TCP Performance in Wireless Access with Adaptive Modulation and Coding

Qingwen Liu*, Shengli Zhou† and Georgios B. Giannakis*
* Dept. of ECE, Univ. of Minnesota, Minneapolis, MN 55455, USA
† Dept. of ECE, Univ. of Connecticut, Storrs, CT 06269, USA

***Abstract*— We study a wireless access system with adaptive modulation and coding (AMC) at the physical layer, finite-length queuing at the data link layer and a TCP protocol at the transport layer. We analyze the end-to-end TCP performance via a fixed-point procedure, that effectively couples TCP with the AMC-based wireless link. Guided by the performance analysis, we present a simple cross-layer design, which optimizes the target packet error rate in AMC at the physical layer, so that the TCP throughput at the transport layer is maximized.**

I. Introduction

Wireless access is the "bottleneck" in wired-wireless networks, not only because wireless resources (bandwidth and power) are more scarce and expensive, but also because the overall system performance degrades markedly due to multipath fading, Doppler, and time-dispersive effects introduced by the wireless propagation. In order to enhance spectral efficiency while adhering to a target error performance over wireless channels, adaptive modulation and coding (AMC) schemes have been widely used to match transmission parameters to time-varying channel conditions (see e.g., [1], and references therein).

However, most existing AMC designs are considered at the physical layer. Their impact on, and the interaction with, higher protocol layers remain largely un-resolved. We have developed a cross-layer design combining AMC with truncated automatic-repeat-request (ARQ) in [5], and investigated the interaction of AMC with finite-length queuing in [4]. On the other hand, performance of the transport control protocol (TCP) at the transport layer has been extensively studied in both wired and wireless settings with fixed modulation and coding (see e.g., [2], [6] and references therein). The coupling of TCP with adaptive forward error correction has been recently investigated in [3].

In this paper, we study an end-to-end connection equipped with AMC at the physical layer, finite-length queuing at the data link layer and the TCP protocol at the transport layer. We rely on appropriate analytical models for TCP and the wireless link; and analyze the system performance via a fixed-point procedure [2]. Based on the performance analysis, we present a simple cross-layer design, which optimizes the target packet error rate of AMC at the physical layer, in order to maximize the TCP throughput at the transport layer, when coupled with finite-length queuing at the data link layer.

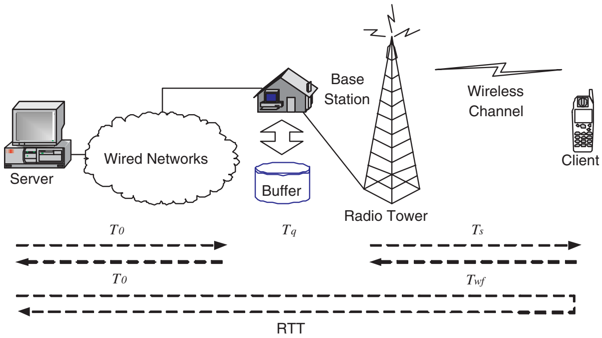

Fig. 1. An end-to-end wired-wireless connection

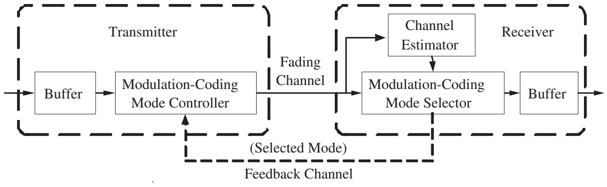

Fig. 2. The wireless link with combined queuing and AMC

II. System Model

A. System Description

Fig. 1 illustrates an end-to-end connection between a server (source) and a client (destination), which includes a wireless link with a single-transmit and a single-receive antenna. As depicted in Fig. 2, a queue (buffer) is implemented at the base station of the wireless link, and operates in a first-in-first-out (FIFO) mode. The AMC controller follows the queue at the base station (transmitter), and the AMC selector is implemented at the client (receiver). The layer structure of the system under consideration and the processing units at each layer are shown in Fig. 3.

At the *physical layer* of the wireless link, we assume that multiple transmission modes are available, with each mode representing a pair of a specific modulation format, and a forward error correcting (FEC) code, as in the HIPERLAN/2 and the IEEE 802.11a standards. Based on channel estimation at the receiver, the AMC selector determines the modulation-coding pair (mode), which is sent back to the transmitter through a feedback channel, for the AMC controller to update the transmission mode. Coherent demodulation and maximum-likelihood (ML) decoding are employed at the receiver. The decoded bit streams are mapped to packets, which are pushed upward to the data link layer.

