

Improving Wireless TCP Throughput by a Novel TCM-Based Hybrid ARQ

Qian Huang, Sammy Chan, *Member, IEEE*, Li Ping, *Senior Member, IEEE*, and Moshe Zukerman, *Fellow, IEEE*

Abstract—A novel hybrid ARQ (HARQ) scheme using a concatenated two-state trellis-coded modulation (CT-TCM) code is proposed for improving wireless TCP throughput. A distinguished feature of the proposed scheme is that the heavily punctured TCM codes are used for retransmissions of the corrupted data block, which are combined at the receiver with the previously received sequences of the same data block for decoding. By this method, significantly improved coding gain and efficient spectrum utilization can be achieved with very low complexity. A Markov model is developed to evaluate TCP throughput over the proposed HARQ in wireless link. By both analysis and simulation, we demonstrate that compared with other existing TCM-based ARQ schemes, significant improvement of TCP throughput over wireless links is achieved by the proposed CT-TCM HARQ while smaller buffer size is required at the access point.

Index Terms—Hybrid automatic repeat request (ARQ), trellis-coded modulation (TCM), wireless TCP, performance analysis.

I. INTRODUCTION

TECHNIQUES that guarantee efficient and reliable data transmission in wireless networks are crucial for various emerging wireless applications, including wireless Transmission Control Protocol (TCP) applications. In wired networks, TCP supports end-to-end reliable data transmission and enables congestion control, which assumes that wired links are stable and reliable enough that most packet losses are due to network congestion. Once packet loss is detected, TCP source reduces its window size to avoid further network congestion and immediately retransmits the lost packets. In wireless networks, however, packet losses are affected by the characteristics such as high bit error rate fading, scarce bandwidth and mobility. If TCP directly runs over wireless networks, the packet loss due to wireless channel errors will be treated as same as the loss due to network congestion and unnecessary congestion control will be invoked, i.e., data transmission rate is reduced when it is unnecessary. It results in serious degradation of the end-to-end throughput [1].

Manuscript received November 2, 2005; revised February 16, 2006; accepted March 10, 2006. The associate editor coordinating the review of this paper and approving it for publication was H.-H. Chen. This work was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 1190/02E) and the Australian Research Council.

Q. Huang, S. Chan, and Li Ping are with the Department of Electronic Engineering, City University of Hong Kong, 83, Tat Chee Avenue, Kowloon, Hong Kong (e-mail: {eeqhuang, eeschan, eeliping}@cityu.edu.hk).

M. Zukerman is with the ARC Special Research Centre for Ultra-Broadband Information Networks, an affiliated program of National ICT Australia, Department of Electrical and Electronic Engineering, The University of Melbourne, Melbourne, Vic. 3010, Australia (e-mail: mzu@unimelb.edu.au).

Digital Object Identifier 10.1109/TWC.2007.05865.

Various methods have been proposed to overcome the above issue. One method is to distinguish the packet loss due to network congestion from the loss due to wireless channel errors. TCP sender will not invoke congestion control if the packet loss is detected due to wireless channel errors [2]–[8]. A cross-layer design between TCP and lower layers is considered in [2], [4], [5] to improve TCP performance in wireless links. Snoop TCP [4] makes use of the TCP layer knowledge (duplicate ACKs) such that lost packets are locally retransmitted in the link layer and duplicate ACKs to the TCP sender are suppressed to avoid data rate reduction. Freeze-TCP [5] and explicit-loss-notification (ELN)-enabled [2] TCP take advantage of the physical layer information to prevent the TCP sender from executing unnecessary congestion control if packet loss due to wireless channel errors is detected. The pros and cons of these solutions are summarized in Table I. We observe that most of these solutions maintain the end-to-end TCP semantics but require modifications on the current TCP congestion control mechanism.

Alternative solutions resort to using link layer error control schemes, such as local retransmission by automatic repeat request (ARQ), or the combination of ARQ and forward error correction (FEC), referred to as hybrid ARQ (HARQ) [9]–[11], to improve TCP performance over wireless links [2], [10], [11]. HARQ techniques have also been considered in the link layer function of future broadband wireless networks, e.g., the third generation of mobile networks (3G) and IEEE 802.16 WiMax [12]. The advantage of these link layer solutions is that a transparent transmission path over wireless links is seen by TCP packets, provided most wireless channel errors can be combatted by link layer error control schemes. However, the error correction capability of a HARQ scheme strongly depends on the underlying FEC code. A FEC code with weaker capability will cause more retransmissions before a packet is correctly received and accordingly longer packet delay over wireless links. Due to the retransmission mechanism of TCP, any failure of recovering packet losses in link layer before TCP timeouts would trigger the end-to-end TCP retransmission, which results in retransmission at both TCP and link layers. A direct solution to this problem is using a code with strong capability of error correction as the underlying FEC code, to increase the probability that a packet can be successfully received without triggering TCP timeouts. Trellis coded modulation (TCM) has demonstrated its attraction to broadband wireless communications due to its advantage in both power and bandwidth efficiency over wireless links [13]–[16]. As an important application of TCM codes over wireless links, various TCM-based HARQ schemes

TABLE I
COMPARISONS OF VARIOUS WIRELESS TCP SOLUTIONS.

Proposals	TCP semantics	TCP modification requirements	Cross-layer considerations	retransmission due to wireless errors
Indirect TCP [3]	split	wireless portion	no	local
Snoop TCP [2]	end-to-end	access point	yes	local
Freeze-TCP [5]	end-to-end	receiver side	yes	end-to-end
ECN-enabled TCP [6], TCP-Jersey [7]	end-to-end	sender, receiver and routers	no	end-to-end
ELN-enabled TCP [2]	end-to-end	sender, receiver	yes	end-to-end
TCP VenO [8]	end-to-end	sender side	no	end-to-end

have been investigated [17]–[21].

