

TCP Performance Dynamics and Link-Layer Adaptation Based Optimization Methods for Wireless Networks

Jatinder Pal Singh, Yan Li, Nicholas Bambos, Ahmad Bahai, Bangnan Xu, and Gerd Zimmermann

Abstract—Almost a decade long research on the performance of TCP in wireless networks has resulted in many proposals and solutions [1]–[9] to the problem of TCP throughput degradation. Several of these measures, however, have their share of drawbacks. With the continuing emergence of wireless technologies ever since the work on TCP performance over wireless began, smart link-layer mechanisms like adaptive modulation and coding, power control, and incremental redundancy have been designed and deployed. In this work, we outline a cross-layer optimization framework based on the congestion control dynamics of a bulk-transfer TCP flow and demonstrate its application to networks which offer link-layer adaptive measures. We begin by observing that the TCP's congestion window dynamics are comprised of certain recurring patterns which we term as *cycles*. We then overlay a TCP throughput optimization methodology that selects link-layer transmission modes (e.g. modulation scheme, coding rate, transmission power, or a combination thereof) in accordance with TCP dynamics and wireless channel conditions. We provide insights into the working of the optimization procedure which protects TCP segments against losses on the wireless channel when the TCP congestion window size (in bytes) is below the bandwidth-delay product of the network. The protection against wireless channel losses is rendered by the link-layer by employing robust modulation and coding schemes, high transmission power, etc. We show that TCP dynamics aware link adaptation measures lead to substantial enhancement of TCP throughput in EGPRS and IEEE 802.11a networks.

Index Terms—Adaptive coding, adaptive modulation, dynamic programming, land mobile radio cellular systems, power control, transport protocols, wireless LAN.

I. INTRODUCTION

THE problem of TCP source throttling in wireless scenarios has been a research focus for over a decade now. The solutions that have been proposed over the years have their

Manuscript received September 11, 2005; revised July 3, 2006; accepted July 28, 2006. The associate editor coordinating the review of this paper and approving it for publication was W. Liao.

J. P. Singh is with Deutsche Telekom Laboratories, Ernst-Reuter-Platz 7, 10179 Berlin, Germany (e-mail: jatinder.singh@telekom.de).

Y. Li is with Qualcomm Inc., 675 Campbell Technology Parkway, Campbell, CA 94304 USA (e-mail: yanli@qualcomm.com).

N. Bambos is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: bambos@stanford.edu).

A. Bahai is with National Semiconductor, Stanford University, and University of California, Berkeley, CA USA (e-mail: bahai@stanford.edu).

B. Xu and G. Zimmermann are with T-Systems, Deutsche Telekom Allee 7, D-64295 Darmstadt, Germany (e-mail: bangnan.xu@t-systems.com; gerd.zimmermann@t-systems.com).

Digital Object Identifier 10.1109/TWC.2007.05678.

share of advantages and drawbacks. The end-to-end schemes [1]–[4] preserve TCP's semantics but require modifications to TCP implementation. The split connection approaches [5], [6] are marred by increased processing overheads, violation of end-to-end semantics of TCP, and slow, complicated handoffs. Enhanced link-layer reliability via link-layer designs that are TCP unaware cannot efficiently shield TCP from the wireless losses and is also associated with increased rate and delay variability [7]. Approaches on the line of the SNOOP protocol [1] represent a TCP aware link-layer design. However, the protocol suffer from overhead of cache maintenance, limited applicability, and inability to function when TCP headers are encrypted and when handoffs are being executed. None of the aforementioned solutions to TCP performance degradation leverage adaptive wireless system design features like FEC, transmission power, and existence of multiple transmission modes. Some recent efforts to examine adaptive link-layer measures include [8], [9], where the authors adopt standard steady state TCP throughput expressions and perform optimization via adaptive FEC, ARQ and power control. The TCP window evolution dynamics and congestion events are not considered in the work. We argue that for performance optimization, the link-layer needs to be adaptive to the instantaneous dynamics of a TCP flow.

In [10] we model TCP's congestion avoidance dynamics and evaluate adaptive power control measures for throughput enhancement in a simplified scenario. In this work, we explore the adaptiveness of link-layer mechanisms in a wireless system for optimization of TCP throughput. We present a generalized cross-layer optimization framework based on TCP dynamics and demonstrate its utility via adaptive power control and link adaptation. We show that the optimization framework results in a significant TCP throughput enhancement for EGPRS and IEEE 802.11a networks. To the best of our knowledge, the work presented in this paper is the first attempt that investigates link-layer adaptation to the congestion control dynamics of a TCP flow under varying wireless channel conditions.

The rest of this paper is structured as follows. We begin by analyzing in Section II, the performance dynamics of a bulk-transfer TCP flow. In Section III we outline an optimization framework based on TCP dynamics. We then demonstrate in Section IV, TCP performance enhancement via the proposed optimization measures and adaptive link-layer techniques. We

conclude the work and provide directions for future research in Section V.

II. CONGESTION CONTROL DYNAMICS OF TCP

The dynamics of a steady state bulk-transfer TCP flow can be described in terms of TCP window size evolution and the congestion events. The slow start and the congestion avoidance algorithms [11] determine the evolution of congestion window. The Timeout (TO) and Triple Duplicate (TD) indications [12], [13] characterize the congestion (or loss) events of TCP. In our work, we model the slow start and congestion avoidance algorithms with maximum window size limitation and consider both TD and TO congestion events. As most of the Reno implementations today have been rendered obsolete by SACK/NewReno deployment, we assume that TCP is able to recover from multiple segment losses in a window in the event of TD loss indication. A TCP flavor may implement its own fast recovery [14] process, for instance, the one assisted by SACK options [15] (TCP SACK) or partial ACK mechanisms [16] (TCP NewReno): we do not specifically model any particular one in our work. We also do not consider TCP's delayed ACK mechanism [17] for reasons of clarity of presentation, but the modeling in the present work can be easily extended to incorporate them.

We observe that any given trace of TCP's window size evolution comprises of recurring patterns which we term as *cycles*. Two kinds of cycles are identified: the *CABegin* cycles beginning in the congestion avoidance (CA) phase (Fig. 1) and the *SSbegin* cycles beginning in the slow start (SS) phase (Fig. 2). Each of these cycles can end either due to TO loss (Figs. 1(b) and 2(b)) or a TD loss (Figs. 1(a) and 2(a)) indication. While the *CABegin* cycle set consists of *CABegin* cycles beginning with all possible initial window sizes, the *SSbegin* cycle set has cycles with different initial window sizes and slow start thresholds [11].

III. OPTIMIZATION FRAMEWORK

We delineate a methodology to relate transport layer dynamics to wireless channel conditions and network congestion. An instance of TCP operation could be a mobile station communicating with a remote server (or vice versa), with wireless-hop connectivity between the mobile host and AP/BS. The progression of *SSbegin* and *CABegin* TCP cycles introduced in the previous section is characterized in terms of *rounds* (defined for a wireline case in [13]). The beginning of a round is marked by the commencement of back-to-back transmission of a window of segments. A round completes at a time which is the larger of the termination time of window transmission and time of arrival of an ACK for a segment in the window. The duration of a round can vary depending on the window size, the variation in transmission time of the frames on the wireless channel and also the Round Trip Time (RTT) in wireline domain. An instance of window size evolution with rounds is shown for a *CABegin* cycle in Fig. 3.

Fig. 4 depicts the framework for optimization based on the dissection of TCP operation into cycles. The dynamics of a round in slow start or congestion avoidance phase are shown in Fig. 4(a). In a round, a cost is incurred for the transmission

of a window of segments. This cost which may be attributed to transmission power, transmission time, etc., is represented by $C(r)$.

The total cost to be incurred (called the cost to go) starting from round r until the end of the cycle is represented by $J(r)$. In case a segment in the round is lost, a cost $C_{Loss}(r)$ is incurred in addition to $C(r)$. If the loss results in a TD indication, the cycle ends at round $r + 1$ (e.g., round $R + 1$ in Fig. 3) that involves the transmission of a window of segments following the segment for which TD indication occurred. $C_{Loss}(r)$ for the TD indication case is thus ascertained as the terminal cost of the cycle, which is evaluated as the negative of the throughput achieved during the cycle. This formulation of the terminal cost enables a cost minimization objective function to prefer high cycle throughputs owing to greater associated cost deduction. In the event that the loss indication is a timeout, $C_{Loss}(r)$ represents the cost incurred during the Timeout Phase. The cost of the timeout phase in Fig. 4(a) is given by $T(r + 2, r)$ where $T(k, r)$ ($k > r + 1$) represents the cost to go from round k given that the timeout occurred due to the loss indication of a segment in round r . Note that as in the case of TD segment loss in round r , round $r + 1$ in TO loss indication case involves the transmission of a segment window following the first lost segment. $T(k, r)$ is depicted in Fig. 4(b). During round k in timeout phase, the cost corresponding to segment transmission is termed as $C_{seg}(k)$. In case the segment transmitted during round k is successfully ACKed, the timeout phase and the cycle ends with a termination cost C_{TO} which is the negative of the throughput achieved during the cycle.

The optimization approach for the above discussed framework is motivated by Dynamic Programming (DP) [18] principles and targets minimization of cost-to-go of a cycle. We will reuse the cost symbols in Fig. 4 and formulate detailed optimization equations for TCP throughput. We let $\vec{\gamma}$ represent the channel condition vector corresponding to the wireless channel states encountered by the segments in a round and \vec{c} represent the network congestion states experienced by the segments. The dimension of $\vec{\gamma}$ depends on the number of segments transmitted in a round, the number of radio blocks required to transmit the segments, and the channel variation pattern during these transmissions. The congestion vector \vec{c} has a dimension equal to the number of segments in a round. The quantities $\vec{\gamma}$ and \vec{c} together ascertain the success of TCP segments in a round. We assume a reliable TCP ACK delivery in our work. The success probability vector of segments in the round is denoted by \vec{s} and the success probability of a round is expressed as $S_{rnd}(\vec{s})$. $J(r)$ is then represented in (1) as the cost-to-go minimized over all possible success probability vectors. The cost-to-go (C_{tg}) for given $\vec{\gamma}$ and \vec{c} is given by (2), with C_{Loss} , the terminal cost incurred on the loss of one or more segments in round r expressed as (3). $\vec{\gamma}$, \vec{c} , and \vec{s} respectively denote the vectors of channel state, congestion conditions, and success probabilities for the segments in round $r + 1$, when there is a segment loss in round r . The probability of a TD loss indication is represented as $p_{TD}(r, \vec{s}, \vec{s}')$. It is evaluated as the probability of three or more segments being successful amongst the ones in rounds r and $r + 1$ transmitted following the first lost segment. $C_L(r, \vec{s}', \vec{\gamma}', \vec{c}')$

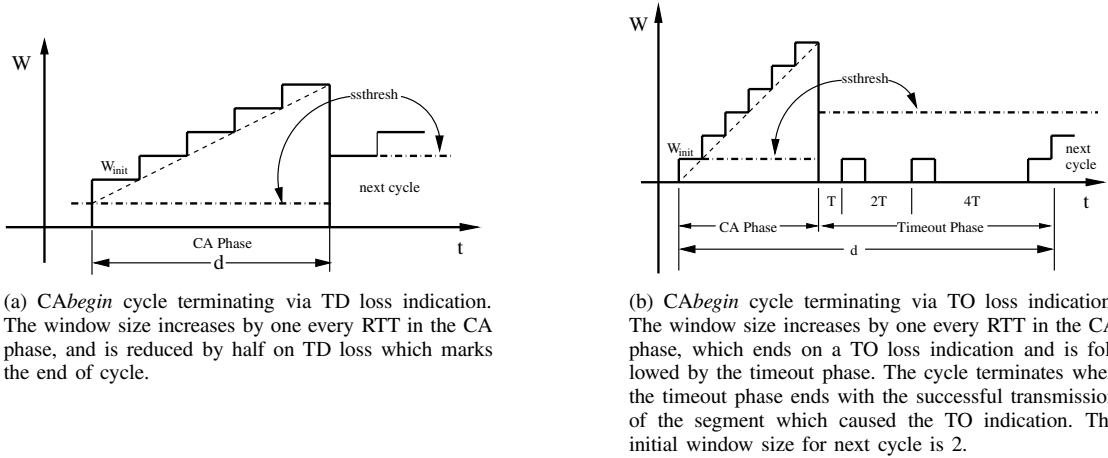


Fig. 1. Window size variation in *CABegin* TCP cycles. W_{init} is the initial window size of the cycle, $ssthresh$ represents the slow start threshold, T is the initial timeout period, and d denotes the duration of the cycle.