We consider the following group of transmission modes:
TM: Convolutionally coded M_n-ary rectangular/square QAM, adopted from the HIPERLAN/2, or, the IEEE 802.11a stan-

The work by Q. Liu and G. B. Giannakis are prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon. The work by S. Zhou is supported by UConn Research Foundation internal grant 445157.

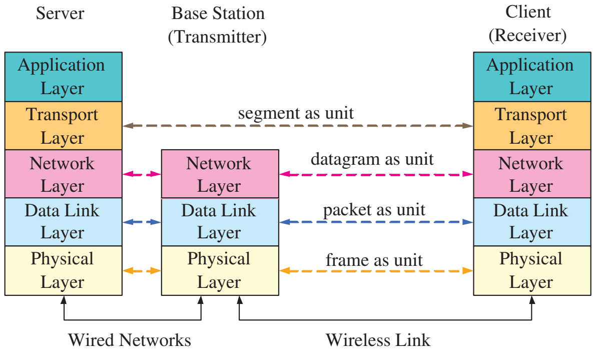

Fig. 3. The cross-layer structure

dards, which are listed under Table I, in a rate ascending order. Although we focus on TM in this paper, other transmission modes can be similarly constructed.

At the *data link layer* of the base station (transmitter), the queue has finite-length (capacity) of K packets. The queue is served by the AMC module at the physical layer. The customers of the queue are packets.

At the *network layer*, we will not deal with routing issues. At the base station, the arrival process of the datagram stream is assumed to be independent of the AMC and queue status.

At the *transport layer* of the server and the client, the TCP Reno protocols are implemented, as in [6]. We assume that the reader is familiar with the TCP Reno "congestion control" mechanism. Here, only the triple-duplicate acknowledgment (ACK) based congestion control is investigated. Other issues such as time-out, window-size limitation and out-of-sequence effects go beyond the scope of this paper.

We next detail the frame, packet, datagram and segment structures, as depicted in Fig. 4:

i) At the *physical layer*, the data are transmitted frame by frame through the wireless link, where each frame contains a fixed number of symbols (N_s). Given a fixed symbol rate, the frame duration (T_f seconds) is constant, and represents the time-unit throughout this paper. Each frame at the physical layer may contain one or more packets coming from the data link layer.

ii) At the *data link layer*, each packet contains a fixed number of bits (N_b), which include packet header, payload, and cyclic redundancy check (CRC) bits. After modulation and coding with mode n of rate R_n (bits/symbol) at the base station, each packet is mapped to a symbol-block containing N_b/R_n symbols. Multiple such blocks, together with N_c pilot symbols and control parts, constitute one frame to be transmitted, as in the HIPERLAN/2 and the IEEE 802.11a standards. If mode n is used, it follows that the number of symbols per frame is $N_s = N_c + N_p N_b / R_n$, which implies that N_p (the number of packets per frame) depends on the chosen mode.

iii) At the *network layer*, each datagram has fixed length of bytes including header and payload, which is contained in only one packet at the data link layer.

iv) At the *transport layer*, each segment contains a fixed number of bytes, which is transported by one datagram through the network.

We next list our operating assumptions:

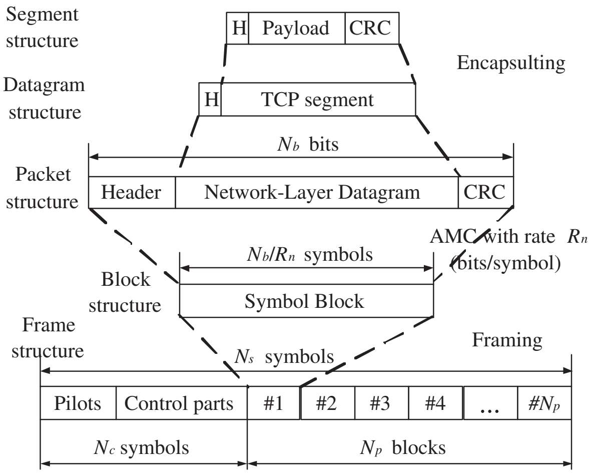

Fig. 4. The processing units at each layer

A1: The channel is frequency flat, and remains invariant per frame, but is allowed to vary from frame to frame. This corresponds to a block fading channel model, which is suitable for slowly-varying wireless channels. As a consequence, AMC is adjusted on a frame-by-frame basis.

