}{j} p_e^j (1 - p_e)^{B-j} . \tag{21}$$

As an example, in Fig. 5, we have plotted the packet error probability as a function of the coding rate ρ.

5.2 Adaptive Coding Evaluation: Rate Unconstrained Case

We begin our evaluation of the TCP model by comparing its accuracy against simulations of the TCP-aware RF scheduler implemented in *ns-2*. In this sub-section we study the case when the TCP sending rate is not limited by the receiver advertised window. To be concrete in this section, we set the raw channel rate C^* to 128 Kbps and the two-way propagation delay a to 200 ms. TCP packet size is set to 1024 bits. A TCP packet is divided into radio blocks of size 256 bits for transmission over the wireless channel.

The same parameters were also used for the *ns-2* simulation. TCP Reno was chosen (since the model fits that version best) and packet errors were assumed to be independent and identically distributed[11]. Since we do not account for time-outs in our model, we only simulated scenarios with large window size that result in few timeouts. Indeed, a previous study ([14]) has recommended the use of large window sizes in cellular networks to precisely avoid such time-outs due to bandwidth oscillations. The duration of each simulation run was 1000 seconds. The bit error probability was held constant for the duration of the simulation, which is true under perfect power control. Hence the packet error probability is solely a function of the coding rate. We ran 20 simulations with different random seeds for each data point and the results reported are in the 95% confidence interval.

It is worth mentioning that we limited ourselves to low packet error probabilities in all the scenarios. The reasons for this are two fold. First, TCP is known to perform well only for low packet errors (less than 5%) and hence it represents the region of interest. The second reason has to do with modeling the packet loss process as an inhomogeneous

[11]Typically in wireless channels, errors are bursty affecting multiple packets. From the perspective of our model, this simply shows up as a single loss event.

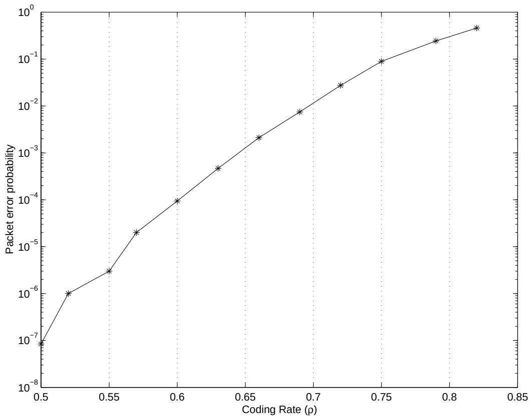


Figure 5: Packet error probability p as a function of the coding rate ρ for a bit error probability of $p_e = 10^{-2}$.

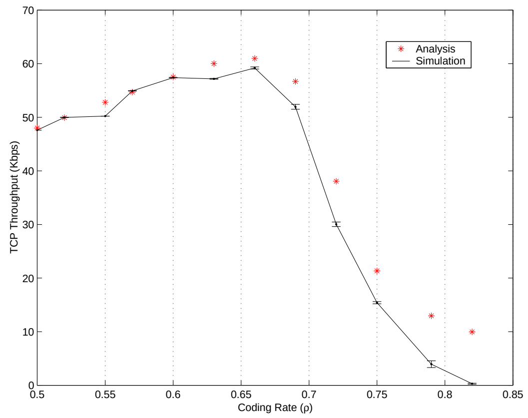


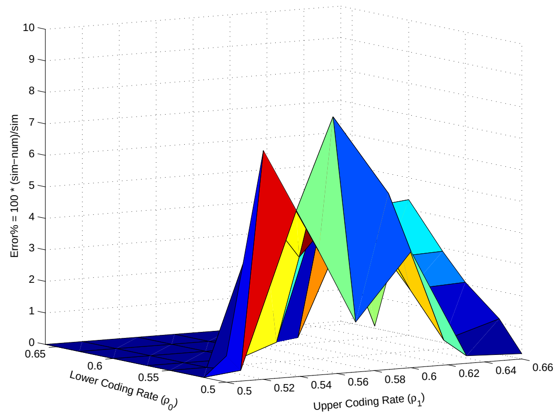
Figure 6: *Single-Rate case*: TCP throughput for a bit error probability of $p_e = 10^{-2}$.

Poisson process which is reasonable only for low packet error probabilities.