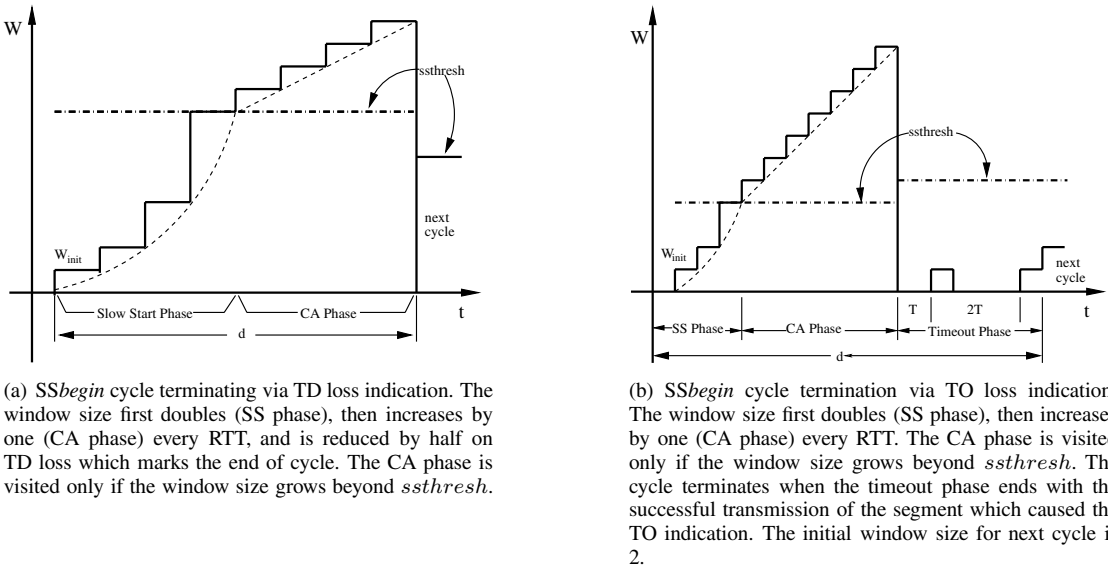


Fig. 2. Window size variation in *SSbegin* TCP cycles. W_{init} is the initial window size of the cycle, $ssthresh$ represents the slow start threshold, T is the initial timeout period, and d denotes the duration of the cycle.

represents the cost associated with transmission of segments in round $r + 1$, the terminal round for a TD loss indication. $C_B(r)$, the terminal cost for the TD loss case, is modeled in (4), as the negative of throughput attained during the cycle, where D_i represents the duration of round i , n_l is the number of segments transmitted in the terminal round $r + 1$, W_i represents the window size during round i , and λ is the scaling factor between the transmission and throughput costs. The term $-\rho(W_r)$, where ρ is an increasing function of W_r , is introduced to influence the evolution of the current cycle to favor high throughput in the subsequent cycle. For instance, when a congestion avoidance phase ends in a TD loss indication with terminal round window of W_r , the next cycle would have a initial window size of $W_r/2$. To favor a higher initial window size for the next cycle, we make the cost for the current cycle have a deduction which is an increasing function of W_r .

The event that loss indication in a cycle is a timeout is represented by probability $1 - p_{TD}(r, \vec{s}, \vec{s}')$ in (3). In

accordance with the dynamics represented in Fig. 4(b), the cost-to-go from round k of the timeout phase, given that the TO indication occurred in round r , is given by (5) where s is the success probability of the transmitted segment, and C_{TO} represents the terminal cost for the cycle. C_{TO} can be evaluated as in (6) with T_o being the average timeout value and $f(i)$ [13] given by (7). The dimension of \vec{c} in (5) is one since a single segment is transmitted in a round in the timeout phase.

The optimization framework discussed in this section can be applied to both *CABegin* and *SSbegin* cycles via the associated window evolutions. Given the initial value, the window sizes during the rounds of *CABegin* cycle can be ascertained (a linear increase every RTT until a loss indication). Similarly with a given initial window size and slow start threshold, the window sizes can be determined for rounds of *SSbegin* cycle. Then, the solution to optimization equations yields the cost minimizing segment success probability vector for all possible channel state and congestion vectors during the rounds of a

$$J(r) = \int_{\vec{\gamma}} \int_{\vec{c}} \min_{\vec{s}} [C_{tg}(r, \vec{s}, \vec{\gamma}, \vec{c})] f_{\vec{\gamma}}(\vec{\gamma}) f_{\vec{c}}(\vec{c}) d\vec{\gamma} d\vec{c} \quad (1)$$

$$C_{tg}(r, \vec{s}, \vec{\gamma}, \vec{c}) = C(r, \vec{s}, \vec{\gamma}, \vec{c}) + S_{rnd}(\vec{s})J(r+1) + (1 - S_{rnd}(\vec{s}))C_{Loss}(r, \vec{s}) \quad (2)$$

$$C_{Loss}(r, \vec{s}) = \int_{\vec{\gamma}} \int_{\vec{c}} \min_{\vec{s}'} [p_{TD}(r, \vec{s}, \vec{s}') [C_L(r, \vec{s}', \vec{\gamma}', \vec{c}') + C_B(r)] + (1 - p_{TD}(r, \vec{s}, \vec{s}')) T(r+2, r)] d\vec{\gamma}' d\vec{c}' \quad (3)$$

$$C_B(r) = \lambda \left(-\frac{\sum_{i=1}^r W_i + n_l}{\sum_{i=1}^r D_i} - \rho(W_r) \right), \quad (4)$$

$$T(k, r) = \int_{\vec{c}} \int_{\vec{\gamma}} \min_s [C_{seg}(r, s, \vec{\gamma}, \vec{c}) + sC_{TO}(k, r) + (1-s)T(k+1, r)] f_{\vec{\gamma}}(\vec{\gamma}) f_{\vec{c}}(\vec{c}) d\vec{\gamma} d\vec{c} \quad (5)$$

$$C_{TO}(k, r) = \lambda \left(-\frac{\sum_{i=1}^r W_i + n_l + (k - (r+1))}{\sum_{i=1}^{r+1} D_i + f(k - (r+1)) T_0} - \rho(W_r) \right) \quad (6)$$

$$f(i) = \begin{cases} 2^i - 1, & i \leq 7 \\ 127 + 64(i - 7), & i \geq 8 \end{cases} \quad (7)$$

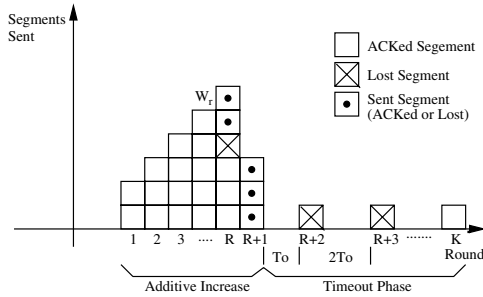


Fig. 3. TCP segment transmission dynamics in a *CABegin* cycle. The window size increases by one if all segments in a round are ACKed. The cycle ends with TD indication at round $R+1$ in case three of the segments with a dot are successful, or else the cycle ends in Timeout phase at round K . W_r represents the window size in round r . The number of segments transmitted in round $R+1$ is restricted to be within a window of size W_R (equal to 6 in the figure) following the lost segment.

cycle.

IV. TCP THROUGHPUT OPTIMIZATION VIA ADAPTIVE LINK-LAYER TECHNIQUES

Based on the framework presented in the previous section, we next demonstrate TCP's performance enhancement via adaptive link-layer measures. We show that by suitably modeling the cost of transmission, measures like power control and link adaptation can be utilized for TCP throughput optimization. To highlight the working of the optimization methodology, we begin by adopting a simple evaluation model and assess the merits of power adaptation for efficient TCP dynamics. We then move on to investigate the benefits of optimization for EGPRS and IEEE802.11a networks.

A. Adaptive Power Control

Power control [19] is desirable in wireless networks for several reasons including limiting the interference that a wireless link generates, and conserving energy for battery power limited mobile devices. TCP dynamics based power adaptation can achieve significant throughput enhancement over conventional power control mechanisms. To discuss the intuition behind the TCP throughput optimization approach, we start out by restricting the assessment in this subsection to a simplified scenario without considering a generic multi-user environment.

1) *Evaluation Model*: For present evaluation we neglect the time spent by TCP in slow start phase and incorporate only *CABegin* cycles in TCP's window evolution. While many works including [13] disregard slow start dynamics, we will incorporate them in subsequent subsections. We attribute the loss of TCP segments to wireless channel errors and the losses due to congestion are neglected here. Constant length TCP segments are assumed to be encapsulated in single data link-layer frames which are transmitted at a suitable power levels. Overheads due to headers are ignored and adaptive modulation, error correction mechanisms like FEC and ARQ are disregarded here. The wireless channel is modeled as an AWGN Rayleigh flat-fading channel and the modulation is taken to be fixed as BPSK. The channel is assumed to decorrelate over transmission of successive frames, and hence the frames undergo independent fading. The wireless link is assumed to be high-bandwidth so that it does not present a bottleneck to the realizable throughput. The duration of each round is assumed to be fixed and taken as average RTT. While a high-bandwidth wireless link will comply with this assumption adopted in popular modeling approaches [13], we