A2: Perfect CSI is available at the receiver relying on training-based channel estimation. The corresponding mode selection is fed back to the transmitter without error and latency [1].

A3: If the queue is full, the additional arriving packets will be dropped, so that the overflow content is lost.

A4: Error detection based on CRC is perfect, provided that sufficiently reliable error detection CRC codes are used.

A5: If a packet is received incorrectly at the client after error detection, we declare loss of its encapsulated datagram, as well as the corresponding segment. Then, the triple-duplicate ACKs are generated, and assumed to be fed back to the server without error and queuing delay at the base station [2].

A6: Packet loss in the wired networks is negligible compared with that over the wireless link [4].

For flat fading channels adhering to A1, the channel quality can be captured by a single parameter, namely the received signal-to-noise ratio (SNR) γ. Since the channel varies from frame to frame, we adopt the general Nakagami-m model to describe γ statistically [1]. The received SNR γ per frame is thus a random variable with a Gamma probability density function:

$$p_\gamma(\gamma) = \frac{m^m \gamma^{m-1}}{\bar{\gamma}^m \Gamma(m)} \exp\left(-\frac{m\gamma}{\bar{\gamma}}\right), \tag{1}$$

where $\bar{\gamma} := E\{\gamma\}$ is the average received SNR, $\Gamma(m) := \int_0^\infty t^{m-1} e^{-t} dt$ is the Gamma function, and m is the Nakagami fading parameter ($m \geq 1/2$).

B. Adaptive Modulation and Coding

The objective of AMC is to maximize the data rate by adjusting transmission parameters to channel variations, while maintaining a prescribed packet error rate P_0. Let N denote the total number of transmission modes available ($N = 5$ for TM). As in [1], we assume constant power transmission, and partition the entire SNR range into $N+1$ non-overlapping consecutive intervals, with boundary points denoted as $\{\gamma_n\}_{n=0}^{N+1}$.

TABLE I

TRANSMISSION MODES WITH CONVOLUTIONALLY CODED MODULATION

	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
Modulation	BPSK	QPSK	QPSK	16-QAM	64-QAM
Coding Rate R_c	1/2	1/2	3/4	3/4	3/4
R_n (bits/sym.)	0.50	1.00	1.50	3.00	4.50
a_n	274.7229	90.2514	67.6181	53.3987	35.3508
g_n	7.9932	3.4998	1.6883	0.3756	0.0900
γ_{pn}(dB)	-1.5331	1.0942	3.9722	10.2488	15.9784

(The generator polynomial of the mother code is $g = [133, 171]$.)

In this case,

$$\text{mode } n \text{ is chosen,} \quad \text{when } \gamma \in [\gamma_n, \gamma_{n+1}). \tag{2}$$

To avoid deep channel fades, no data are sent when $\gamma_0 \leq \gamma < \gamma_1$, which corresponds to the mode $n = 0$ with rate $R_0 = 0$ (bits/symbol). The design objective for AMC is to determine the boundary points $\{\gamma_n\}_{n=0}^{N+1}$.

For simplicity, we approximate the instantaneous packet error rate (PER) as [5, eq. (5)]:

$$\mathrm{PER}_n(\gamma) \approx \begin{cases} 1, & \text{if } \ 0 < \gamma < \gamma_{pn}, \\ a_n \exp\left(-g_n \gamma\right), & \text{if } \ \gamma \geq \gamma_{pn}, \end{cases} \tag{3}$$

where n is the mode index, γ is the received SNR, and the mode-dependent parameters a_n, g_n, and γ_{pn} are obtained by fitting (3) to the exact PER [5]. With packet length $N_b = 1,080$, the fitting parameters for TM are provided in Table I [5]. Based on (1) and (2), the mode n will be chosen with probability [1, eq. (34)]:

$$\Pr(n) = \int_{\gamma_n}^{\gamma_{n+1}} p_\gamma(\gamma) d\gamma = \frac{\Gamma(m, m\gamma_n/\bar{\gamma}) - \Gamma(m, m\gamma_{n+1}/\bar{\gamma})}{\Gamma(m)}, \tag{4}$$