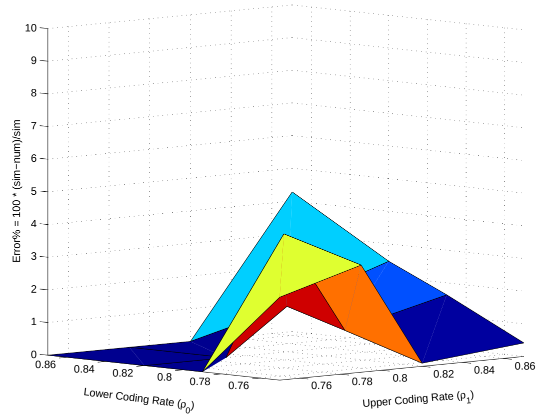
To begin, we show the results of comparison of the model and simulations for the special case of $\rho_0 = \rho_1$, *i.e.*, there is only a *single rate*. In this case, our model simplifies to the scenario considered in [3] where a *single* static coding rate is utilized. We plot the packet error probabilities as a function of the coding rate in Fig. 5 for a target bit error probability of 10^{-2} and the corresponding throughputs of both the model and simulation as a function of the coding rate in Fig. 6. One observes the close agreement between the model and simulation. The exact nature of the curves are discussed in the next section. We also studied the performance for target bit-error probabilities of $10^{-1}, 10^{-3}$ and 10^{-4} but omit them since the results were similar.

Figs. 7(a) and 7(b) present results for the more general case when ρ_0 and ρ_1 are different. They show the % error between the model and simulation across all feasible[12] coding rates (ρ_0, ρ_1) on a three dimensional grid for two different bit error probabilities $(10^{-2}, 10^{-3})$. Observe again that the model matches the simulation results closely with errors typically less than 5%.

[12]By "feasible", we imply $\rho_0 < \rho_1$, $\rho_1 \leq 2\rho_0$ and $p_i \ll 1$.



(a) $p_e = 10^{-2}$.



(b) $p_e = 10^{-3}$.

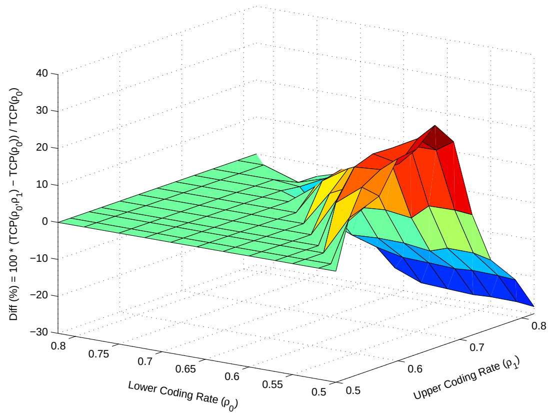
Figure 7: *Two-Rate case* TCP throughput: % difference between model and simulation.

To demonstrate the importance of capturing the correlation between TCP and the scheduler as well as the presence of two states, we plot in Fig. 8 the % difference between the two rate model (which we have shown above to be accurate) and a *single rate* model that takes only *one* set of parameters, either those due to coding rate ρ_0 or those due to ρ_1 into consideration. We observe that there exist regions where there is substantial difference in predicted throughput. Hence a single rate model may not be always feasible to tune performance in such systems.

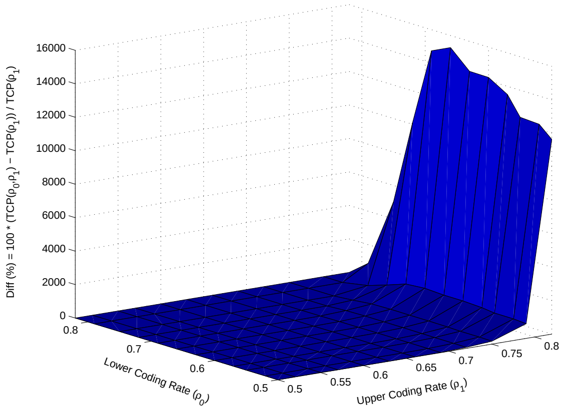
5.3 Optimal Coding Rates

We next turn our attention to the determination of the coding rates that maximize TCP throughput. Recall that in our model for adaptive resource allocation, the scheduler switches between two modes depending on the TCP sending rate. If we assume that the allocation is controlled through the coding rate, then the scheduler is essentially switching between two coding rates ρ_0 and ρ_1.

Clearly, the particular choice of ρ_0 and ρ_1 affects the achieved throughput. Intuitively, if packet error probability is close to 0 then increasing the coding rate increases throughput as well. The reason is that, in this situation, channel rate increases as a linear function of the coding rate whereas the packet error probability remains negligible re-



(a) $p_e = 10^{-2}$. % Deviation of Single-Rate model with parameters (C_0, p_0, R_0).