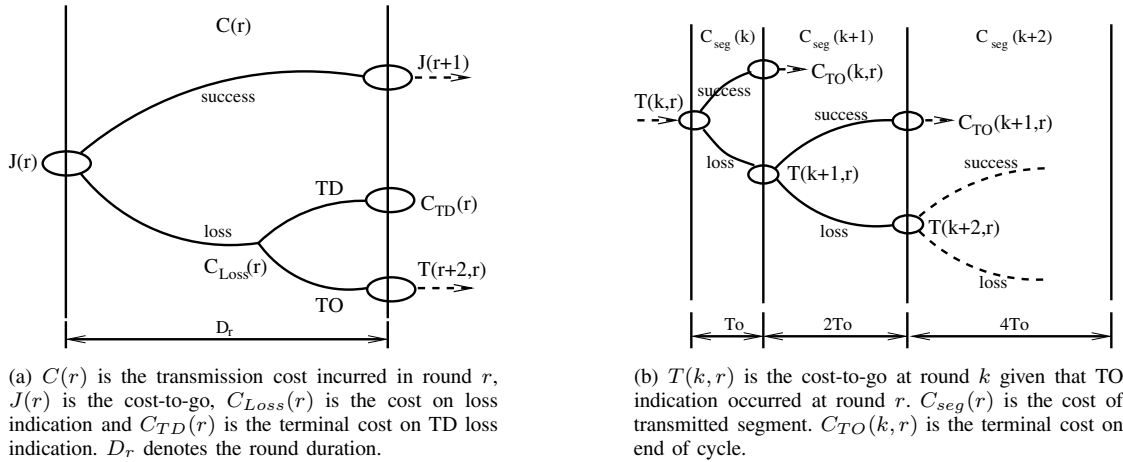


Fig. 4. Evolution of a Slow Start/Congestion Avoidance round (left) and Timeout phase (right) for SS_{begin} and $CABegin$ cycles.

$$J(r) = \int_{\vec{\gamma}} \min_{\vec{s}} \left[\sum_{j=1}^{W_r} E_{seg}(s_j, \gamma_j) + \left(\prod_{j=1}^{W_r} s_j \right) J(r+1) + \left(1 - \prod_{j=1}^{W_r} s_j \right) C_{Loss}(r, \vec{s}) \right] f_{\vec{\gamma}}(\vec{\gamma}) d\vec{\gamma} \quad (8)$$

$$J(r) = \min_s \left[W_r \int_{\gamma} E_{seg}(s, \gamma) f_{\gamma}(\gamma) d\gamma + s^{W_r} J(r+1) + (1 - s^{W_r}) C_{Loss}(r, s) \right] \quad (9)$$

$$C_{Loss}(r, s) = p_{TD}(r, s) \left[n_l \int_{\gamma'} E_{seg}(s, \gamma') f_{\gamma'}(\gamma') d\gamma' + C_B(r) \right] + [1 - p_{TD}(r, s)] T(r+2, r) \quad (10)$$

$$p_{TD}(r, s) = \frac{1}{(1 - s^{W_r})} \sum_{i=1}^{W_r} s^{i-1} (1 - s) [I_{\{W_r > 3\}} \sum_{j=3}^{W_r-1} \binom{W_r-1}{j} s^j (1 - s)^{W_r-1-j}] \quad (11)$$

$$= \left[\sum_{i=3}^{W_r-1} \binom{W_r-1}{i} s^i (1 - s)^{W_r-1-i} \right] I_{\{W_r > 3\}} \quad (12)$$

$$T(k, r) = \min_s \left[\int_{\gamma} E_{seg}(s, \gamma) f_{\gamma}(\gamma) d\gamma + s C_{TO}(k, r) + (1 - s) T(k+1, r) \right] \quad (13)$$

will relax it after this section and evaluate round duration accurately. The retransmission timeout value T_o is taken to be fixed, and the updates in timeout value are considered only in subsequent subsections. The channel state vector $\vec{\gamma}$ is represented by SNR experienced on the wireless channel by frames encapsulating TCP segments. A segment is in error if any of the bits of the encapsulating frame is in error. Neglecting the link-layer headers, the segment error probability $p(\gamma)$ can be expressed as $1 - [1 - p_b(\gamma)]^N$, where γ is the SNR during frame transmission duration and N is the frame length in bits. $p_b(\gamma)$ represents bit error probability for AWGN channel and BPSK modulation, and γ follows a Rayleigh distribution.

2) *Optimization Framework*: We model the cost of transmission as the energy expended for the transmission of TCP segments in a round. The transmission energy cost can represent, for example, the battery drain of an energy constrained

system. Adopting the assumptions discussed in the previous subsection, (1) and (2) reduce to simplified form (8) where $\gamma_1, \gamma_2, \dots, \gamma_{W_r}$ are the components of SNR vector $\vec{\gamma}$ for round r . E_{seg} represents the energy spent in transmission of a segment.

The optimization framework allows us to determine the cost-minimizing success probability vector \vec{s} and transmission energy for the segments in a round for a given $\vec{\gamma}$. However, when transmission of segments in a round commences, we may only have an estimate of SNR for the current frame transmission. We hence modify the optimization formulation to yield a causal cost model. We select the targeted packet success probability to be the same for all segments in the round. Then \vec{s} simplifies to a single-element optimization parameter s . We also assume that for the case when there is a loss indication in a round, the target success probability of the following round (e.g. round $R+1$ in Fig. 3) is the same as

that of the current round. The resulting formulation of $J(r)$ is then given by (9). Accordingly, (3) reduces to (10). As we will see, in spite of these assumptions to yield a causal framework, significant performance enhancement in TCP throughput is obtained.

We next discuss the evaluation of $p_{TD}(r, s)$, the probability of a TD loss indication conditioned on the event that a segment is lost in round r . The probability of one or more segments in a round being lost is $(1 - s^{W_r})$, where s is the target success probability for segments in a round. The probability that the first segment lost in a round is the i^{th} one is given by $s^{i-1}(1 - s)$. The formulation for $p_{TD}(r, s)$ is then given by (11) where $I_{\{W_r > 3\}}$ is the indicator function which assumes a value of 1 when the window size in round r is greater than 3 and is 0 otherwise. The indicator function in (11) is multiplied by the probability that three or more segments in the window of segments after the first loss in round r are successful. On simplification, $p_{TD}(r, s)$ reduces to (12). The timeout cost from (5) can accordingly be expressed as (13).

3) *Performance Assessment*: Based on the discussed assumptions and methodology, we simulate bulk-transfer of TCP segments. The optimization framework is solved via Dynamic Programming. (9) and (13) are first converted to finite-period. Round numbers R_t and K_t , defined to be the terminal rounds for evaluation of optimization solutions to (9) and (13), are selected as 200 each. (9) and (13) can then be solved together with (10), (12), (4), (6) and (7) for optimal segment transmission energies. However, we note that the timeout period optimization in (13) is required to be solved for the timeout rounds for every possible termination of congestion avoidance phase (at round 1 to round R_t). This bears heavily upon the computational complexity and storage in a real-time system. Note that the cost $T(r + 2, r)$ in (9) represents the cost-to-go for the timeout phase beginning at round $r + 2$, given that loss indication occurred at round r . It incorporates a running transmission energy cost and a termination cost C_{TO} . Now instead of the optimization formulation (13), we approximate the cost-to-go for the timeout phase as the termination cost after three timeout rounds, which can be expressed as $C_{TO}(r+5, r)$ with $C_{TO}(k, r)$ given by (6). This in essence means that we disregard the energy cost modeling in timeout phase, and thus reduce the computational complexity by not having to solve for optimal segment energies. As we will show, considerable throughput enhancement is realized in spite of the timeout phase approximation which we will retain through the rest of the paper. We also note that the timeout phase assumption is restricted to optimization solutions, and the simulations are performed strictly as per the TCP dynamics.

We choose the function $\rho(W_r)$ in (4) and (6) to be $\frac{W_r}{2}$ and approximate n_l as $\frac{W_r}{2}$ [13]. The duration of rounds D_i in (4) and (6) is taken as the RTT. The following procedure is then performed for the CA_{begin} cycle and for a range of values of the cost ratio λ . A look up table S_{opt} composed of target success probability $s_{opt}(r, W_{init})$ for every round number r in the congestion avoidance phase, and every initial window size W_{init} , is generated. For a given round and initial window, the value s that minimizes the integrand in (9) is the success probability stored in the look up table. W_{init} ascertains the evolution of window size in a cycle for determining the

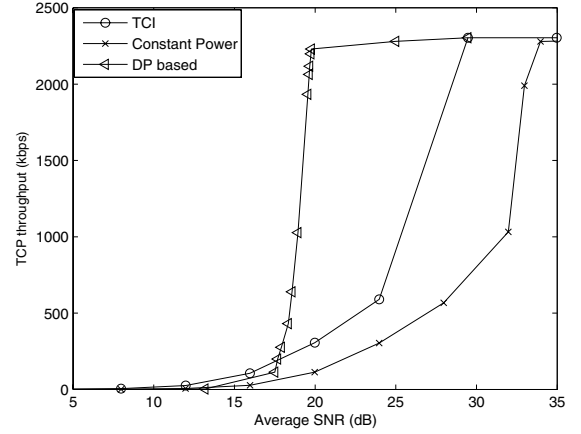


Fig. 5. TCP throughput with different power adaptation schemes

terminal costs (4) and (6), and can take values from 1 to the maximum window size, W_{max} .

For a simulation run, SNR value is drawn for every segment from a Rayleigh distribution. The target segment success probability for segment transmission in congestion avoidance phase is retrieved from look-up tables for current round and initial window size with which the cycle began. The transmission energy for frames in that round is then ascertained to achieve the target segment success probability on the wireless channel. For the timeout phase, the target success probability for each transmission attempt is taken as the maximum target segment success probability over all rounds in the congestion avoidance phase of the current cycle as noted in the look-up table. The choice is justified, since timeout phase is detrimental to TCP performance and can be exited via successful segment transmission. TCP Throughput is evaluated as the ratio of number of segments transmitted during the simulation run and the duration of the run. Several runs are performed to obtain an average throughput value. With increasing values of λ in (4) and (6), priority given to throughput cost in the dynamic programming formulation increases relative to the energy cost. Hence increasing throughput values are obtained, but at the cost of higher average segment transmission energy. The throughput variation based on DP solutions is evaluated with a segment size 1500 bytes, RTT 250 ms, T_o 3 s and the maximum window size W_{max} as 48. The transmission energy for the frames containing TCP segments is varied by adapting the transmission power levels. The performance is compared in Fig. 5 with that of a Truncated Channel Inversion (TCI) [20] power control policy with a cut-off threshold γ_c of 5 dB, and a constant transmission power policy. The TCI is a power adaptation mechanism that gives close to optimal performance for capacity realization in a single-user fading channel.

As can be seen in Fig. 5 the DP based solutions result in significant enhancement in TCP throughput. For instance, at 20 dB the optimized throughput is almost 10 times larger as compared to the one with TCI. At low SNRs, the throughput for all power control measures is insignificant since the data frames transmitted over the channel are unsuccessful owing to the absence of the support of any physical or link-layer error correction and recovery mechanisms. TCP times out waiting

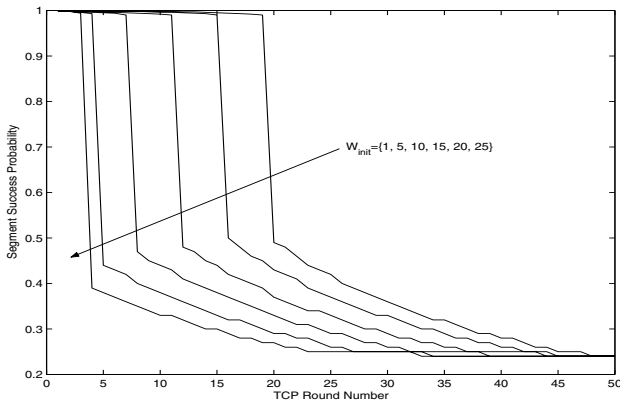


Fig. 6. Target segment success probability for DP based power adaptation. Average SNR is 18.6 dB.