In this paper, we propose a novel HARQ scheme based on a recently developed concatenated two-state trellis-coded modulation (CT-TCM) codes [16]. We study the performance of the proposed HARQ with the CT-TCM codes and its impact on TCP throughput over wireless links. In Section II, we first provide a brief review of various TCM-based HARQ schemes and the CT-TCM codes, which is followed by a detailed description of the proposed HARQ in Section III. The analytical models to calculate the link layer and TCP throughput bounds of the proposed scheme over additive white Gaussian noise (AWGN) channels are presented in Section IV and verified by simulation in Section V, where the performance of our scheme is also evaluated by comparisons with other TCM-based HARQ schemes. Finally, we conclude our investigations in Section VI.

II. RELATED WORK ON TCM-BASED HARQ SCHEMES

A HARQ scheme with incremental-redundancy can achieve higher efficiency than a Type-I HARQ with repetition coding. However, most of existing TCM-based HARQ schemes [17]–[20] adopt the same repetition method as that used in Type-I HARQ due to the nature of TCM that it is difficult to separate the information bits and parity-check bits of a TCM code.

A comparison between different TCM-based HARQ schemes is investigated in [18]. All the schemes in [18] use the Turbo-based TCM (TTCM) code proposed in [13] with 8PSK modulation as the underlying FEC scheme. Scheme 1-TTCM and Scheme 2-TTCM are Type-I ARQ of [18] with repetition coding based on a rate 2/3 Turbo code in each retransmission. The difference between them is that in Scheme 2-TTCM, the receiver uses equal gain diversity combining to combine all copies of the received packet into a single packet of the same block size by adding the demodulated log-likelihood values of each packet. Scheme 3-TTCM of [18] is a Type-II HARQ with incremental redundancy. It uses a rate 1/3 Turbo code as the mother code mapped to an 8PSK constellation. The information bits are sent in the first transmission and the redundancy bits are incrementally sent in retransmissions. At the receiver, all bits received in retransmissions are combined with the bits received in the first transmission for decoding.

The scheme of [19] is also based on repetition coding, but it uses an improved combining strategy in decoding. At the receiver, it is proposed to use soft combining of the squared distances between the soft detector output and the possible

signals. If $\vec{r}_i = (r_i^{(1)}, r_i^{(2)}, \dots, r_i^{(N)})$ is the i -th received version of a packet and $r_i^{(n)}$ are the 2^{m+1} dimensional vectors representing the n -th symbol, the resulting packet obtained by combining L received versions of the packet is $\vec{z}_i = (z_i^{(1)}, z_i^{(2)}, \dots, z_i^{(N)})$ where $z_i^{(j)} = \sum_{l=1}^L r_i^{(j)}$. This combining strategy has better throughput performance than that of the average diversity combining scheme which simply averages the soft decision values of the coordinates of each received symbol in all packets.

In a recently proposed TCM-based HARQ scheme [20], two different TCM codes are used in transmission. For the first transmission, the message is encoded by one TCM code. If the received copy contains uncorrectable errors, the message is encoded by the other TCM code for retransmission. After two transmissions, the two TCM codes are used alternatively in retransmissions. This scheme improves the rate at which the squared free distance of the combined codes increases. However, it switches to a repetition mode after two transmissions.

Unlike the above methods, the HARQ scheme of [21] uses different pseudo-random interleaving pattern in each transmission for performance improvement. This scheme involves two identical Turbo encoders concatenated in parallel. Either of them generates a rate 1/2 Turbo code as the error correction code. For the initial transmission of a packet, the input information sequence is encoded by one of the Turbo encoders. If a retransmission is requested, the input information sequence is first interleaved by a pseudo-random interleaver, and then encoded by the other Turbo encoder and transmitted. For each additional retransmission, a different pseudo-random interleaving pattern is used so that the output weight during retransmissions is higher. At the receiver, the received retransmissions and the initial transmission of the data packet are combined for decoding.

Our proposed HARQ uses the CT-TCM codes in [16] as the underlying FEC scheme. Preliminary work in [16] demonstrates that the CT-TCM codes have very low complexity but comparable performance to some existing Turbo-type coded-modulation schemes [14], [15]. Fig. 1 presents the global structure of a CT-TCM encoder which comprises M component encoders concatenated in parallel by M symbol interleavers. Here, $\mathbf{d}_k = (d_{k,0}, \dots, d_{k,n-1})$ ($k \geq 0$) represents a binary n -tuple information symbol, $\{\mathbf{d}_k\}$ is the input information sequence; $\mathbf{z}^{(m)}$ denotes the symbol interleaver for component encoder m ($0 \leq m < M$). Each component encoder consists of a binary two-state trellis encoder

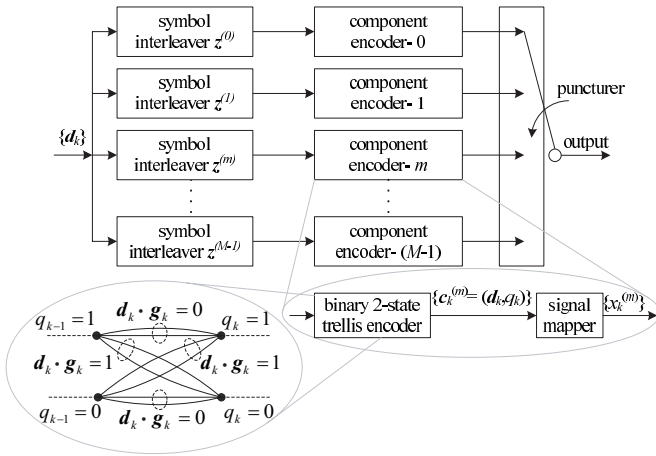


Fig. 1. The structure of a CT-TCM encoder with M component encoders.

characterized by the two-state trellis shown in Fig. 1 and followed by a multi-ary signal mapper. The coded symbol produced by component encoder m is denoted as $\mathbf{c}_k^{(m)}$, which contains a parity-check bit q_k specified in [16]. The coded symbol $\mathbf{c}_k^{(m)}$ is then mapped to a signal constellation of size 2^{n+1} , producing a modulated symbol $x_k^{(m)}$. A sequence of modulated symbols $\{x_k^{(m)}\}$ corresponding to information sequence $\{d_k\}$ is referred to as a CT-TCM component code.