(b) $p_e = 10^{-2}$. % Deviation of Single-Rate model with parameters (C_1, p_1, R_1).

Figure 8: Comparison between Two-Rate and Single-Rate model.

sulting in increased throughput. However, if the packet error probability is large, then increasing the coding rate decreases throughput and eventually reduces it to 0. Consequently, one expects the existence of a coding rate that maximizes TCP throughput. This behavior is clearly evident in Fig. 6 where the TCP throughput initially increases as the coding rate increases, because the increase in capacity is far more than the increase in packet error probability, and then decreases. Similar curves have also been previously obtained in [3] for finite capacity and [7, 22] for infinite capacity models.

We extend these previous results by identifying the optimal *pair of coding rates* within the framework of an adaptive TCP-aware RF scheduler. Table 1 presents the pair of optimal coding rates $\tilde{\rho}_0$ and $\tilde{\rho}_1$ obtained from both, the model and simulations, that maximize the TCP throughput $\overline{X}(\rho_o, \rho_1)$ and the corresponding throughput for different target bit error probabilities. Observe that in most cases, there is a close match between the optimal coding rates (and corresponding throughput) obtained from the model and those from simulations. In order to quantify the benefits of using two coding rates, Table 2 presents the relative increase in throughput as a consequence of using adaptive coding compared to a single coding rate for each bit error

	Analysis		Simulation	
p_e	$(\tilde{\rho}_0, \tilde{\rho}_1)$	Thr. (Kbps)	$(\tilde{\rho}_0, \tilde{\rho}_1)$	Thr.(Kbps)
10^{-1}	$0.125, 0.125$	10.85	$0.125, 0.125$	8.89
10^{-2}	$0.630, 0.720$	65.19	$0.600, 0.660$	65.66
10^{-3}	$0.820, 0.910$	85.21	$0.820, 0.870$	85.35
10^{-4}	$0.910, 0.960$	91.99	$0.910, 0.960$	92.83
10^{-5}	$0.960, 0.990$	94.65	$0.960, 0.990$	94.91

Table 1: Optimal adaptive coding rates ($C^* =$ 128 Kbps).

p_e	Analysis	Simulation
10^{-1}	0%	0%
10^{-2}	6.9%	10.9%
10^{-3}	4.2%	9.1%
10^{-4}	5.3%	9.0%
10^{-5}	2.7%	2.4%

Table 2: Gain of adaptive coding.

probability. The gain factor is defined as

$$\text{gain} = 100 \times \frac{\overline{X}(\tilde{\rho}_0, \tilde{\rho}_1) - \overline{X}(\tilde{\rho})}{\overline{X}(\tilde{\rho})} \tag{22}$$

where $\tilde{\rho}$ represents the optimum coding rate for the single rate case. Hence, for each bit error probability, the gain factor was computed by comparing throughput at the *optimal* coding rates for adaptive and static coding respectively. In all the cases, we see that the optimum throughput with adaptive coding is *at least* equal to that of static coding and higher in several situations, with improvements of up to 10%.

Intuitively, the reason adaptive coding yields a gain in throughput over static coding even with the same target bit error probability is that the RF scheduler with adaptive coding exploits knowledge of TCP sending rate. Specifically, when TCP has a small window, it sends at a low rate. Hence, the RF scheduler can offer a smaller rate to the source but with a lower error probability. As the rate increases, the scheduler switches to a higher rate to cope with TCP. This is not possible in the single rate case.

5.4 Adaptive Coding Evaluation: Rate Constrained Case

We have also evaluated the efficacy of our TCP model for the case when the sending rate is constrained by the receiver advertised maximum window size (which translates into a maximum advertised rate). We used the same parameters as in Section 5.2 to compare the accuracy of our model against *ns-2* simulations. The percentage difference between the model and simulations for the *two rate* case was found to be of the same order as the rate unconstrained case, less than 7%, and hence not shown here. Similar results were also obtained for the *single rate* case, *i.e.*, $\rho_0 = \rho_1$.