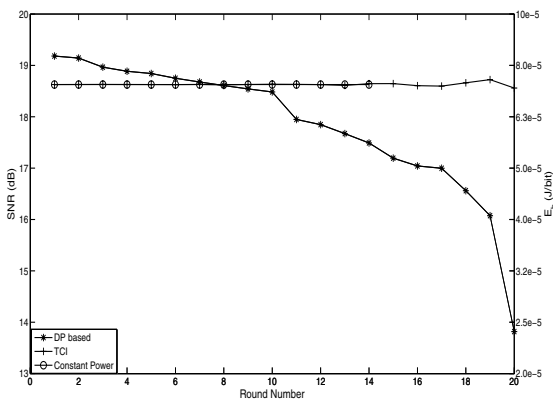


Fig. 7. Average SNR (dB) and Energy per bit (Jules/bit) for a round with different power control policies and an overall average SNR of 18.6 dB. The energy per bit is evaluated for a noise spectral density of -150 dBm/Hz, a receiver separation of 1000 m from the transmitter, and a path loss exponent of 4.

for the ACKs of the transmitted segments and exponentially increases the timeout period on each unsuccessful attempt. All power control policies thus yield negligible TCP throughput at low SNRs. There is a noticeable performance enhancement between 17-28 dB. At around 28 dB, TCI with no limitation of transmission power level quantization is able to achieve the maximum realizable TCP throughput given by $\frac{W_{max} * N_{bit}}{RTT}$ where N_{bit} is the TCP segment size in bits. Similar relative performance between power adaptation measures as noted in Fig. 5 is observed for different TCP segment sizes, W_{max} and $\hat{\Delta}RTT$ values. In conjunction with link-layer measures for error correction and recovery, the gains of our optimization methodology can be realized for a wide range of channel conditions. We will demonstrate this in subsequent sections by taking the example of EGPRS and IEEE 802.11a networks.

4) *Operational Analysis:* We now discuss some insights into the ability of DP based optimization in enhancing TCP performance. The target success probability ($s_{opt}(r, W_{init})$) for segments in a round and different initial window sizes of a cycle is plotted in Fig. 6 for an average SNR of 18.6 dB. The success probability is evaluated as solution to (9). It can be seen that TCP dynamics aware optimization sets a

higher target success probability for early rounds of a cycle. To understand the rationale behind protection of early rounds, consider the case when a *CABegin* TCP cycle starts with a small initial window size. TCP begins to probe network capacity by increasing the window size every round. If a segment loss occurs in one of the early rounds, the window size is further reduced (to half via TD or to one via TO indication) and throughput takes an adverse hit. On the other hand, a segment loss later on in a cycle when window size is large and has grown well beyond the bandwidth delay product of the network, is not that detrimental to TCP throughput. The early round protection in Fig. 6 can be seen to be more prolonged for smaller initial window sizes. For example, the target success probability for a cycle starting with initial window size of 5 is close to one for round numbers up to 16. However, it has a high value only up to round 4 when initial window size is 25. The target segment success probabilities are adaptive to channel conditions. For instance, Fig. 6 when regenerated for lower SNRs results in lower success probabilities for all rounds. Fig. 7 plots the average SNR noted during different rounds and the corresponding energy per bit for an overall average SNR of 18.6 dB. In DP solutions based power control, the initial rounds can be seen to be protected via higher SNR achieved by higher transmission power levels. However the TCI and the constant power schemes do not distinguish between segments belonging to different rounds.

B. Link Adaptation for EGPRS

GSM has been evolving towards a 3rd generation mobile cellular system with the standardization of technologies like EDGE [21] and GPRS [22]. EGPRS [23] is the part of EDGE technology targeted towards the enhancement of GPRS, which is the packet data component of GSM. While the fundamental physical layer specifications of GSM (TDMA bursts, channel spacing, etc.) are retained in EGPRS, enhancements like multi-level modulation, better code granularity, and incremental redundancy are introduced to increase the link-layer throughput. In this subsection we investigate TCP performance over EGPRS and demonstrate that TCP dynamics aware link adaptation entails, for users in the network, increased TCP throughput over conventional link adaptation methods. Since the attainable user-throughput in an EGPRS network decreases with an increase in the number of admitted users, the throughput reduction can be compensated by employing TCP dynamics aware link adaptation. Hence more of users can then be admitted in the network while retaining the same level of user-throughput.

1) *EGPRS Data Transmission Methods:* EGPRS employs nine coding schemes five of which (MCS 5 to MCS 9) are 8-PSK modulated and the remaining use GMSK modulation. Each of the coding schemes has different puncturing patterns. MCS 1, 2, 5 and 6 have two classes of puncturing patterns (P1 and P2), while the remaining coding schemes have three classes, namely P1, P2 and P3. A Packet Data Unit (PDU) from the Logical Link Control (LLC) layer is transmitted via Radio Link Control (RLC) blocks and the amount of data fitted into a block is decided by the coding scheme used for transmission. The process of matching the coding scheme to

TABLE I

PARAMETERS (BLOCK LENGTH, TRANSMISSION RATE, AND CODING RATES) FOR MCS-1 TO 9. R_{1+2} AND R_{1+2+3} DENOTE EFFECTIVE CODE RATES ON IR RECOMBINATION OF A BLOCK WITH ONE AND TWO PREVIOUS TRANSMISSIONS RESPECTIVELY.

Mode	Bl. Len.	Rate	R	R_{1+2}	R_{1+2+3}
MCS-1	176 bits	8.8 kbps	1	0.5	.33
MCS-2	224 bits	11.2 kbps	0.92	0.46	0.31
MCS-3	296 bits	14.8 kbps	0.76	0.38	0.25
MCS-4	352 bits	17.6 kbps	0.49	0.24	-
MCS-5	448 bits	22.4 kbps	0.37	0.19	-
MCS-6	592 bits	29.6 kbps	1	0.5	0.33
MCS-7	896 bits	44.8 kbps	0.85	0.42	0.28
MCS-8	1088 bits	34.4 kbps	0.66	0.33	-
MCS-9	1184 bits	59.2 kbps	0.53	0.26	-

the prevailing radio link conditions is called Link Adaptation (LA). EGPRS also employs Incremental Redundancy (IR) [23] that involves the combination of a retransmitted block with the previous transmission attempts of the block having distinct puncturing patterns. The data bits per block for each of the transmission modes and the associated data rates, modulation schemes and coding rates [24] are presented in Table I.

The throughput for a TU ideal frequency hopping and interference-limited scenario is presented for different EGRPS coding schemes, as a function of C/I_c (ratio of carrier power to co-channel interference) in [23]. The probability that a data block transmitted via MCS-1 to 9 modes is in error can be obtained as a function of C/I_c from these plots which have been parameterized in [25].

2) *Evaluation Model*: We consider a slow frequency hopping scenario where the transmission frequency is updated every 20 ms TDMA burst. The block error process is taken to decorrelate over different bursts as TCP data is sent in a time slot. We assume that the wireless link adaptation process does not have information pertaining to the network congestion, as would be the case in a real scenario. The optimization is hence not performed over congestion characteristics and \bar{c} in (1), (3) and (5) is modeled as a constant TCP segment loss process. We retain the causality assumptions and methodology in Section IV-A.2. The transmission energy cost $E_{seg}(s, \gamma)$ for a fixed transmission power level is now proportional to $T_{seg}(s, \gamma)$, the total time over the air during which bursts containing the data corresponding to a TCP segment are transmitted with γ denoting the C/I_c of the channel. We hence model the cost of transmission of a TCP segment as the time required to transmit the segment via the radio link data blocks and evaluate TCP throughput optimizing link adaptation measures. With that, (9), (10), (12) and (13) apply to the present scenario.

A TCP segment is transmitted over a variable number of EGPRS blocks, the transmission modes for which are selected via the link adaptation policy employed. The TCP Timeout period T_o is no longer assumed to be constant but is updated based on RTT observations, as described in [26]. We designate the delay experienced by a segment in the Internet and the wired part of EGPRS network excluding the wireless link transmission time as D_{Int} . Delay due to transmission on the wireless link is variable and depends on the channel conditions

and the transmission modes selected for the blocks containing TCP segment data. We model the channel state variations due to wireless channel effects and multi-user interference as a Markov Chain with its two states *good* and *bad* represented by C/I_c uniformly distributed in the intervals $[0, 15]$ dB and $(16, 30]$ dB respectively. The time spent in these states is exponentially distributed with respective rates r_g and r_b . The discrete time representation of the channel has transition probabilities $p_{g,b}$ and $p_{b,g}$ between its two states, which are related to the transition rates as $\frac{r_g}{r_b+r_g}$ and $\frac{r_b}{r_b+r_g}$ respectively. The same model has been used for the work in [27].

3) *Optimization Framework*: Methodology similar to the one highlighted in Section IV-A.3 is adopted for obtaining dynamic programming solutions. However, we now generalize the TCP dynamics to have both *SSbegin* and *CABegin* cycles. The duration of rounds is no longer assumed to be constant and can vary with the data transmission time on the wireless link. Approximations for the timeout phase as discussed in Section IV-A.3 are retained for complexity reduction.

The sample set for minimization over s in (9) would comprise all possible ways of transmission of a TCP segment via EGPRS blocks. The segment success probability for each element in this set and the corresponding average cost $\bar{T}_{seg}(s)$ (which replaces the average energy cost in (9)) can be ascertained, and optimization be performed. However, as the computation complexity becomes intractable, we restrict the segment success probability sample set (\bar{S}_{set}) by only considering identical modes to be employed for transmission of blocks carrying TCP segment data. The cardinality of \bar{S}_{set} then reduces to the number of modes of transmission. The identical mode transmission assumption is limited to the evaluation of \bar{S}_{set} and is not carried over to the selection of modes for transmission of TCP segments via EGRPS modes. \bar{S}_{set} is then ascertained as

$$\bar{S}_{set} = \{1 - \overline{SER}(1), \dots, 1 - \overline{SER}(9)\}, \quad (14)$$

where $\overline{SER}(m)$ represents the average error probability for transmission of segment via blocks of mode- m , and is given by

$$\overline{SER}(m) = \overline{BLER}_{RTX}(m)^{nBlocks(m)}, \quad (15)$$

where $nBlocks(m)$ represents the number of blocks of mode m required to transmit a TCP and $\overline{BLER}_{RTX}(m)$ is the average block error rate for mode- m with ARQ retransmissions. $\overline{BLER}_{RTX}(m)$ is ascertained as

$$\overline{BLER}_{RTX}(m) = \prod_{r=1}^{RTL} \overline{BLER}_r(m). \quad (16)$$

$\overline{BLER}_r(m)$ in the above equation represents the average (over channel state) error probability of a block when transmitted via mode- m . The subscript r denotes the transmission attempt and varies from 1 for first transmission to the retransmission limit RTL for the last one (RTL includes all transmissions of a data block, including the first one). Note that the transmission attempts yield different average block error probabilities because of IR combination. The average cost $\bar{T}_{seg}(s_m)$ corresponding to a success probability element s_m in \bar{S}_{set} is evaluated as $nBlocks(s_m)\bar{T}_{block}(m)$. $\bar{T}_{block}(m)$

$$\bar{T}_{block}(m) = \sum_{K=1}^{RTL} \prod_{r=1}^{K-1} \overline{BLER}_r(m) (1 - \overline{BLER}_K(m)) (K T_{burst}) + \prod_{r=1}^{RTL} \overline{BLER}_r(m) (RTL T_{burst}) \quad (17)$$

$$m_b = \min_{m \in \{1, \dots, 9\}} [(s_{target}(r, m) - s_{mode}(\gamma, m))^+ + \beta(t_{target}(r, m) - t_{mode}(\gamma, m))^+] \quad (18)$$

is the average time spent on transmission of a mode- m block and is given by (17), where T_{burst} represents the duration of an EGPRS radio burst of 20 ms.