For spectrum efficiency, a CT-TCM code is finally produced by heavily puncturing the modulated symbols of all component codes under a selected puncture pattern which satisfies the following constraints: (i) At any time $k \geq 0$, one and only one modulated symbol of $\{x_k^{(m)}, 0 \leq m < M\}$, from the component encoders, carrying the same information d_k , is selected and transmitted. The other modulated symbols are punctured; (ii) The punctured symbols are uniformly distributed in each component code. For the CT-TCM with M component encoders, the puncture pattern satisfying constraints (i) and (ii) can be represented as an $M \times M$ indication array Γ of which only one entry per row is one and the others are zero. For example, puncture pattern

$$\Gamma = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

is used for the 8PSK-based CT-TCM codes in [16]. When puncturing, at time instant k ($k \geq 0$), $x_k^{(m)}$ from component encoder m is selected and transmitted if the $(k \bmod M, m)$ th entry of Γ is 1, otherwise it is punctured. Thus, after puncturing only one of M components per modulated symbol is selected to be transmitted for an input information symbol. Simultaneously the modulated symbols that have not been punctured are uniformly distributed in each component code.

At the receiver, CT-TCM employs an iterative decoder based on a multidimensional Turbo decoder incorporating the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm [22]. Fig. 2 presents the structure of the CT-TCM decoder which comprises M local *a posteriori* probability (APP) decoders, one for each CT-TCM component code. The M local APP decoders operate successively. For the m th decoder, T is for delay of one

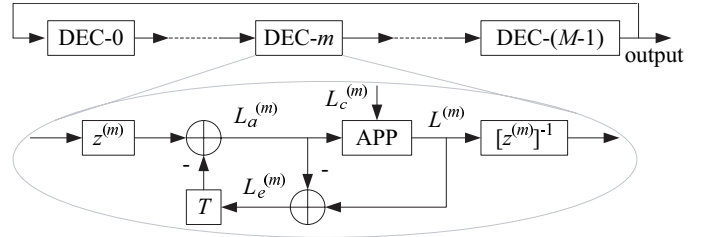


Fig. 2. The global decoder structures of a CT-TCM algorithm with M component encoders.

iteration, $\mathbf{z}^{(m)}$ is the symbol interleaver, and the other variables are the log-likelihood (LL) values and defined as follows.

$L^{(m)}$: *a posteriori* LL values for all information symbols after decoding the m th component code.

$L_c^{(m)}$: LL values $\{\log p(y_k|x_k)\}$ based on individual channel observations of the m th component code, where x_k is the k th modulated symbol and y_k is the k th received symbol; its elements $\{L_{c,k}^{(m)}\}$ are calculated as $L_{c,k}^{(m)} = \log p(y_k|x_k)$ for unpunctured symbols, $L_{c,k}^{(m)} = 0$ for punctured symbols.

$L_a^{(m)}$: *a priori* LL values for all information symbols for the m th component code. It is initialized to zeroes, implying no *a priori* information.

$L_e^{(m)}$: extrinsic information produced by the m th component code, defined by $L_e^{(m)} = L^{(m)} - L_a^{(m)}$.

III. CT-TCM BASED HARQ SCHEME

The proposed HARQ scheme utilizes the heavily puncturing feature of the CT-TCM codes. Instead of dropping the punctured parts of a CT-TCM code, we propose to store these punctured parts in a buffer and to use them in subsequent retransmissions when necessary. At the receiver, the received parts from different retransmissions can be combined together for decoding so as to achieve improved coding gain. This approach overcomes the limitation in traditional TCM-based HARQ schemes. Let s denote a link layer data block to be transmitted. Let γ_1 denote the puncture pattern used for the first transmission of s . The CT-TCM HARQ performs as follows.

Step 1: Initially, puncture pattern γ_1 is used to generate sequence s_1 for the first transmission of s . The punctured parts are stored in a buffer at the sender for retransmissions. If the received sequence s'_1 (s_1 plus channel noise) is successfully decoded, an acknowledgement (ACK) is returned to the sender. Otherwise, s'_1 is stored in a buffer at the receiver and a negative acknowledgement (NAK) is returned, then the ARQ scheme moves to *Step 2*.

Step i ($2 \leq i \leq M$): Another puncture pattern γ_i ($\gamma_i \neq \gamma_{i-1}$) is used to select a new sequence, denoted s_i , out of the punctured parts in the sender's buffer for the i th transmission of s . If no error is detected in the received sequence s'_i , an ACK is returned to the sender. Otherwise, s'_i is combined with all previously received sequences $\{s'_j : 1 \leq j \leq i-1\}$ for decoding based on the turbo decoding principle. If no error is detected in decoding the combined sequence, an ACK is returned. Otherwise, s'_i is stored in the receiver's buffer and an NAK is returned. Then, our scheme moves to *Step i+1* if

$i < M$, or returns to *Step 1* and the receiver buffer is cleared if $i = M$.

With respect to the aforementioned constraints (i) and (ii), a number of qualified puncture patterns can be found. Our preliminary work in [23] have demonstrated by simulation that the assignment of puncture patterns in each transmission attempt of a data block is strongly related to the coding gain obtained in retransmission. Therefore, a key issue of the CT-TCM HARQ is to find appropriate puncture patterns for each transmission attempt of a data block.