where $\Gamma(m, x) := \int_x^\infty t^{m-1} e^{-t} dt$ is the complementary incomplete Gamma function. Let $\overline{\mathrm{PER}}_n$ denote the average PER corresponding to mode n. In practice, we have $\gamma_n > \gamma_{pn}$, and thus obtain $\overline{\mathrm{PER}}_n$ in closed-form as (c.f. [1, eq.(37)]):

$$\begin{aligned} \overline{\mathrm{PER}}_n &= \frac{1}{\Pr(n)} \int_{\gamma_n}^{\gamma_{n+1}} a_n \exp(-g_n \gamma) p_\gamma(\gamma) d\gamma \qquad (5) \\ &= \frac{1}{\Pr(n)} \frac{a_n}{\Gamma(m)} \left(\frac{m}{\bar{\gamma}}\right)^m \frac{\Gamma(m, b_n \gamma_n) - \Gamma(m, b_n \gamma_{n+1})}{(b_n)^m}, \end{aligned}$$

where $b_n := m/\bar{\gamma} + g_n$. The average PER of AMC can then be computed as the ratio of the average number of packets in error over the total average number of transmitted packets [1]:

$$\overline{\mathrm{PER}} = \frac{\sum_{n=1}^{N} R_n \Pr(n) \overline{\mathrm{PER}}_n}{\sum_{n=1}^{N} R_n \Pr(n)}. \tag{6}$$

We want to find the thresholds $\{\gamma_n\}_{n=0}^{N+1}$, so that the prescribed P_0 is achieved for each mode: $\overline{\mathrm{PER}}_n = P_0$, which naturally leads to $\overline{\mathrm{PER}} = P_0$ based on (6). Given P_0, $\bar{\gamma}$, and m, the following threshold searching algorithm determines $\{\gamma_n\}_{n=0}^{N+1}$ and guarantees that $\overline{\mathrm{PER}}_n$ is exactly P_0 [4]:

Step 1: Set $n = N$, and $\gamma_{N+1} = +\infty$.

Step 2: Search the unique $\gamma_n \in [0, \gamma_{n+1}]$ that satisfies:

$$\overline{\mathrm{PER}}_n = P_0 \ . \tag{7}$$

Step 3: If $n > 1$, set $n = n - 1$ and go to Step 2; otherwise, go to Step 4.

Step 4: Set $\gamma_0 = 0$.

The SNR region $[\gamma_n, \gamma_{n+1})$ corresponding to transmission mode n constitutes the channel state indexed by n. To describe the transition of these channel states, we rely on a finite state Markov chain (FSMC) model, which we develop next.

C. Finite State Markov Chain Channel Model

As in [4], we adopt an FSMC channel model to analyze the performance of our system. Assuming slow fading conditions so that transition happens only between adjacent states, the probability of transition exceeding two consecutive states is zero; i.e.,

$$P_{l,n} = 0, \qquad |l - n| \geq 2. \tag{8}$$

The adjacent-state transition probability can be determined by:

$$\begin{aligned} P_{n,n+1} &\approx \frac{N_{n+1} T_f}{\Pr(n)}, \quad \text{if } n = 0, \ldots, N-1, \\ P_{n,n-1} &\approx \frac{N_n T_f}{\Pr(n)}, \qquad \text{if } n = 1, \ldots, N, \end{aligned} \tag{9}$$

where N_n is the cross-rate of mode n (either upward or downward), which can be estimated as:

$$N_n = \sqrt{2\pi \frac{m\gamma_n}{\bar{\gamma}}} \frac{f_d}{\Gamma(m)} \left(\frac{m\gamma_n}{\bar{\gamma}}\right)^{m-1} \exp\left(-\frac{m\gamma_n}{\bar{\gamma}}\right), \tag{10}$$

where f_d denotes the mobility-induced Doppler spread. The probability of staying at the same state n is:

$$P_{n,n} = \begin{cases} 1 - P_{n,n+1} - P_{n,n-1}, & \text{if } \ 0 < n < N, \\ 1 - P_{0,1}, & \text{if } \ n = 0, \\ 1 - P_{N,N-1}, & \text{if } \ n = N. \end{cases} \tag{11}$$