Dynamic programming solutions to (9) can now be obtained for various cyclic patterns. For the *CABegin* cycle, target success probability for the congestion avoidance phase is a function of round number and initial window size, i.e., $s_{opt}^{CA}(r, W_{init})$. This constitutes the look-up table S_{opt}^{CA} . The target success probability for the slow start and congestion avoidance phases in *SSbegin* cycle is in addition a function of the slow start threshold $ssthresh$ at the beginning of the cycle, and can be expressed as $s_{opt}^{SS}(r, W_{init}, ssthresh)$: these values constitute the look-up table S_{opt}^{SS} . For the transmission of TCP segments during slow start and congestion avoidance phases, the optimal segment success probability ($s_{opt}(r)$) can be drawn for round r , from the appropriate look-up tables for *SSbegin* or *CABegin* cycles depending on the nature of current cycle and the associated W_{init} and $ssthresh$ (for *SSbegin* cycle) values. If the round r falls in timeout phase, $s_{opt}(r)$ is taken to be the maximum, across all rounds, of the success probabilities for the cycle noted in the look-up table. Following the selection of segment success probability, the choice of modes for transmission of EGPRS blocks carrying TCP segment needs to be made. If the estimated C/I_c for the current burst is γ , then mode m_b which minimizes the cost of deviation from the target success probability and transmission time is selected for block transmission. The mode m_b is given by (18) where β is the cost ratio, and the function $(x)^+$ represents $\max\{0, x\}$. $s_{target}(r, m)$ and $t_{target}(r, m)$ respectively represent, for mode- m and round r , the target success block probability and target block transmission time as required by the optimization methodology for transmission. $s_{mode}(\gamma, m)$ and $t_{mode}(\gamma, m)$ on the other hand represent the estimate of success probability and transmission time that can be offered by mode- m for a C/I_c of γ . The target success probability for mode selection, $s_{target}(r, m)$, is taken to be $[s_{opt}(r)]^{\frac{1}{nBlocks(m)}}$. Similar to the \bar{S}_{set} evaluation assumption of segment transmission via identical modes, the estimation of target block error probability from segment error probability is done here by raising s_{opt} (an element of \bar{S}_{set}) to the inverse power of $nBlocks(m)$. The block success probability $s_{mode}(\gamma, m)$ in (18) is ascertained as $1 - BLEER_{RTX}(\gamma, m)$, where $BLEER_{RTX}(\gamma, m)$ is the estimate of block error rate offered by mode- m with ARQ retransmissions. Having C/I_c information only for the current block transmission $BLEER_{RTX}(\gamma, m)$ is estimated as

$$BLEER_{RTX}(\gamma, m) = BLEER_1(\gamma, m) \prod_{r=2}^{RTL} \overline{BLER}_r(m), \quad (19)$$

by employing average block error rate for retransmission attempts. The target time $t_{target}(r, m)$ in (18) can be evaluated as

$$t_{target}(r, m) = \frac{\overline{RTT}}{W_r nBlocks(m)}, \quad (20)$$

where \overline{RTT} is the running average round trip time (as maintained by a TCP implementation [26]) of a TCP segment, and W_r is the TCP window size. The transmission time estimate for a mode- m block transmission is taken to be $t_{mode}(m) = T_{block}(\gamma, m)$, where $T_{block}(\gamma, m)$ is given by (21). The three terms separated by addition signs in (21) respectively represent the cases of block transmission success at first attempt, failure across all attempts, and success at K^{th} attempt where $2 \leq K \leq RTL$. As long as there is outstanding TCP segment data to be transmitted, transmission modes are selected according to (18). If the transmission is successful, the transmitted data is removed from the payload buffer. Mode selection as per the present channel conditions is then repeated to transmit the remaining TCP segment data. The process is repeated until the segment is completely transmitted.

4) *The Early Round Protection (ERP) Scheme*: ERP targets TCP throughput enhancement in wireless networks, and is based on the observation that earlier rounds in a TCP cycle are crucial for attaining higher throughput and need a transmission bias (e.g. higher transmission power, robust coding). As per ERP scheme, adaptive mode selection is done for EGPRS following the measures highlighted in the previous subsections. After the growth of window size beyond a limit W_{erp} , ordinary LA/IR adaptation [23] procedures for data transmission take over. The window size limit W_{erp} is adaptive depending on the size of TCP segments and RTT.

To demonstrate the merits of the ERP policy, simulations are performed for two separate sets of cases: a simplified one with no IR and the other with approximate IR. The cost ratio β in (18) is found to give optimal or close to optimal throughput for a value of 0.1 when time is in units of seconds. The parameter λ in (6) and (4), is varied to find optimal TCP throughput. The EGPRS Radio Link Control and block transmission mechanisms are executed in the simulations. The TCP Timeout period To is initialized to $3s$ and is updated based on RTT observations, as described in [26]. For DP based optimization, To in (6) is set to the average timeout value. W_{erp} is approximated as the ratio of running average RTT (as maintained by a TCP implementation [26]) of TCP segments, and the segment transmission time. The segment transmission time is estimated as the average over γ and mode- m of the product $nBlocks(m) T_{block}(\gamma, m)$, where $T_{block}(\gamma, m)$ is given by (21). For the evaluation case with no IR, the BLER does not depend on the transmission attempt number since there is no block combining. The BLER subscripts in

$$\begin{aligned}
T_{block}(\gamma, m) &= [1 - BLER_1(\gamma, m)] T_{burst} \\
&+ BLER_1(\gamma, m) \sum_{K=2}^{RTL} \prod_{r=1}^{K-1} \overline{BLER}_r(m) (1 - \overline{BLER}_K(m)) (r T_{burst}) \\
&+ BLER_1(\gamma, m) \prod_{r=2}^{RTL} \overline{BLER}_r(m) (RTL T_{burst})
\end{aligned} \tag{21}$$

TABLE II

S_{opt}^{CA} AND S_{opt}^{SS} TABLES FOR TCP OVER EGPRS UNDER DIFFERENT CHANNEL CONDITIONS, WITH IR, $D_{Int}=1000$ MS, $p_{Int} = 0\%$, $N_{seg}=512$ BYTES, AND $W_{max}=64$. THE TWO TABLES ARE RESPECTIVELY PRESENTED FOR A CA_{begin} CYCLE WITH A W_{init} OF 2, AND FOR AN SS_{begin} CYCLE WITH A W_{init} OF 2 AND AN $ssthresh$ OF 16. \vec{S}_{set} FOR $p_{bg} = 0.2$ IS $\{0.3942, 0.3886, 0.2979, 0.2492, 0.1368, 0.1747, 0.1578, 0.1276, 0.1018\}$, FOR $p_{bg} = 0.4$ $\{0.5945, 0.5895, 0.5082, 0.4602, 0.3413, 0.3917, 0.3684, 0.3235, 0.2697\}$, AND FOR $p_{bg} = 0.8$ $\{0.9436, 0.9415, 0.9221, 0.9086, 0.8886, 0.8994, 0.8730, 0.8293, 0.7685\}$

$s_{opt}^{CA}(r, 2)$						$s_{opt}^{SS}(r, 2, 16)$					
$p_{bg} = 0.2$		$p_{bg} = 0.4$		$p_{bg} = 0.8$		$p_{bg} = 0.2$		$p_{bg} = 0.4$		$p_{bg} = 0.8$	
r	s	r	s	r	s	r	s	r	s	r	s
1-17	0.3942	1-9	0.5945	1-16	0.9436	1-6	0.3942	1-3	0.5945	1-8	0.9436
18-31	0.3886	10-16	0.5895	17-33	0.9415	7-21	0.3886	4-5	0.5895	9-23	0.9415
32-43	0.2979	17-20	0.5082	34	0.8994	22-32	0.2979	6-9	0.5082	24-200	0.8293
44-52	0.2492	21	0.4602	35-200	0.8293	33-41	0.2492	10	0.4602		
53-200	0.1747	22-24	0.3917			42-200	0.1747	11-14	0.3917		
		25-31	0.3684					15-20	0.3684		
		32-200	0.3235					21-200	0.3235		

(19) and (21) designating the transmission number can hence be dropped. $BLER(\gamma, m)$ is evaluated using the EGPRS throughput performance plots in [23]. We also demonstrate ERP merits for another scenario with approximate IR. The performance information on IR for some MCS schemes is tabulated in works including [23]. However, complete set of data is not available in literature. To obviate the limitation, we adopt measures to approximate IR mechanisms. Upon every block retransmission, the code rate of an MCS scheme is reduced due to IR combining with previous transmissions. This code rate reduction for second and third transmission is noted in Table I. To obtain the block error rate for retransmissions, we determine the MCS of the same modulation as the retransmitted block, which has its code rate $R1$ closest to the code rate of a retransmitted data block upon IR combination (R_{1+2} or R_{1+2+3}). We then approximate the BLER of the retransmitted block to be the same as that of this MCS. For example R_{1+2} for MCS-9 (0.5) is the closest to $R1$ of MCS-6 (0.49). Since these schemes have the same modulation (8PSK), we approximate BLER for first IR combination of a retransmitted MCS-9 block as that of an MCS-6 transmission. BLER for MCS-6 transmission can be ascertained using the EGPRS throughput plots [23]. The above mentioned approach enables us to approximate IR performance for high code rate schemes. For low code rate schemes it has been observed that IR is anyway not very beneficial [23].

Table II shows the S_{opt}^{CA} and S_{opt}^{SS} look up tables for different transition probability p_{bg} values and for a sample scenario with approximate IR, Internet congestion loss p_{Int} as 0%, delay D as 1000 ms, TCP segment size N_{seg} as 512 bytes, and W_{max} as 64. The target segment success probabilities are low for adverse channel conditions and high for good channel states. The early rounds can be observed to have higher segment

success probabilities than the later ones. Figs. 8 and 9 show TCP throughput versus $p_{b,g}$ with N_{seg} as 512 bytes, W_{max} as 64, and p_{Int} as 0% and 5% respectively. For the approximate IR case, results are presented for P1+P2 and P1+P2+P3 recombinations, which represent the IR combination of blocks transmitted via puncturing schemes P1, P2 and P3.

TCP throughput performance for no congestion loss and $D_{Int} = 1000$ ms can be seen for $RTL=2$ with IR (represented by IR(P1+P2)) and for $RTL=2$ with no recombination, in Fig. 8(a). The corresponding results for $RTL=3$ with IR (represented by IR(P1+P2+P3)), and for $RTL=3$ with no recombination are shown in Fig. 8(b). We note that for the modes that offer only two puncturing schemes, P1 puncturing is used for the third transmission attempt. Fig. 9 plots TCP throughput for approximate P1+P2 and P1+P2+P3 (denoting combination of blocks transmitted via puncturing schemes 1, 2 and 3) combining, with $D_{Int} = 500$ ms and 1000 ms, and Internet congestion loss of 5%.