In order to achieve optimal performance in combined decoding at the CT-TCM receiver, it is generally preferred that the received information symbols of a data block are evenly distributed in each CT-TCM component code. Based on this preference, the puncture patterns used in two consecutive transmissions of a data block are preferably selected from the qualified puncture patterns whose combination satisfies constraint (ii). That is, if the combined puncture pattern is used for puncturing at the sender, the modulated symbols that have not been punctured are still uniformly distributed in each CT-TCM component code. For example, in the CT-TCM HARQ with $M = 4$, we use puncture pattern Γ_0 of (1) for the initial transmission of a data block.

$$\begin{aligned} \Gamma_0 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \Gamma_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \\ \Gamma_2 &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \Gamma_3 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \end{aligned} \quad (1)$$

With respect to the above selection rule, the puncture pattern used for the second transmission of the data block is preferred to be Γ_1 of (1). At the receiver, the combination of the received sequences with puncture patterns Γ_0 and Γ_1 is equivalent to a sequence generated by using puncture pattern

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

at the sender, which provides evenly puncturing on each CT-TCM component code. Similarly, we select puncture patterns Γ_2 and Γ_3 of (1) for the third and fourth transmission of the data block.

IV. THROUGHPUT ANALYSIS

A. Link Layer Throughput

Suppose a data block s transmitted over an AWGN channel via the CT-TCM HARQ. Denote $D_i^{(j)}$, $F_i^{(j)}$ the events that the received sequence of s is detected with errors and the combination of the currently and all previously received sequences for s is detected with errors by combined decoding, respectively, when s is transmitted by the i -th puncture pattern at the j -th time after it has been sequentially transmitted by previous $i-1$ puncture patterns $\gamma_1, \dots, \gamma_{i-1}$ for j times. Denote $P(D_i^{(j)})$, $P(F_i^{(j)})$ the probabilities that the events $D_i^{(j)}$, $F_i^{(j)}$

occur, respectively. It is assumed that one data block can be retransmitted j times till it is successfully received, $j \rightarrow \infty$. It is assumed that the error contained in any decoded sequence can be detected at the receiver. The average number of transmissions per data block over the wireless link, denoted Λ , is obtained by

$$\begin{aligned} \Lambda &= 1 + \sum_{j=1}^{\infty} \sum_{i=1}^M P(D_1^{(1)}, D_2^{(1)}, F_2^{(1)}, \dots, D_M^{(1)}, F_M^{(1)}, \dots, \\ &\quad D_1^{(j-1)}, D_2^{(j-1)}, F_2^{(j-1)}, \dots, D_M^{(j-1)}, F_M^{(j-1)}, \\ &\quad D_1^{(j)}, D_2^{(j)}, F_2^{(j)}, \dots, D_i^{(j)}, F_i^{(j)}). \end{aligned} \quad (2)$$

In (2), $P(D_1^{(1)}, \dots, F_M^{(1)}, \dots, D_1^{(j-1)}, \dots, F_M^{(j-1)}, D_1^{(j)}, \dots, F_i^{(j)})$ can be expressed by

$$\begin{aligned} &P(D_1^{(1)}, \dots, F_M^{(1)}, \dots, D_1^{(j-1)}, \dots, F_M^{(j-1)}, D_1^{(j)}, \dots, F_i^{(j)}) \\ &= \prod_{j=1}^{\infty} \prod_{i=1}^M P(F_i^{(j-1)} | D_1^{(1)}, D_2^{(1)}, F_2^{(1)}, \dots, D_M^{(1)}, F_M^{(1)}, \dots, \\ &\quad D_1^{(j-2)}, \dots, F_M^{(j-2)}, D_1^{(j-1)}, D_2^{(j-1)}, F_2^{(j-1)}, \dots, D_i^{(j-1)}) \cdot \\ &\quad P(D_i^{(j-1)} | D_1^{(1)}, D_2^{(1)}, F_2^{(1)}, \dots, D_M^{(1)}, F_M^{(1)}, \dots, D_1^{(j-2)}, \\ &\quad \dots, F_M^{(j-2)}, D_1^{(j-1)}, D_2^{(j-1)}, F_2^{(j-1)}, \dots, D_{i-1}^{(j-1)}, F_{i-1}^{(j-1)}) \\ &\quad \cdot \prod_{i_1=1}^i P(F_{i_1}^{(j)} | D_1^{(1)}, D_2^{(1)}, F_2^{(1)}, \dots, D_M^{(1)}, F_M^{(1)}, \dots, D_1^{(j-1)}, \\ &\quad \dots, F_M^{(j-1)}, D_1^{(j)}, D_2^{(j)}, F_2^{(j)}, \dots, D_{i_1}^{(j)}) \cdot P(D_i^{(j)} | D_1^{(1)}, \\ &\quad D_2^{(1)}, F_2^{(1)}, \dots, D_M^{(1)}, F_M^{(1)}, \dots, D_1^{(j-1)}, \dots, F_M^{(j-1)}, \\ &\quad D_1^{(j)}, D_2^{(j)}, F_2^{(j)}, \dots, D_{i-1}^{(j)}, F_{i-1}^{(j)}). \end{aligned} \quad (3)$$

In (3), $P(D_i^{(j)} | D_1^{(1)}, D_2^{(1)}, F_2^{(1)}, \dots, F_M^{(1)}, \dots, D_1^{(j-1)}, \dots, F_M^{(j-1)}, D_1^{(j)}, D_2^{(j)}, F_2^{(j)}, \dots, D_{i-1}^{(j)}, F_{i-1}^{(j)})$ is the probability of failure in decoding the received sequence of s generated by γ_i at j th time, conditional on that all the previous transmission attempts of s failed. Denote such a conditional event as E . Denote the set of conditional events that at least one successful transmission attempt before using puncture pattern γ_i by j times as A . According to Bayes' Theorem, we have $P(D_i^{(j)}) = P(D_i^{(j)} | A)P(A) + P(D_i^{(j)} | E)P(E)$. However, with the CT-TCM HARQ, there is no successful transmission attempt before using puncture pattern γ_i at the j -th time, i.e., $P(A) = 0$. Thus, $P(D_i^{(j)}) = P(D_i^{(j)} | E)P(E)$. Due to the fact that $P(E) \leq 1$, we obtain $P(D_i^{(j)} | E) \geq P(D_i^{(j)})$. Similarly, we have $P(F_i^{(j)} | E) \geq P(F_i^{(j)})$ for the term of $P(F_i^{(j)} | D_1^{(1)}, D_2^{(1)}, F_2^{(1)}, \dots, D_M^{(1)}, F_M^{(1)}, \dots, D_1^{(j-1)}, \dots, F_M^{(j-1)}, D_1^{(j)}, D_2^{(j)}, F_2^{(j)}, \dots, D_i^{(j)})$ in (3). Then we can rewrite (3) as,

$$\begin{aligned} &P(D_1^{(1)}, \dots, F_M^{(1)}, \dots, D_1^{(j-1)}, \dots, F_M^{(j-1)}, D_1^{(j)}, \\ &\quad \dots, F_i^{(j)}) \geq \prod_{j=1}^{\infty} \prod_{i_1=1}^M P(D_{i_1}^{(j)}) \prod_{i_2=2}^M (F_{i_2}^{(j)}). \end{aligned} \quad (4)$$