In summary, we model the channel as an FSMC with an $(N+1) \times (N+1)$ state transition matrix, as in [4, eq. (9)]:

$$\mathbf{P}_c = [P_{i,j}]_{(N+1)\times(N+1)}. \tag{12}$$

III. PERFORMANCE ANALYSIS

To analyze the system performance, we will rely on a fixed-point procedure, that has already been used for studying TCP traffic both in wired environments and in wireless settings [2]. This procedure is illustrated in Fig. 5, where the TCP model is coupled with the wireless link (WL) model. Having as input the traffic rate B at which data arrive from the transport layer, the WL model yields as outputs the average delay per segment over the wireless link T_{wl} and the segment loss rate

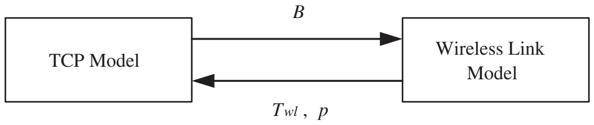

Fig. 5. The fixed-point analytical model

p. These two key parameters are given as inputs to the TCP model, based on which we then derive new estimates of the traffic arrival rate for the WL model. The procedure is repeated until convergence on the parameter estimates is reached. The steady-state behavior of the system can then be analyzed. We next detail the analytical models for the TCP and WL.

A. *The TCP Model*

We adopt the TCP model established in [6]. The TCP segment sending rate (the average number of segments sent from the server per time-unit) is approximated as [6, eq. (30)]:

$$B \approx \frac{1}{\mathrm{RTT}}\sqrt{\frac{3}{2bp}}\ , \tag{13}$$

where RTT is the average end-to-end round-trip-time; $b = 2$ is the number of segments that are acknowledged by a received ACK; and p is the segment loss rate. Based on the TCP retransmission mechanism, the average number of transmissions per segment is:

$$N_r = \sum_{k=1}^{\infty} kp^{k-1}(1-p) = \frac{1}{1-p}. \tag{14}$$

Based on (13) and (14), the TCP throughput is:

$$\eta = \frac{B}{N_r} = B(1-p) = \frac{1-p}{\mathrm{RTT}}\sqrt{\frac{3}{2bp}}. \tag{15}$$

It is clear from (15) that η is a function of p and RTT.

As depicted in Fig. 1, the RTT can be approximated as (c.f. [2, eq. (22)]):

$$\mathrm{RTT} \approx 2T_0 + \underbrace{(T_q + T_s)}_{:=T_{wl}} + T_{wf} \tag{16}$$

where $2T_0$ is the average delay (forward and backward) over the wireline part within the end-to-end connection; T_q is the average waiting time per segment in the queue at the base station; T_s is the average transmission time per segment over the wireless channel, i.e., the average service time per segment upon the queue; T_{wf} is the feedback delay for an ACK over the wireless channel from the client to the base station, which includes the processing, scheduling, and queuing delays; and T_{wl} is the average transmission delay per segment over the wireless link, which includes waiting and service time. As in [2], we assume T_0 and T_{wf} are known constants, and thus only T_{wl} is left to determine from our following WL model.

B. *The Wireless Link Model*

1) Model of Queuing Service Process based on AMC: Different from non-adaptive modulations, AMC dictates a dynamic, rather than deterministic, service process for the queue, with a variable number of packets transmitted per time unit. Let t index the time units, and C_t (packets/time-unit) denote the number of packets transmitted using AMC at time t. Corresponding to each transmission mode n, let c_n (packets/time-unit) denote the number of packets transmitted per time-unit. We then have:

$$C_t \in \mathcal{C}, \qquad \mathcal{C} := \{c_0, c_1, \ldots, c_N\}\ , \tag{17}$$

where c_n takes positive integer values. Suppose that for the rate $R = 1$ transmission mode (e.g., Mode 2 in TM), a total of d packets can be accommodated per frame. We then have $c_n = dR_n$, where d is up to the designer's choice.

As specified in (17), the AMC module yields a queue server with a total of $N+1$ states $\{c_n\}_{n=0}^{N}$, with the service process C_t representing the evolution of server states. Since the AMC mode n is chosen when the channel enters the state n, we model the service process C_t as an FSMC with transition matrix given by (12).