From the presented results, DP based link adaptation can be seen to yield substantial performance enhancement over LA which selects transmission mode based on the maximum throughput for a given C/I_c . For low values of $p_{b,g}$ (representing the case when the channel tend to stay in the bad state) TCP dynamics based optimization measures can be seen to result in several folds throughput increase. Noticeable improvement in throughput can be observed for good channel conditions as well. The returns of performance enhancement can be seen in Figs. 8 and 9 to be diminishing as RTL increases. However, enhancing TCP throughput while keeping the RTL low is desirable for several reasons. Latency and bandwidth overheads due to retransmissions increase with RTL . Especially real-time applications running on a device with high RTL setting may experience degraded performance.

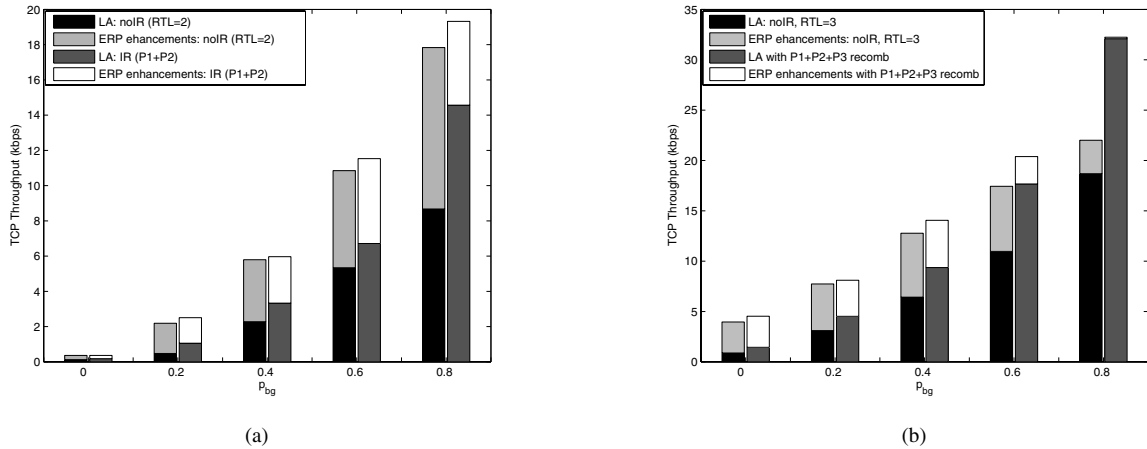


Fig. 8. TCP throughput over EGPRS with and without IR for $D_{Int}=1000$ ms, $p_{int} = 0\%$, $N_{seg}=512$ bytes, and $W_{max}=64$. IR(P1+P2) signifies that a block, if in error, can be retransmitted once via puncturing scheme P2 and be combined with the previous transmission (of puncturing scheme P1) at the receiver. IR(P1+P2+P3) can be explained similarly.

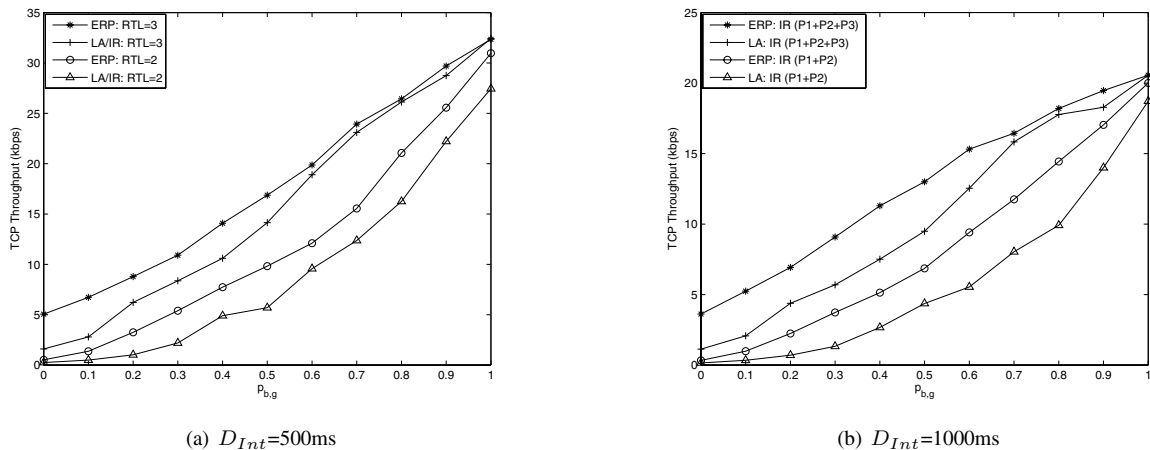


Fig. 9. TCP throughput over EGPRS with IR for $p_{int} = 5\%$, $N_{seg}=512$ bytes, and $W_{max}=64$.

Again, with a high RTL, not only are the data units sent multiple times, but additional control traffic must flow to request retransmissions.

C. Link Adaptation for IEEE 802.11a

The IEEE 802.11 [28] is a widely prevalent WLAN standard with several compliant products in use. 802.11a [29] is a high speed PHY extension to the original standard and provisions data rates from 6 Mbps to 54 Mbps. The multiple data rates are supported via transmission modes which employ a combination of distinct modulation and coding schemes. The process of adapting these transmission modes to radio conditions is called link adaptation. The target of a conventional link adaptation policy could be high MAC goodput, low frame error rate, etc. For instance, a dynamic mode selection policy for IEEE 802.11a PHY is suggested in [27] for realization of high link-layer goodput under DCF operation. Mode selection is performed for every MPDU transmission and retransmissions. In this section we will adopt the goodput evaluation approach suggested in [27] and demonstrate the benefits of TCP throughput optimization based link adaptation.

TABLE III
PHY MODES FOR IEEE 802.11a. N_{DBPS} REPRESENTS DATA BYTES PER OFDM SYMBOL

Mode	Data Rate	Modulation	Code Rate	N_{DBPS}
1	BPSK	6Mbps	1/2	3
2	BPSK	9Mbps	3/4	4.5
3	QPSK	12Mbps	1/2	6
4	QPSK	18Mbps	3/4	9
5	16-QAM	24Mbps	1/2	12
6	16-QAM	36Mbps	3/4	18
7	64-QAM	48Mbps	2/3	24
8	64-QAM	54Mbps	3/4	27

1) *IEEE 802.11 PHY and Data Transmission:* IEEE 802.11a employs an OFDM PHY with 52 subcarriers, 48 of which are used to carry data. The data rates, modulation and code rate pertaining to the 8 transmission modes are shown in Table III.

The transmission time of an l byte data payload using PHY

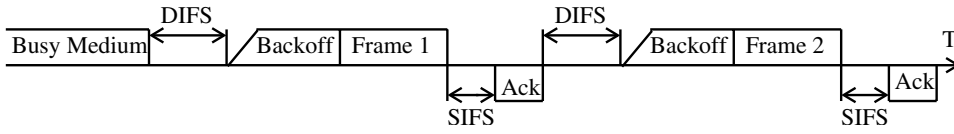


Fig. 10. Timing of successful frame transmission under DCF

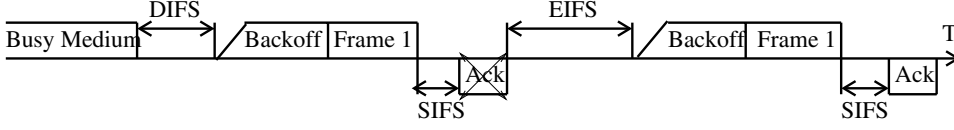


Fig. 11. Frame retransmission due to ACK failure

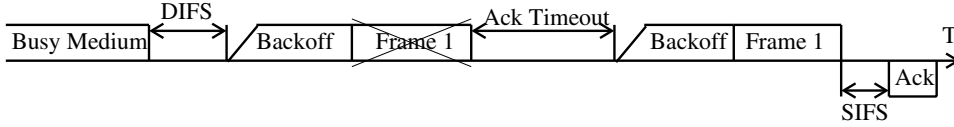


Fig. 12. Frame retransmission due to an erroneous frame reception

$$T_{wait}(\gamma, m) = FER(\gamma, m) [tSIFSTime + Tack(m') + tSlotTime] + [1 - FER(\gamma, m)] AER(\gamma, m') \cdot [tSIFSTime + Tack(m') + tSIFSTime + Tack(1) + tDIFSTime]; \quad (25)$$

mode m is shown in [27] to be

$$T_{data}(l, m) = 20\mu s + \left\lceil \frac{30.75 + l}{N_{DBPS}(m)} \right\rceil \cdot 4\mu s, \quad (22)$$

and the transmission duration of an ACK frame using mode m' :

$$T_{ack}(m') = 20\mu s + \left\lceil \frac{16.75}{N_{DBPS}(m')} \right\rceil \cdot 4\mu s \quad (23)$$

The data frames can be transmitted at the any of the supported rates. However the ACK frames can be transmitted only at one of the rates in the BSS rate set: {6Mbps, 12Mbps, 24Mbps}. Also an ACK frame is transmitted at the highest rate in BSS basic rate set that is less than or equal to the rate of data frame it is acknowledging.

Before the transmission of a frame, a sender waits for a random backoff interval. The backoff interval which is in units of Slot time, has a uniform distribution in $[0, CW]$, where CW , the contention window has minimum and maximum permissible values of $aCWmin$ and $aCWmax$. In case a transmission is unsuccessful, CW is updated to $[2(CW+1)-1]$. When the transmission is successful CW value is reset to $aCWmin$. The average backoff interval before the i^{th} transmission attempt of a frame can be evaluated, as shown in [27], as

$$\bar{T}_{backoff}(i) = \frac{\min[2^{i-1}(aCWmin + 1) - 1, aCWmax]}{2} tSlotTime \quad (24)$$

with values $tSlotTime$, $aCWmin$ and $aCWmax$ for 802.11a PHY taken as $9\mu s$, 15 and 1023 respectively.

The frame transmission timings under DCF operation, as shown in [27], are depicted in Figs. 10, 11 and 12. A successful frame transmission duration (Fig. 10) is given

by the sum of backoff delay, data transmission time, SIFS time ($tSIFSTime$), ACK transmission time and a DIFS time ($tDIFSTime$). However, when a frame transmission fails, the transmitting station has to wait for an EIFS interval (if the frame is lost, Fig. 11) or an Ack timeout interval (if ACK for the frame is lost, Fig. 12) before executing the backoff preceding retransmission. An EIFS interval is equal to the sum of $tSIFSTime$, $tDIFSTime$ and Ack transmission time at most robust 6Mbps. An Ack timeout is equal to a SIFS time plus an ACK transmission time plus a slot time. The SIFS and DIFS times are for 802.11a PHY are $16\mu s$ and $34\mu s$ respectively. With that, the average wait time after a failed transmission of mode m at SNR γ , can be expressed as (25). The frame error rate $FER(\gamma, m)$ denotes the probability that an MPDU transmitted via mode m and at SNR γ is in error. The corresponding ACK error probability is represented as $AER(\gamma, m')$ with the approximation that SNR γ for the frame and ACK transmission is the same. The transmission mode for ACK frame corresponding to frame transmission mode m is expressed as m' .