5.5 Rate Adaptation: Processing Gain Control

In the previous sub-sections we assumed that the tradeoff between the channel capacity and frame error rate was controlled by the error coding rate. Next, we study scenarios that are more representative of current CDMA cellular networks. In CDMA networks, the dominant RF control variable that governs this trade-off is the *spreading factor*. The spreading factor is defined as the ratio of the CDMA chip rate to the actual *data rate* and relates the channel capacity and the bit error probability in the following manner. Let W denote the *chip rate* of the CDMA system and r_i the spreading factor allocated in mode i. Then the data rate C_i and "Bit Energy to Noise Ratio" are given by:

$$C_i = \frac{W}{r_i}, \qquad \frac{E_b^i}{I_0} = \frac{r_i \cdot E_c}{I_0},$$

where E_c is the *chip energy* of the data signal and I_0 the wide-spectrum interference. It should be clear from the above relations that as we *decrease* r_i, the channel capacity C_i increases, but the SINR, which directly affects the bit error probabilitydecreases.

Typically, the bit error probability is extremely sensitive to SINR and hence, in practice, some amount of power control is required to prevent a steep rise in the error probability when the spreading factor is decreased. Specifically, the chip energy is "boosted" by a small factor that is a function of the spreading rate to prevent a sharp drop in the quality of service. Consequently, the relationship between the SINR and the spreading factor r_i is modified slightly to:

$$\frac{E_b^i}{N_0} = \frac{E_c \cdot r_i E(r_i)}{I_0}, \tag{23}$$

where $E(r_i)$ is a decreasing function of r_i and represents the factor by which the original chip energy E_c is boosted. Hence, in addition to the spreading factor, one can also control the bit error probability by an appropriate choice of the function $E(r)$, which we shall denote as the *energy profile*.

We now demonstrate how our model can be used to study the impact of rate adaptation in such an environment. Since signal power is also a resource in the above framework, we must look at both, the mean TCP throughput as well as *energy consumption*. To quantify the latter, we use the notion of *normalized energy consumption*. It represents the average energy spent per bit as a function of TCP throughput and is computed as follows. The energy spent per bit in mode i is given by $E_b^i = E_c \cdot (r_i E(r_i))$. Further, assume that the mobile session spends a fraction of time π_i in mode i and achieves an average throughput $\overline{X}_i$ in that mode. Then the normalized energy consumption is given by:

$$\begin{aligned} E_{norm} &= \frac{1}{\overline{X}} \sum_i \pi_i \overline{X}_i \cdot E_b^i = \frac{1}{\overline{X}} \sum_i \widehat{f}_i(2) E_b^i \\ &= E_c \frac{1}{\overline{X}} \cdot \sum_i \widehat{f}_i(2) r_i E(r_i) \,. \end{aligned} \tag{24}$$

In our experiments, we set the chip rate W to 1.2288 Mchips/sec and the basic pilot RSSI (E_c/I_0) to (-7) dB based on CDMA2000 standards. We assume QPSK modulation (also a CDMA2000 standard) and a fixed coding rate of 0.6. The spreading factor r_i was allowed to take values from the set $\{4, 12, 16\}$, which gives a rate set of $\{76.8, 102.4, 153.6\}$ Kbps. We note that 78.6 Kbps and 153.6 Kbps are the two highest data rates available in CDMA2000 1xRTT with Radio Configuration Type 3 [2].

We tested different *energy profiles*, $E(r)$, and present results for two of them, denoted by E_1 and E_2. To give a sense of the numbers, Table 3 and Table 4 present the results for the single rate and two rate cases, respectively, with the energy profile E_1. The first column in each table represents the channel rates (and hence spreading factors) used. The last column presents the energy consumption for each configuration, measured in multiples of E_c. The *optimal configuration* of spreading factors, which was chosen based on the *maximum throughput*, is marked in bold for both the single and the two rate cases. The gain in TCP throughput and difference in energy consumption between the adaptive and static cases, under their respective *optimal configurations*, are shown in Table 5 for both energy profiles. The energy savings is computed as:

Energy Profile	Analysis		Simulation	
	Rate Gain	Eng. Sav.	Rate Gain	Eng. Sav.
E_1	10.8%	−3.5%	16.8%	−3.5%
E_2	15.8%	−4.8%	24.8%	4.2%

Table 5: Adaptive vs. static spreading: comparison for max rate configuration.

$$\text{Energy Savings} = 100 \times \frac{E_{norm}^{static} - E_{norm}^{adaptive}}{E_{norm}^{static}} \,.$$

There are a number of interesting observations regarding the results. From Table 5, the analytical model indicates that a small increase in energy consumption, on the order of $3 - 4\%$, in conjunction with a two rate strategy boosts TCP throughput by $10 - 15\%$ as compared to the single rate scenario even though the allowed *peak* channel capacities are the same. The rate gain and energy savings predictions of the simulation are comparable to those of the model for E_1 and for rate gain in E_2, but predicts higher benefits than the model for energy savings in the case of E_2. In fact, it predicts a *positive* energy savings of 4.5%.