2) Queuing Analysis: Having modeled the queuing service process, we now focus on the queue itself. Let U_t denote the queue state (the number of packets in the queue) at the end of time-unit t, or, at the beginning of time-unit $t+1$. Let A_t denote the number of packets arriving at time t. It is clear that $U_t \in \mathcal{U} := \{0, 1, \ldots, K\}$, and $A_t \in \mathcal{A} := \{0, 1, \ldots, \infty\}$. Here, A_t only needs to be stationary and independent of U_t and C_t. For convenience, we assume that A_t is Poisson distributed with parameter λ:

$$P(A_t = a) = \frac{\lambda^a \exp(-\lambda)}{a!}, \qquad a \geq 0, \tag{18}$$

where the ensemble-average $E\{A_t\} = \lambda = B$ is determined by the TCP sending rate.

Let (U_{t-1}, C_t) denote the pair of queue and server states, whose variation is modeled as an augmented FSMC [4]. We have proved that the stationary distribution of (U_{t-1}, C_t) exists and is unique; see [4, eq. (19)] for the calculation of the stationary distribution denoted as:

$$P(U = u, C = c) := \lim_{t\to\infty} P(U_{t-1} = u, C_t = c). \tag{19}$$

3) Wireless Link Performance: We are now ready to evaluate the segment loss rate p and the delay T_{wl} when finite-length queuing is coupled with AMC.

Since each segment is carried by one datagram and each datagram is encapsulated into one packet, the segment loss rate p equals the packet loss rate (the ratio of the number of incorrectly received packets at the client over those arriving upon the queue at the base station). Let P_d denote the packet dropping (overflow or blocking) probability upon the queue. Based on $P(A_t = a)$ in (18) and $P(U = u, C = c)$ in (19), we can readily compute P_d, as illustrated in [4, eq. (28)]. A packet is correctly received by the client, only if it is not dropped from the queue (with probability $1 - P_d$), and is correctly received through the wireless channel (with probability $1 - P_0$). Hence, we can express the segment loss rate, as in [4, eq. (22)]:

$$p = 1 - (1 - P_d)(1 - P_0). \tag{20}$$

We now derive the average delay over the wireless link T_{wl}. With the stationary distribution $P(U = u, C = c)$ in (19), we can derive the average number of segments in the wireless link (both in the queue and in transmission) as:

$$N_{wl} = \sum_{u\in\mathcal{U},c\in\mathcal{C}} u\cdot P(U=u,C=c) + \sum_{u\in\mathcal{U},c\in\mathcal{C}} \min\{u,c\}\cdot P(U=u,C=c). \tag{21}$$

Based on Little's Theorem, the average delay per segment through the wireless link can be calculated as:

$$T_{wl} = \frac{N_{wl}}{B(1-P_d)}. \tag{22}$$

In summary, given a target packet error rate P_0, Doppler spread f_d, average SNR $\bar{\gamma}$, Nakagami parameter m, queue length K and TCP sending rate B, we can obtain the system performance of the wireless link analytically through the following steps:
1) Determine the boundary points of AMC $\{\gamma_n\}_{n=0}^{N+1}$ by the threshold searching algorithm.
2) Build the transition matrix $\mathbf{P}_c$ in (12) for the channel as well as the queue server states.
3) Compute the stationary distribution $P(U=u,C=c)$ as in (19).
4) Calculate the segment loss rate p from (20), and the average delay T_{wl} from (22).

The p and T_{wl} in the wireless link model can then be used by the TCP model in Section III-A. Through the fixed-point procedure, we are now able to evaluate the end-to-end system performance, as will be tested in Section V.

IV. Cross-Layer Optimization

Based on our analysis so far, we notice that the end-to-end TCP throughput and delay depend on the parameters P_0, K, f_d, $\bar{\gamma}$, m, T_0, and T_{wf}. The parameters f_d, $\bar{\gamma}$, and m are channel dependent, while T_0 and T_{wf} are also decided by a given network environment. However, P_0 and K are design parameters, which can be adjusted to optimize the system performance via AMC design or scheduling.

To illustrate the parameter optimization, we now present a simple example. We maximize the TCP throughput η by tuning P_0 in AMC design at the physical layer. The steps are:
Step 1: Determine TCP throughput $\eta(P_0)$ via the fixed-point procedure for each $P_0 \in \mathcal{P}$, where $\mathcal{P}$ is the set of possible target PER values.
Step 2: Select the optimal P_0 as:

$$P_0^{\text{opt}} = \arg\max_{P_0\in\mathcal{P}} \eta(P_0)\ . \tag{23}$$

These steps are repeated, each time the parameters are updated.