Substituting (4) into (2), we obtain

$$\begin{aligned} \Lambda &\geq 1 + \sum_{j=1}^{\infty} \prod_{j'=1}^{j-1} \prod_{i_1=1}^M P(D_{i_1}^{(j')}) \prod_{i_2=2}^M P(F_{i_2}^{(j')}) \\ &\quad \cdot \sum_{i=1}^M \prod_{i_1=1}^i P(D_{i_1}^{(j)}) \prod_{i_2=2}^i P(F_{i_2}^{(j)}). \end{aligned} \quad (5)$$

Also, the following relationships hold:

$$P(D_1^{(1)}, D_2^{(1)}, F_2^{(1)}, \dots, F_M^{(1)}, \dots, D_1^{(j-1)}, \dots, F_M^{(j-1)}, D_1^{(j)}, D_2^{(j)}, F_2^{(j)}, \dots, D_i^{(j)}, F_i^{(j)}) \leq P(F_i^{(j)}), \quad (6)$$

$$P(D_1^{(1)}, D_2^{(1)}, F_2^{(1)}, \dots, F_M^{(1)}, \dots, D_1^{(j-1)}, \dots, F_M^{(j-1)}, D_1^{(j)}, \dots, D_M^{(j)}, F_M^{(j)}, D_1^{(j+1)}) \leq P(D_1^{(j+1)}). \quad (7)$$

Substituting (6) and (7) into (2), we obtain

$$\Lambda \leq 1 + \sum_{j=1}^{\infty} P(D_1^{(j)}) + \sum_{i=2}^M P(F_i^{(j)}). \quad (8)$$

Combining (5) and (8), the bounds of Λ are given as follows,

$$1 + \sum_{j=1}^{\infty} \prod_{j'=1}^{j-1} \prod_{i_1=1}^M P(D_{i_1}^{(j')}) \prod_{i_2=2}^M P(F_{i_2}^{(j')}) \cdot \sum_{i=1}^M \prod_{i_1=1}^i P(D_{i_1}^{(j)}) \prod_{i_2=2}^i P(F_{i_2}^{(j)}) \leq \Lambda \leq 1 + \sum_{j=1}^{\infty} P(D_1^{(j)}) + \sum_{i=2}^M P(F_i^{(j)}). \quad (9)$$

Denote n the number of information bits carried by each channel symbol. Denote η the link layer throughput efficiency, representing the average number of information-bits-per-symbol transmitted in the channel, $\eta = n/\Lambda$. In terms of (9), we obtain the bounds of the link layer throughput for our CT-TCM HARQ scheme as follows,

$$\frac{n}{1 + \sum_{j=1}^{\infty} P(D_1^{(j)}) + \sum_{i=2}^M P(F_i^{(j)})} \leq \eta \leq n \left/ \left[1 + \sum_{j=1}^{\infty} \prod_{j'=1}^{j-1} \prod_{i_1=1}^M P(D_{i_1}^{(j')}) \prod_{i_2=2}^M P(F_{i_2}^{(j')}) \cdot \sum_{i=1}^M \prod_{i_1=1}^i P(D_{i_1}^{(j)}) \prod_{i_2=2}^i P(F_{i_2}^{(j)}) \right] \right. \quad (10)$$

The values of $P(D_i^{(j)})$, $P(F_i^{(j)})$ are obtained by simulation of the decoding scheme in Fig. 2.

B. TCP Throughput

In this section, we develop an analytical model to evaluate wireless TCP throughput over an arbitrary hybrid stop-and-wait ARQ scheme. This model enables us to compare the impacts on TCP-level throughput due to different HARQ schemes in Section V.

We consider TCP connections traversing a wired line from a server to the access point (AP) and a wireless link from the AP to a wireless receiver. An arbitrary HARQ scheme is deployed in the wireless link. We assume that the traffic in the system is mainly due to the wireless user's downloading from the server. The related parameters are defined as follows.

D , d : round trip propagation delay per TCP packet in the wired and wireless link, respectively;

T , T_w : TCP packet service time over a HARQ scheme and queuing delay at the AP, respectively;

t_a : transmission delay of a link layer ACK or NAK packet over the wireless link;

B : TCP packet buffer size at the AP;

C : the wireless link capacity in bits/s;

L_f : the length of a wireless link layer frame;

q : TCP packet loss rate over the wireless link;

N : the number of link layer frames per TCP packet;

$\bar{\tau}$: average round trip time (RTT) of a TCP packet between the server and the receiver;

M_r : the maximum number of transmissions per link layer frame over the wireless link;

w_p : the maximum size of congestion window (*cwnd*) of a TCP connection.

We make the following reasonable assumptions to simplify our analysis. The wired path from the server to the AP is assumed error-free. The wireless link is assumed to be an AWGN channel, where the packet errors are independently and identically distributed (i.i.d.). With i.i.d. packet errors, the TCP sender is not likely to experience two or more consecutive packet losses. Hence, it is reasonable to assume that the probability of TCP timeouts due to consecutive packet losses is small enough to be negligible. It is also reasonable to assume that most of the traffic from the server to the receiver are aggressive large flows. Under these circumstances, we ignore the slow-start phase and only consider the congestion avoidance phase of TCP.