The differences in energy prediction for E_2 are caused due to the model and simulation picking different *optimal configurations* in the single rate case, while yielding the same configuration in the two rate case. Specifically the model overestimated (by about 9%) the TCP throughput in the single rate case for the highest rate configuration of 153.6 Kbps causing it to pick that as the optimal. In comparison the simulation chose the 102.4Kbps rate which has higher energy consumption.

6. CONCLUSION

In this paper, we proposed a two-state TCP-aware scheduler as a means to improve TCP throughput on a wireless channel that can vary its rate dynamically and analyzed the performance of such a system. Our proposed system adapts its channel rate in response to the TCP sending rate allowing it to trade-off channel rate and FER in a "TCP friendly" way. We developed an analytical model to study TCP throughput in such a system that captures several of its salient features, in particular the interaction between TCP and the scheduler, as well as the presence of two distinct regimes in terms of channel rate, packet error probability and round trip times. The accuracy of the analysis was confirmed by simulations with *ns-2*. To demonstrate the utility of our model we applied it do maximize TCP throughput in scenarios where different RF control variables are used. In particular we explored optimization of coding rates as well as the spreading factor. Throughput improvements on the order of $15 - 25\%$ were observed as compared to previous cases where only a single coding rate was assumed.

In the future, we hope to extend our current formulations to cover the following important cases:

C Kbps	Analysis		Simulation	
	Thr. (Kbps)	Energy Consumption ($\cdot E_c$ J/bit)	Thr. (Kbps)	Energy Consumption ($\cdot E_c$ J/bit)
76.8	56.946	13.814	56.784	13.8144
102.4	75.098	13.745	75.8112	13.7454
153.6	**88.594**	**12.83**	**83.195**.8	**12.829**

Table 3: Spreading factors: Single-Rate case ($C^* = 153.6$ Kbps) Energy Profile E_1.

(C_0, C_1) Kbps	Analysis		Simulation	
	Thr. (Kbps)	Energy Consumption ($\cdot E_c$ J/bit)	Thr. (Kbps)	Energy Consumption ($\cdot E_c$ J/bit)
$76.8, 102.4$	74.545	13.77	74.912	13.77
$76.8, 153.6$	89.83	13.08	88.185	13.12
$\mathbf{102.4, 153.6}$	**98.180**	**13.26**	**96.939**	**13.284**

Table 4: Spreading factors: Two-Rate case ($C^* = 153.6$ Kbps) Energy Profile E_1.

- *Multiple channel rates:* Extend our model to incorporate more than two transmission rates. Current cellular networks allow at least four transmission rates.
- *Multiple TCP flows:* Two scenarios can be considered where (1) multiple flows are destined to a single user, and, (2) multiple flows are destined to multiple users. The first case fits in our model if the multiple flows are aggregated at a splitting point and transmitted over the wireless channel as a single TCP flow [11]. Moreover, in both cases, if individual flows are isolated from each other, by means of appropriate scheduling or channel allocation policies (e.g strict round-robin), then our results still apply. Otherwise, our model should be extended to address the following questions:
  1. What are the optimal channel rates for the RF scheduler?
  2. How should the channel be shared among various TCP flows (users)?
  3. What is the optimal strategy that maximizes TCP throughput by jointly optimizing both the channel rates (1) and the scheduling policy (2)?

7. REFERENCES

[1] E. Altman, C. Barakat, and V. M. R. Ramos. Analysis of AIMD protocols over paths with variable delay. In *Proc. IEEE INFOCOM*, Hong Kong, Mar. 2004.

[2] T. I. Association. TIA EIA IS-2000. Available at www.tiaonline.org/standards/sfg/imt2k/cdma2000/.

[3] F. Baccelli, R. Cruz, and A. Nucci. CDMA channel parameters maximizing TCP throughput. In *Proc. Workshop on Information Theory and its Applications*, Feb. 2006.

[4] F. Baccelli and K. B. Kim. TCP throughput analysis under transmission error and congestion losses. In *Proc. IEEE INFOCOM*, Hong Kong, Mar. 2004.

[5] F. Baccelli, K. B. Kim, and D. D. Vlleschauwer. Analysis of the competition between wired, DSL and wireless TCP flows in an access network. In *Proc. IEEE INFOCOM*, Mar. 2005.