This cross-layer design has the following benefits:
i) The end-to-end TCP throughput improves relative to the case where P_0 is set "blindly" at the physical layer without taking into account upper layers;
ii) It requires minimal cross-layer information, because only T_0 at the transport layer and K at the data link layer need to be transferred to the physical layer to determine P_0^{opt};
iii) It has low complexity, because P_0^{opt} only needs to be updated based on slow-varying parameters, i.e., K, f_d, $\bar{\gamma}$, m, T_0 and T_{wf}; and

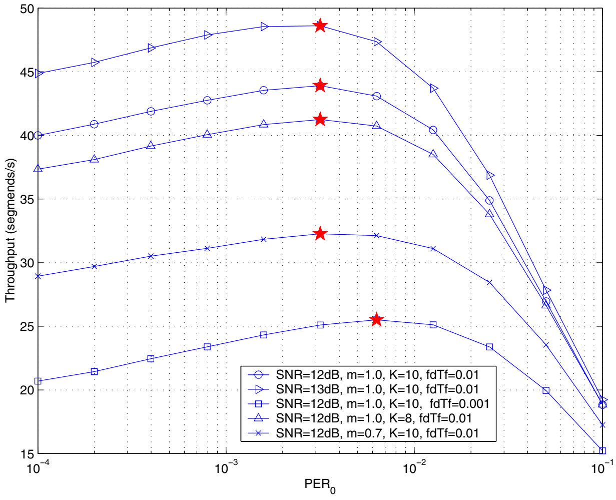

Fig. 6. TCP throughput vs. target PER (stars denote the maximum value)

iv) It is compatible with separate-layer designs, because it can be implemented in existing systems by simply adding the corresponding functions for cross-layer information exchange. In summary, the cross-layer design improves TCP throughput, requires minimal cross-layer information, has low-complexity and is backward compatible.

V. Numerical Results

We assume that $d=2$, $T_f=10$ (ms), $T_0=50$ (ms) and $T_{wf}=3$ (ms), respectively [6]. As a reference, we fix a basic set of parameters as: average SNR $\bar{\gamma}=12$ (dB), normalized Doppler frequency $f_dT_f=0.01$, buffer length $K=10$ and Nakagami parameter $m=1.0$. We plot the curves of the TCP throughput η (segments/second) versus the target packet error rate P_0 in Fig. 6. We then modify only one parameter from the reference parameters each time: average SNR $\bar{\gamma}=13$ (dB), normalized Doppler frequency $f_dT_f=0.001$, buffer length $K=8$ and Nakagami parameter $m=0.7$, as shown in Fig. 6, respectively. On each curve, the maximum value of η is depicted by star; and the corresponding P_0 is the solution of the cross layer design in (23).

From Fig. 6, the TCP throughput improvement by using the cross-layer design and the effects of different parameters on TCP throughput are demonstrated clearly.

References

[1] M. S. Alouini and A. J. Goldsmith, "Adaptive modulation over Nakagami fading channels," *Kluwer Journal on Wireless Commun.*, vol. 13, no. 1-2, pp. 119–143, May 2000.
[2] C. F. Chiasserini and M. Meo, "A reconfigurable protocol setting to improve TCP over wireless," *IEEE Trans. on Veh. Tech.*, vol. 51, no. 6, pp. 1608–1620, Nov. 2002.
[3] B. Liu, D. L. Goeckel, and D. Towsley, "TCP-cognizant adaptive forward error correction in wireless networks", in *Proc. of GLOBECOM*, vol. 3, pp. 2128–2132, Nov. 2002.
[4] Q. Liu, S. Zhou, and G. B. Giannakis, "Queuing with adaptive modulation and coding over wireless links," in *Proc. of MILCOM Conf.*, Boston, MA, October 13-16 2003 (invited).
[5] Q. Liu, S. Zhou, and G. B. Giannakis, "Cross-layer combining of adaptive modulation and coding with truncated ARQ over wireless links," *IEEE Trans. on Wireless Commun.*, 2004.
[6] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, "Modeling TCP Reno performance: a simple model and its empirical validation," *IEEE/ACM Trans. on Networking*, vol. 8, no. 2, pp. 133–145, Apr. 2000.