Each TCP packet from the server is queued in a buffer at the AP and fragmented into smaller link layer frames with fixed length L_f before they are sent. We ignore t_a since the link layer frames commonly have fairly smaller size than TCP packets. For the same reason, we also ignore any delay in transmission of the link layer ACK and NAK. As usual, $D \gg d$, then we assume $D + d \approx D$.

Assuming that TCP sessions arrive as a Poisson process, based on the results from [24] that the performance of statistical bandwidth sharing is insensitive to the flow size distribution and flow arrival process, we can model the arrival of aggregated TCP packets at the AP as a Poisson process with rate λ . This allows us to use the M/G/1 processor sharing model for TCP packets arriving at the AP buffer. It is known from [26] that for a first-come-first-service discipline, the M/G/1 processor sharing model has the same result as a M/G/1 queue [25]. Hence, the mean queueing delay T_w of a TCP packet at the AP can be derived from the M/G/1 queue by

$$\bar{T}_w = \lambda E[T^2] / [2(1 - \lambda \bar{T})], \quad (11)$$

where \bar{T} and $E[T^2]$ are the first and second moments of T . The average RTT ($\bar{\tau}$) is calculated by

$$\bar{\tau} = D + \bar{T} + \bar{T}_w = D + \bar{T} + \lambda E[T^2] / [2(1 - \lambda \bar{T})]. \quad (12)$$

A TCP packet is successfully received only when all of its link layer frames are correctly received. Consider a HARQ scheme with a maximum of M_r transmissions per link layer frame. If any frame of a TCP packet fails in M_r transmissions, this packet will be dropped and the next packet queueing in the buffer is to be sent. In order to obtain the first and second moments of T per TCP packet over the HARQ scheme in AWGN channels, we propose the following discrete-time

Markov chain. Denote (n, j) a state of the discrete-time Markov chain model, representing the n th frame of a TCP packet transmitted the j th time by a HARQ scheme. Denote T_f the transmission time of a link layer frame over the wireless link, $T_f = L_f/C$. The sojourn time at each state is a fixed time slot given by T_f . The discrete-time Markov chain is shown in Fig. 3, where $1 \leq n \leq N$, $1 \leq j \leq M_r$, g_j and e_j are the probabilities that a link layer frame succeeds and fails in the j th transmission, respectively, $g_j + e_j = 1$. The values of g_j and e_j depends on individual HARQ schemes. For these HARQ schemes which use the repetition coding in transmission without combining techniques in the AWGN channel, it is known that $g_{j+1} = g_j$ and $e_{j+1} = e_j$ for all $1 \leq j < M_r$. However, in the TCM codes-based HARQ schemes with combining techniques, different values of g_j and e_j are obtained in different retransmissions. The values of g_j and e_j in the AWGN channel can be obtained by simulation of the HARQ schemes.

Denote Ω the set of states of the discrete-time Markov chain model. The total number of states in Ω is NM_r . Denote $s = (n-1)M_r + j$ the index of state (n, j) , $1 \leq s \leq NM_r$. After a TCP packet finishes due to either failure in M_r transmissions (i.e., at $s \in \{s = nM_r : 1 \leq n \leq N\}$) or successfully received (i.e., at $s \in \{s = (N-1)M_r + j : 1 \leq j \leq M_r\}$), the state transits to $(1, 1)$ ($s = 1$) and the next TCP packet in buffer is transmitted. Denote $p_{ss'}$ the state transition probability from $s \in \Omega$ to $s' \in \Omega$. The values of $\{p_{ss'}\}$ are given by

$$p_{ss'} = \begin{cases} e_{M_r}, & s = nM_r, s' = 1 : 1 \leq n \leq N; \\ g_j, & s = (N-1)M_r + j, 1 \leq j \leq M_r, s' = 1; \\ g_j, & s = (n-1)M_r + j, s' = nM_r + 1, \\ & 1 \leq n \leq N-1, 1 \leq j \leq M_r; \\ e_s, & s = (n-1)M_r + j, s' = s + 1, \\ & 1 \leq n \leq N, 1 \leq j \leq M_r - 1; \\ 0, & s = s'; s, s' \in \Omega. \end{cases}$$

We obtain the transition probability matrix $\mathbf{P}_{NM_r \times NM_r} = [p_{ss'}]$. Denote π_s the probability of the discrete-time Markov chain being in state s . Define a row vector $\Pi = [\pi_s]$ the state probabilities of the discrete-time Markov chain. We now have a set of steady state equations $\pi_s = \sum_{s' \in \Omega} \pi_{s'} p_{s's}$ for all $s \in \Omega$ and the normalizing equation $\sum_{s \in \Omega} \pi_s = 1$. By solving these equations, we obtain the set of stationary state probabilities Π .

However, on calculating \bar{T} and $E[T^2]$, we are also interested in the transient state probabilities of the model. Define $\Pi^{(0)}$ as a row vector representing the initial probabilities for state $s \in \Omega$ with elements $\pi_s^{(0)} = 1$ for $s = 1$ and $\pi_s^{(0)} = 0$ for $s \in \Omega - \{1\}$. Denote Φ a NM_r by NM_r matrix, representing the state transition probabilities of our Markov chain, excluding the transitions to state $(1, 1)$ ($s = 1$). Denote $\phi_{ss'}$, $1 \leq s, s' \leq NM_r$, the element of Φ . The values of $\phi_{ss'}$ are given by

$$\phi_{ss'} = \begin{cases} p_{ss'}, & s \neq s', s' \neq 1; \\ 0, & \text{other } s, s' \in \Omega. \end{cases}$$

Define $X = [x_s]$ and $Y = [y_s]$ as two column vectors representing the transition probabilities from state $s \in \Omega$ to state $(1, 1)$ due to success and failure in a TCP packet transmission, respectively. From Fig. 3, we have $x_s = g_j$ for $\{s = (N-1)M_r + j : 1 \leq j \leq M_r\}$ and $x_s = 0$ for all other