[6] H. Balakrishnan, S. Seshan, and R. H. Katz. Improving reliable transport and handoff performance in cellular wireless networks. *Wireless Networks*, 1(4):469–481, 1995.

[7] C. Barakat and E. Altman. Bandwidth tradeoff between TCP and link-level FEC. *Computer Networks*, 39(5):133–150, 2002.

[8] C. Barakat, E. Altman, and W. Dabbous. On TCP performance in an heterogeneous network: A survey. *IEEE Communications Magazine*, pages 40–46, Jan. 2000.

[9] D. Barman, I. Matta, E. Altman, and R. E. Azouzi. TCP optimization through FEC, ARQ and transmission power trade offs. In *Proc. WWIC*, Frankfurt, Germany, Feb. 2004.

[10] C. Casetti, M. Gerla, S. Mascolo, M. Y. Sanadidi, and R. Wang. TCP Westwood: end-to-end congestion control for wired/wireless networks. *Wireless Networks*, 8(5):467–479, 2002.

[11] R. Chakravorty, S. Katti, J. Crowcroft, and I. Pratt. Flow aggregation for enhanced TCP over wide-area wireless. In *Proc. IEEE INFOCOM*, San Francisco, USA, Mar. 2003.

[12] M. C. Chan and R. Ramjee. TCP/IP performance over 3G wireless links with rate and delay variation. In *Proc. ACM MOBICOM*, Atlanta, USA, Sep. 2002.

[13] M. C. Chan and R. Ramjee. Improving TCP/IP performance over third generation wireless networks. In *Proc. IEEE INFOCOM*, Hong Kong, Mar. 2004.

[14] E. Chaponniere, S. Kandukuri, and W. Hamdy. Effect of physical layer bandwidth variation on TCP performance in CDMA2000. In *Proc. IEEE VTC*, Korea, Apr. 2003.

[15] A. Chockalingam, E. Altman, J. V. K. Murthy, and R. Kumar. Cross-layer design for optimizing TCP performance. In *Proc. IEEE ICC*, Seoul, Korea, May 2005.

[16] A. Chockalingam and G. Bao. Performance of TCP/RLP protocol stack on correlated rayleigh fading ds-cdma links. *IEEE Trans. Veh. Technol.*, 49(1):28–33, 2000.

[17] H. Elaarag. Improving TCP performance over mobile networks. *ACM Comput. Surv.*, 34(3):357–374, 2002.

[18] M. Ghaderi, A. Sridharan, H. Zang, D. Towsley, and R. Cruz. TCP-aware resource allocation in CDMA networks. Technical Report *RR06-ATL-030566*, Sprint Advanced Technology Labs, Mar. 2006.

[19] T. Goff, J. Moronski, D. S. Phatak, and V. Gupta. Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments. In *Proc. IEEE INFOCOM*, 2000.

[20] K. L. Gray and D. L. Noneaker. The effect of adaptive-rate coding on TCP performance in wireless communications. In *Proc. EUROCOMM*, Munich, Germany, May 2000.

[21] T. Klein, K. Leung, and H. Zheng. Improved TCP performance in wireless IP networks through enhanced opportunistic scheduling algorithms. In *Proc. IEEE Globecom*, Nov. 2004.

[22] B. Liu, D. L. Goeckel, and D. Towsley. TCP-cognizant adaptive forward error correction in wireless networks. In *Proc. IEEE Globecom*, Taipei, Taiwan, Nov. 2002.

[23] R. Ludwig and R. H. Katz. The Eifel algorithm: Making TCP robust against spurious retransmissions. *ACM SIGCOMM Comput. Commun. Rev.*, 30(1):30–36, 2000.

[24] K. Mattar, A. Sridharan, H. Zang, I. Matta, and A. Bestavros. On the interaction of TCP and the wireless channel in CDMA2000 networks. Technical Report RR06-ATL-052666 , Sprint Advanced Technology Labs, May 2006.

[25] F. J. McWilliams and N. J. A. Sloane. *The Theory of Error-Correcting Codes*. North-Holland, 1977.

[26] V. Misra, W. B. Gong, and D. Towsley. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED. In *Proc. ACM SIGCOMM*, Aug. 2000.

[27] J. P. Singh, Y. Li, and N. Bambos. Channel state awareness based transmission power adaptation for efficient TCP dynamics in wireless networks. In *Proc. IEEE ICC*, May 2005.

[28] M. Zorzi, A. Chockalingam, and R. R. Rao. Throughput analysis of TCP on channels with memory. *IEEE J. Select. Areas Commun.*, 18(7):1289–1300, 2000.