2) *TCP Dynamics Aware Link Adaptation*: With the framework outlined above, we will evaluate TCP dynamics aware link adaptation measures for 802.11a through a similar approach as adopted for EGPRS. We make the similar assumptions as in Section IV-B.3 to determine the target success probability set \vec{S}_{set} . The set is evaluated as in (14) with 8 modes. $\overline{SER}(m)$ represents, as before, the segment error rate assuming transmission via mode m . We take that a TCP segment is encapsulated in a single MPDU and the average segment error rate $\overline{SER}(m)$ then equals the average MPDU Error Rate with MAC ARQ retransmissions. With the average MPDU error rate abbreviated as $\overline{FER}_{RTX}(m)$ (Frame Error

$$T_{frame}(\gamma, m) = [1 - FER(\gamma, m)]T_{success}(\gamma, m) + FER(\gamma, m) \sum_{n=2}^{RTL} [\overline{FER}(m)]^{n-1} \overline{FER}(m) T_{RTX}(\gamma, m, n) + FER(\gamma, m) [\overline{FER}]^{RTL-1} T_{fail}(\gamma, m) \quad (31)$$

$$T_{success}(\gamma, m) = tDIFStime + \bar{T}_{backoff}(1) + T_{data}(m) + tSIFStime + T_{ack}(m') \quad (32)$$

$$T_{RTX}(\gamma, m, n) = tDIFStime + [T_{wait}(\gamma, m) + (n-2)\bar{T}_{wait}(m)] + \sum_{i=1}^n \bar{T}_{backoff}(i) + nT_{data}(m) + tSIFStime + \bar{T}_{ack}(m) \quad (33)$$

$$T_{fail}(\gamma, m) = tDIFStime + [T_{wait}(\gamma, m) + (RTL-2)\bar{T}_{wait}(m)] + \sum_{i=1}^n \bar{T}_{backoff}(i) + nT_{data}(l, m) \quad (34)$$

Rate), we then have

$$\overline{SER}(m) = \overline{FER}_{RTX}(m), \quad (26)$$

where $\overline{FER}_{RTX}(m)$ is estimated as

$$\overline{FER}_{RTX}(m) = (\overline{FER}(m))^{RTL} \quad (27)$$

In the above relation, it is assumed that MPDU retransmissions are restricted to be done via mode m . However, as before, this assumption is restricted to the evaluation of \bar{S}_{set} and, as we will see, to mode selection - the actual link adaptation will ascertain a suitable mode for every MPDU transmission and retransmissions.

Using Dynamic Programming, look up tables \bar{S}_{opt}^{CA} and \bar{S}_{opt}^{SS} are evaluated as in Section IV-B.3 and mode selection policy (18) is employed. The target success probability $s_{target}(m)$ for every mode m is taken as $s_{opt}(r)$, the optimal success probability for TCP segments given by DP solutions. For link-adaptation, $s_{opt}(r)$ is drawn for the current round r from \bar{S}_{opt}^{CA} or \bar{S}_{opt}^{SS} , depending on the cycle in progress.

The success probability for the transmission modes is estimated as

$$s_{mode}(m, \gamma) = 1 - FER_{RTX}(\gamma, m) \quad (28)$$

where, similar to EGPRS case, $FER_{RTX}(\gamma, m)$ for the success probability set evaluation is taken to be

$$FER_{RTX}(\gamma, m) = 1 - FER(\gamma, m) (\overline{FER}(m))^{RTL-1} \quad (29)$$

$FER(\gamma, m)$ above represents the error probabilities of an MPDU when transmitted via mode m : its evaluation will be discussed at the end of this subsection. The target time $t_{target}(m)$ in mode selection (18) is taken to be

$$t_{target}(m, r) = \frac{RTT}{W_r} \quad (30)$$

The transmission time of a mode- m frame is selected as $t_{mode}(m) = T_{frame}(\gamma, m)$ where the frame transmission time, $T_{frame}(\gamma, m)$, is estimated as in (31). Note that the

assumption of transmission via same mode is again made again for mode selection. The expression (31) is presented as the sum of three terms representing respectively the cases of frame transmission success on the first attempt, success on a retransmission attempt, and failure through all attempts. The transmission time for success on the first attempt (Fig. 10) is given by (32). It incorporates the $tDIFStime$ wait, the backoff and payload- l frame transmission time by the transmitting station. It further includes the $tSIFStime$ wait time and ACK transmission time by the receiving station. The small air propagation delay has been neglected in these computations [27]. The frame transmission time for n^{th} transmission attempt is given by (33). It includes the time T_{wait} that a transmitting station has to wait before each retransmission attempt. In the event that none of the transmission attempts is successful, the time spent for frame transmission is given by $T_{fail}(\gamma, m)$ in (34).

3) *Performance Assessment*: For demonstrating the benefits of TCP dynamics based power adaptation, we select a TCP payload size in a segment (N_{seg}) to be 2000 bytes and use the MAC goodput plots in [27] for MSDU of 2000 bytes. We assume that an MSDU payload is transmitted via a single MPDU. From the goodput plot for a given mode m , the ratio of goodput at a given SNR and the maximum goodput, represents the success probability of an MPDU at that SNR. The frame error rate $FER(\gamma, m)$ is thus ascertained. The ACK error rate $AER(\gamma, m')$ in (25) is taken to be the same as $FER(\gamma, m')$. The maximum goodput for a mode m represents the rate $R(m)$ that TCP is offered by that mode. These rates for 2000 bytes MPDU size are approximately {5.7, 7, 11, 15.5, 20, 27.5, 34, 37} Mbps as ascertained from the goodput plots.

Optimization methodology similar to Section IV-B.3 is adopted for obtaining link adaptation measures. Whenever a new TCP segment is to be transmitted, the transmission mode is selected based on the mode selection policy presented in the previous subsection. If the MPDU encapsulating TCP the segment is not successful, the subsequent retransmission

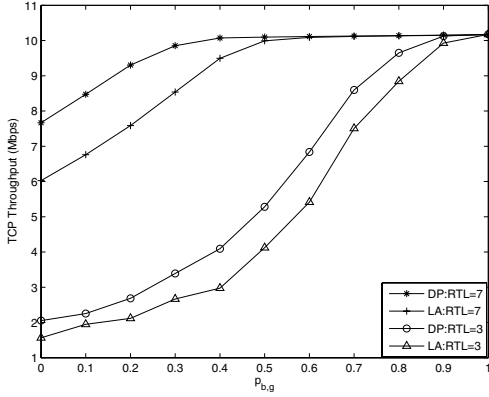


Fig. 13. TCP throughput over IEEE 802.11a for $D=100\text{ms}$, $p_{int} = 0\%$, $N_{seg}=2000$ bytes, and $W_{max}=64$.

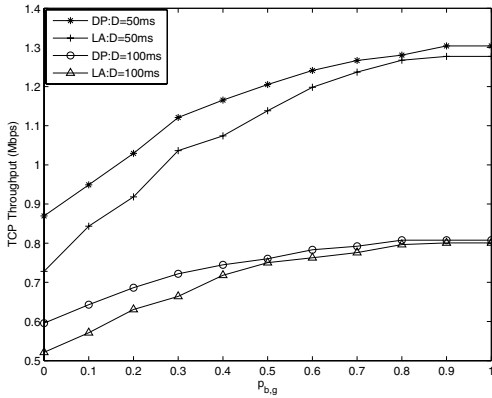


Fig. 14. TCP throughput over IEEE 802.11a for $RTL=3$, $p_{int} = 5\%$, $N_{seg}=2000$ bytes, and $W_{max}=64$.

modes are selected by ascertaining the mode which offers the maximum goodput (or the minimum frame error rate) at the current SNR.

The throughput achieved via the policy above is compared with that of pure goodput based link adaptation whereby the modes for every MPDU transmission are selected based on maximum goodput they offer.

Fig. 13 plots TCP throughput versus the transition probability $p_{b,g}$, for $D=50$ ms and 100 ms, and no Internet congestion loss. Fig. 14 shows the performance plot for Internet loss of $p_{int} = 5\%$ and RTL of 3.

It can be seen for all scenarios that link adaptation based on TCP dynamics yields a higher throughput than MPDU link adaptation based on maximum MAC goodput. In Fig. 13 the throughput can be seen to be improved by upto about 65% for $RTL=3$, and 30% for $RTL=7$. Again, Fig. 13 shows that the throughput is increased by upto about 30% for $D=50$ ms, and 20% for $D=100$ ms.

For the present IEEE 802.11a scenario, increasing RTL seems to have a greater impact on TCP throughput than incorporating link adaptation based TCP throughput optimization. However, as discussed at the end of Section IV-B.4, improving TCP throughput while keeping the RTL low is desirable for several reasons.

V. CONCLUSION AND DIRECTIONS FOR FUTURE RESEARCH

The window size trace of any bulk-transfer TCP flow is composed of recurring patterns which when suitably identified can be used for overlaying a cross-layer optimization strategy that enables the utilization of adaptive link-layer measures to enhance TCP throughput. The optimization approach suggested in this paper protects the rounds of TCP during the nascent evolution of the congestion window before it grows to the bandwidth-delay product of the network. The protection is rendered by employing, for example, robust modulation and coding, higher transmission power, etc. We demonstrate the principle via power control in a simplified scenario and link adaptation in EGPRS and IEEE 802.11a networks.

The work presented in this paper leads to several potential areas for further investigation, the networking stack architecture for implementing TCP dynamics aware link adaptation being one of them. Consider the scenario where a wireless device is the sender end of a TCP connection. The device can have data arriving at the link-layer from multiple transport layer protocols and also several TCP flows at the transport layer. In order to facilitate cross-layer TCP throughput optimization, the link-layer should permit selective link-adaptation for the data belonging to different TCP flows. This requires the implementation of an interface between the transport and link layers to identify the flows. Another aspect of cross-layer architecture is making TCP flow information available to the link-layer. The TCP dynamics aware link adaptation requires a wireless host to adapt its transmission mode to the round, cycle, and phase to which TCP segments belong. The round number, cycle, and phase information can be tunneled from the transport-layer to the link-layer of the host. On receiving TCP segment data, the link-layer can then strip the information and ascertain the transmission modes for the TCP payload as per the prevailing radio conditions.

When the mobile host is at the receiving end of the TCP connection, the round number, cycle, and phase information can be tunneled with the TCP segments by the sender and parsed by the BS (or AP). However, the sender may be an arbitrary host in the Internet with no support for TCP dynamics tunneling. The TCP dynamics information may then be inferred by the BS (or AP) by observing the sequence numbers of the TCP segments and the times of their arrival. The modes for transmission of the TCP payload can then be decided by the wireless transmitter.

In Section IV-A, we evaluated the TCP throughput optimization framework via power control for a single wireless transmitter. While the suggested power adaptation measures would directly apply to a scenario where the transmissions of multiple transmitters are strictly separated in time and/or frequency, it would make for an important contribution to analyze power control policies for the case where interference is responsive to power at which the transmitters transmit. A starting point could be to have the transmitters adopt heuristics for target segment success probability variation with round number in TCP cycles (for instance as highlighted in Fig. 6). From that, the target Signal to Interference Ratios (SIRs) for the TCP segments will then need to be estimated by the

transmitters. All transmitters can then adapt their powers in accordance with a distributed power control algorithm (the design of which is also an open issue) that attains these target SIRs for the TCP segments. Subsequently, the distribution of interference experienced by the transmitters in such a scenario can be evaluated, for instance, via simulation methods. With that, the heuristics for target segment success probability can be updated by solving the optimization framework of Section III. The iterative procedure for update of heuristics can be repeated until the gain in aggregate TCP throughput of the network saturates. The final heuristics can then be adopted in conjunction with the distributed power control algorithm in a real network.