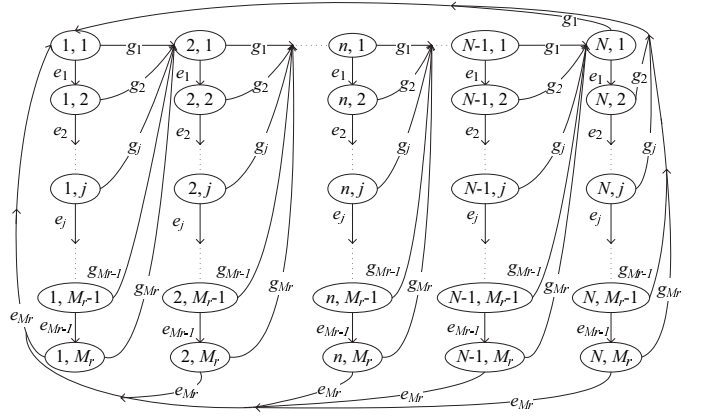


Fig. 3. Discrete-time Markov chain of TCP transmission over a HARQ scheme.

s ; $y_s = e_{M_r}$ for $\{s = nM_r : 1 \leq n \leq N\}$ and $y_s = 0$ for all other s . Denote l the number of state time slots experienced in one TCP packet transmission (including success and failure). Based on the property of Markov chains [25], the transient state probabilities for state $s \in \Omega$ at discrete time l is given by $\Pi^{(l)} = \Pi^{(0)} \Phi^{l-1} (X + Y)$. From Fig. 3, l is given from $\min(M_r, N)$ to NM_r . Thus, the first and second moments of service time per TCP packet over the wireless link with a HARQ scheme can be calculated by

$$\bar{T} = \sum_{l=\min(M_r, N)}^{NM_r} \Pi^{(0)} \Phi^{l-1} (X + Y) l T_f, \quad (13)$$

$$E[T^2] = \sum_{l=\min(M_r, N)}^{NM_r} \Pi^{(0)} \Phi^{l-1} (X + Y) (l T_f)^2. \quad (14)$$

Substituting (13)–(14) into (12), we obtain the average RTT $\bar{\tau}$ of a TCP packet through the wireless link deploying any HARQ scheme. The mean arrival rate λ of TCP packets is determined by the average number of TCP packets transmitted within the average RTT $\bar{\tau}$, which depends on the following two factors: buffer size at the AP and residual packet losses.

Firstly, if packet losses due to wireless channel errors are thoroughly combated by the link layer HARQ scheme, the size of $cwnd$ additively increases every RTT (since we assumed no congestion-related losses) till the buffer at the AP is filled. Then, the subsequently arrived TCP packets are dropped due to buffer overflow, and the size of $cwnd$ is halved (e.g. for TCP Reno [27]). Denote μ the TCP packet transmission rate over the wireless link, $\mu = 1/\bar{T}$. We have $w_p = (B + \mu\bar{\tau} + 1)$, determined by the buffer size at the AP. Denote \bar{K} the average number of TCP packets successfully received when $cwnd$ increases from $w_p/2$ to w_p , $\bar{K} = 3w_p^2/8$. λ is given by

$$\lambda = \bar{K} / \sum_{w_p/2}^{w_p} \bar{\tau} = (3w_p)/(4\bar{\tau}). \quad (15)$$

Secondly, if the packet losses due to wireless channel errors cannot be combated by M_r link layer transmissions of the HARQ scheme, the size of $cwnd$ is determined by the detected residual packet losses. Given the TCP packet loss rate q over the wireless link, the average number of TCP packets transmitted between two consecutively packet losses is $1/q$. Since the size of $cwnd$ halves due to packet losses, the average number of TCP packets transmitted between $w_p/2$

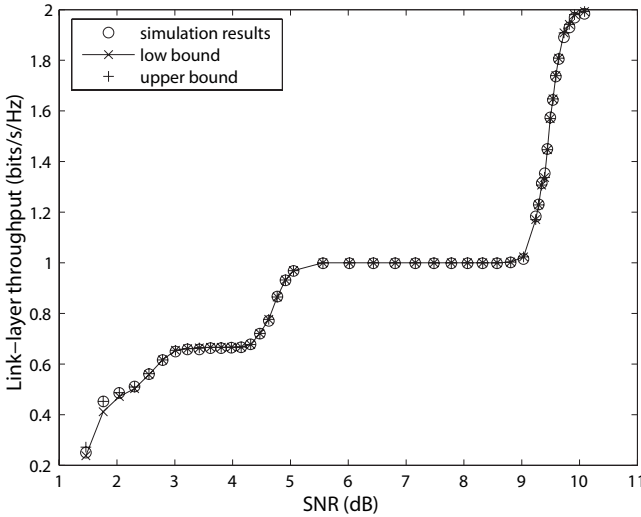


Fig. 4. Link layer throughput bounds.

and w_p can be approximated to be $1/q$, then we have $w_p = \sqrt{8/(3q)}$. For a TCP connection over an arbitrary HARQ scheme, the packet loss rate q corresponding to a maximum of M_r link layer transmissions can be obtained by $q = (\Pi Y)/[\Pi(X + Y)]$, based on Π obtained from the discrete-time Markov chain. Substituting q into $w_p = \sqrt{8/(3q)}$, we have $w_p = \sqrt{8\Pi(X + Y)/(3\Pi Y)}$.

Considering the above two factors, limited buffer size B and limited M_r transmissions per link layer frame, the mean TCP packet arrival rate λ is calculated from (12)–(15) as follows,

$$\lambda = \frac{3}{4\bar{T}} \min \left(\sqrt{\frac{8\Pi(X + Y)}{3\Pi Y}}, \left[B + \frac{D}{\bar{T}} + \frac{\lambda E[T^2]}{2\bar{T}(1 - \lambda\bar{T})} + 2 \right] \right). \quad (16)$$

How to solve for λ in (16) is a fixed point problem [28]. Rearranging (16) to be a form of $f(\lambda) = 0$, we see that $f(0) > 0$, $f(1/\bar{T}) < 0$ and $\frac{df}{d\lambda} < 0$ for $0 \leq \lambda \leq 1/\bar{T}$. It shows that there is a unique solution to (16). Thus, we can obtain the value of λ by fixed point iteration. Finally, the TCP level efficiency over the general HARQ scheme is obtained by $\lambda N L_f / C$.