Finally, it would be interesting to perform the modeling and evaluation of the optimization measures suggested in this work with TCP's delayed ACK mechanism.

ACKNOWLEDGMENT

We thank Andrea Goldsmith for her useful comments on the work. Thanks to Bernd Girod and Gerhard Kadel for the discussions on link-layer adaptation measures. The authors also thank the reviewers who helped in improving the clarity of presentation and results.

REFERENCES

- [1] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Trans. Networking*, vol. 5, no. 6, pp. 756–769, Dec. 1997.
- [2] P. K. Gupta and J. Kuri, "Reliable ELN to enhance throughput of TCP over wireless links via TCP header checksum," in *Proc. IEEE GLOBECOM*, Nov. 2002, vol. 2, pp. 1985–1989.
- [3] R. Balan, B. Lee, K. Kumar, L. Jacob, W. Seah, and A. Ananda, "TCP HACK: TCP header checksum option to improve performance over lossy links," in *Proc. IEEE INFOCOM*, Apr. 2001, vol. 1, pp. 309–318.
- [4] J. Cobb and P. Agarwal, "Congestion or corruption? A strategy for efficient wireless TCP sessions," in *Proc. IEEE Symposium Computers Commun.*, June 1995, pp. 262–268.
- [5] K. Brown and S. Singh, "M-TCP: TCP for mobile cellular networks," in *Proc. ACM SIGCOMM*, Oct. 1997, vol. 27, no. 5, pp. 19–43.
- [6] A. Bakre and R. Badrinath, "Handoff and system support for Indirect TCP/IP," in *Proc. 2nd USENIX Symposium Mobile Location-Dependent Computing*, Apr. 1995, pp. 11–24.
- [7] M. C. Chan and R. Ramjee, "TCP/IP performance over 3G wireless links with rate and delay variation," in *Proc. MobiCom*, Sep. 2002, pp. 71–82.
- [8] G. M. Laura Galluccio and S. Palazzo, "An analytical study of a tradeoff between transmission power and FEC for TCP optimization in wireless networks," in *Proc. IEEE INFOCOM*, Mar. 2003, vol. 3, pp. 1765–1773.
- [9] D. Barman, I. Matta, E. Altman, and R. E. Azouzi, "TCP optimization through FEC, ARQ, and transmission power tradeoffs," in *Proc. 2nd Intl. Conf. Wired/Wireless Internet Commun.*, Feb. 2004, pp. 87–98.
- [10] 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, vol. 5, pp. 3553–3559.
- [11] W. Stevens, "TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms," RFC 2001, Jan. 1997.
- [12] E. Altman, K. Avrachenkov, and C. Barakat, "A stochastic model of TCP/IP with stationary random losses," *IEEE/ACM Trans. Networking*, vol. 13, pp. 356–369, Apr. 2005.
- [13] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," in *Proc. ACM SIGCOMM*, Sep. 1998, pp. 303–314.
- [14] M. Allman, V. Paxson, and W. Stevens, "TCP congestion control," RFC 2581, Apr. 1999.
- [15] M. Mathis, J. Madhavi, S. Floyd, and A. Romanow, "TCP selective acknowledgement options," RFC 2018, Oct. 1996.
- [16] S. Floyd and T. Henderson, "The NewReno modification to TCP's fast recovery algorithm," RFC 2582, Apr. 1999.
- [17] R. Braden, "Requirements for Internet hosts - Communication layers," RFC 1122, Oct. 1989.
- [18] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 2nd ed. Belmont, MA: Athena Scientific, 2001.
- [19] J. M. Rulnick and N. Bambos, "Mobile power management for wireless communication networks," *ACM/Baltzer J. Wireless Networks*, vol. 3, no. 1, pp. 3–14, Mar. 1997.
- [20] A. J. Goldsmith and P. Varaiya, "Capacity of fading channels with channel side information," *IEEE Trans. Inform. Theory*, vol. 63, no. 6, pp. 1986–1992, Nov. 1997.
- [21] S. Hamiti, M. Hakaste, M. Moisio, N. Nefedov, E. Nikula, and H. Vilpponen, "EDGE circuit switched data - An enhancement of GSM data services," in *Proc. IEEE WCNC*, Sep. 1999, vol. 3, pp. 1437–1441.
- [22] G. Brasche and B. Walke, "Concepts, services, and protocols of the new GSM phase 2+ general packet radio service," *IEEE Commun. Mag.*, pp. 94–104, Aug. 1997.
- [23] D. Molkdar, W. Featherstone, and S. Lamotharan, "An overview of EGPRS: the packet data component of EDGE," *Electronic Commun. Eng. J.*, vol. 14, no. 1, pp. 21–38, Feb. 2002.
- [24] A. Furuskar, D. Bladsjo, S. Eriksson, M. Frodigh, S. Javerbring, and H. Olofsson, "System performance of the EDGE concept for enhanced data rates in GSM and TDMA/136," in *Proc. IEEE WCNC*, Sep. 1999, vol. 2, pp. 752–756.
- [25] D. Krishnaswamy, "Network-assisted link adaptation with power control and channel re-assignment in wireless networks," in *Proc. 3G Wireless Conf.*, May 2002, pp. 165–170.
- [26] V. Paxson and M. Allman, "Computing TCP's retransmission timer," RFC 2988, Nov. 2000.
- [27] D. Qiao and S. Choi, "Goodput analysis and link adaptation for IEEE 802.11a wireless LANs," *IEEE Trans. Mobile Comput.*, vol. 1, no. 4, pp. 278–292, Oct. 2002.
- [28] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11, 1997.
- [29] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification-Amendment 1: High Speed Physical Layer in the 5 GHz band*, IEEE Std. 802.11a, 1999.



Jatinder Pal Singh is a Senior Research Scientist at Deutsche Telekom Laboratories, Berlin, where he is currently working in the areas of resource allocation for heterogeneous network access, controlled and secure sharing of broadband access infrastructure, and cross-layer design of wireless networks. He received his Ph.D. and M.S. degrees in Electrical Engineering from Stanford University in 2005 and 2002 respectively. He received his B.S. in Electrical Engineering from the Indian Institute of Technology, Delhi in 2000. He interned with IBM T.J. Watson

Research Center, Hawthorne, NY; Robert Bosch Corporation, Palo Alto, CA; Intel Corporation, San Jose, CA; and IBM India Research Labs, New Delhi, during the summers of 2003, 2002, 2001, and 2000 respectively. He was named P. Michael Farnwald fellow under the aegis of Stanford Graduate Fellowship award from 2000–2004. In 2005, he was awarded the Deutsche Telekom fellowship.

His research has spanned cross-layer design issues in wireless networks, routing in ad-hoc networks, performance evaluation of 802.11 based WLANs, application layer multicast, and transport layer optimization via adaptive link-layer techniques. He has served as a reviewer for journals and conferences including IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and INFOCOM. He has authored patents (granted and pending) in the areas of TCP throughput optimization, central scheduling algorithms for IEEE 802.11e networks, and peer-to-peer media streaming.



Yan Li received his B.S. degree in Electrical Engineering and Computer Science from University of California at Berkeley, Berkeley CA, in 2001, and his M.S. and Ph.D. degrees in Electrical Engineering from Stanford University, Stanford CA, in 2003 and 2006, respectively. He is currently working at Qualcomm Inc., Campbell CA. His research interests include supporting multimedia streaming over wireless networks and performance engineering in ad hoc networks.



Nicholas Bambos received his PhD in EECS from the University of California at Berkeley in 1989 and graduated in EE from the National Technical University of Athens-Greece in 1984. He is currently a Professor at Stanford University, having a joint appointment in the Department of Electrical Engineering and the Department of Management Science and Engineering. He heads the Network Architecture and Performance Engineering research group, conducting research in wireless/wireline network architectures, the Internet infrastructure, network management and information service engineering.

He has held the Cisco Systems Faculty Development Chair (1999-2003) in computer networking at Stanford and has won the IBM Faculty Award (2002) for high-impact research in performance engineering of computer systems and networks. He has been the David Morgenthaler Faculty Scholar (1996-99) at Stanford, and has received the National Young Investigator Award (1992) from the National Science Foundation (NSF) for research in computer networks and distributed computing architectures, as well as the NSF Research Initiation Award (1990) for studies in performance modeling of computer systems.

His current technology research interests include high-performance networking, autonomic computing, and service engineering. His methodological interests are in network control, online task scheduling, queuing systems and stochastic processing networks.



Ahmad Bahai received his MS degree from Imperial College, University of London in 1988 and Ph.D. degree from University of California at Berkeley in 1993, all in Electrical Engineering. From 1992 to 1994 he worked as a member of technical staff in the wireless communications division of TCSI. He joined AT&T Bell Laboratories in 1994 where he was Technical Manager of Wireless Communication Group in Advanced Communications Technology Labs until 1997. He was involved in research and design of several wireless standards such as PDC,

IS-95, GSM, and IS-136 terminals and Base stations, as well as ADSL and Cable modems. He is one of the inventors of Multi-carrier CDMA (OFDM) concept and proposed the technology for high speed wireless data systems. He was the co-founder and Chief Technical Officer of ALGOREX Inc. and currently is a Fellow and the Chief Technologist of National Semiconductor. He is an adjunct/consulting professor at Stanford University and University of California, Berkeley. His research interest includes adaptive signal processing and communication theory. He is the author of more than 50 papers and reports and his book on "Multi-carrier Digital Communications" is published by Kluwer/Plenum. He holds nine patents in Communications and Signal Processing field and served as an editor of IEEE COMMUNICATION LETTERS.



Bangnan Xu is a Research Expert at T-Systems, Deutsche Telekom, leading various research projects in ad hoc/mesh networks, seamless IP mobility, fixed and mobile convergence, and next generation access networks. His current research interests include Next Generation Mobile Network (NGMN, 3GPP LTE SAE), Next Generation Network (NGN, ETSI TISPAN), IMS and beyond, IP optimised fixed and mobile converged (FMC), and Quadruple play services. He received his Ph.D. in Electrical Engineering and Information Technology from Aachen Univ. of Technology (2002), as well as M.Sc. (1989) and B.Sc. (1986) degrees in Electrical Engineering from Dalian Maritime University. From 1989 to 1996, he was a lecturer on communication systems in Dalian Maritime University. Before joining T-Systems in 2001, he was with the chair of Communication Networks, Prof. Dr.-Ing. B. Walke, Aachen Univ. of Technology, Aachen, Germany. He is one of the inventors of W-CHAMB concept and holds several patents in multihop communications, ad hoc routing, and seamless IP mobility. He has published many technical papers, three of them with best paper awards.



Gerd Zimmermann received his Dipl.-Ing. degree in Electrical Engineering in 1988 from University of Kaiserslautern in Germany. From 1988 to 1994 he was with the Department of Communication Engineering (with Prof. Rupprecht) at the same university and received his Ph.D. degree in the area of receiver structures for the GSM mobile radio system. In 1994 he joined the Research Center of Deutsche Telekom in Darmstadt, now integrated into Deutsche Telekom's subsidiary T-Systems, where he held the position of a senior expert and project

manager for mobile and wireless solutions. At Deutsche Telekom he is involved in projects on digital broadcasting, digital mobile radio systems and mobile network convergence, and also as member of national and international regulatory bodies and consortia like ITU-R, WorldDAB and DRM (Digital Radio Mondiale). His current interests are in radio network planning and provisioning of IP services in vehicles for public transportation like high speed trains.