V. NUMERICAL RESULTS

A. Link Layer Performance

We develop a simulation model in OPNET [29] to evaluate the performance of the proposed CT-TCM HARQ scheme with $M = 4$ and 8PSK modulation, referred to as CT-TCM-HARQ hereafter, over AWGN channels and compare with other HARQ schemes in [18]–[21]. The length per link layer frame of the CT-TCM-HARQ scheme is 2048 bits and each block is sent at most four times by the CT-TCM-HARQ. For simplicity, the ARQ feedback messages (ACKs and NAKs) are assumed to be always correctly received.

We plot in Fig. 4 the upper and lower bounds of the link layer throughput obtained by (10), as well as the throughput obtained by simulation. All three curves are indistinguishable when plotted. It demonstrates that the bounds are very tight

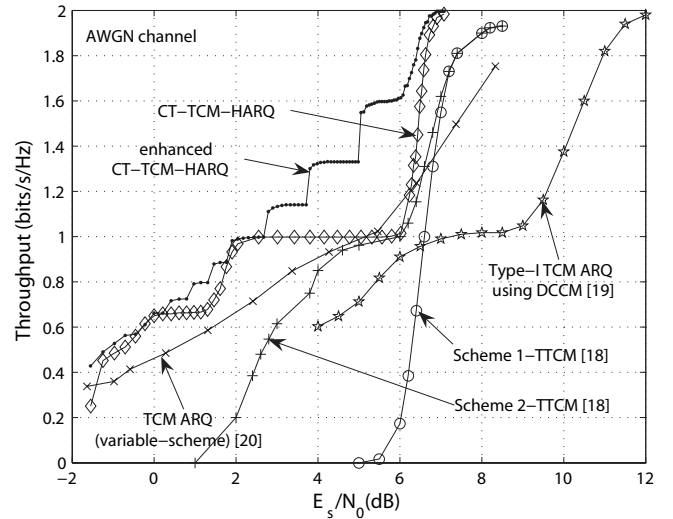


Fig. 5. Link layer throughput of the CT-TCM-HARQ and the TCM-based HARQ schemes of [18], [19], [20] over an AWGN channel.

and accurate. For further performance evaluation, we plot in Fig. 5 the analytical link layer throughput of the CT-TCM-HARQ, and compare the results with those of the schemes in [18]–[20]. In [18], three TTCM HARQ schemes were investigated. Here we only consider a comparison with Scheme 1-TTCM and Scheme 2-TTCM of [18], because the maximum spectrum efficiency of Scheme 3-TTCM in [18] is 3 bits/s/Hz while the CT-TCM-HARQ achieves a maximum of 2 bits/s/Hz. We observe that the throughput achieved by our CT-TCM-HARQ outperforms the other TCM-based HARQ schemes, except for a small E_s/N_0 range (5.2~6.2 dB) where our scheme is worse than [20]. It results from the discrete coding gains provided by the CT-TCM-HARQ in successive retransmissions, since we adopt the puncture patterns that satisfy constraints (i) and (ii). This problem can be eliminated by an enhanced CT-TCM-HARQ using smaller fragments of the punctured parts in retransmissions, as shown in Fig. 5.

Fig. 6 provides the bit error rate (BER) as a function of E_s/N_0 obtained by our scheme and the schemes of [18], [20]. The BER of the TTCM HARQ of [18] is cited from the same authors' work in [30] for the case of CRC-24. The results demonstrate the following two aspects. First, we compare a Type-I ARQ which simply repeats the same CT-TCM code in each retransmission with Scheme 1-TTCM of [18]. Both of the schemes use the repetition coding in retransmission. We observe that significantly improved coding gain is achieved by the Type-I ARQ using the CT-TCM code. This demonstrates that the CT-TCM code has better performance than the TTCM code of [13]. Second, we observe that further coding gain is achieved by the proposed CT-TCM-HARQ because different punctured parts are used in subsequent retransmissions, comparing with the repetition coding used in the Type-I ARQ with the CT-TCM code and the scheme of [20] which switches between two TCM codes after two transmissions. Fig. 7 presents the frame error rate (FER) against E_b/N_0 obtained by the CT-TCM-HARQ and the ARQ with a rate 1/2 Turbo code [21]. It demonstrates again that significantly improved coding gain is achieved by the CT-TCM-HARQ than the Turbo ARQ

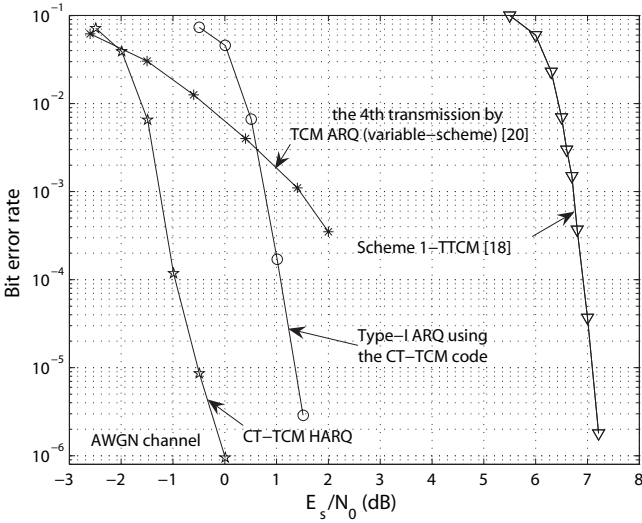


Fig. 6. BER of our scheme and the TCM-based HARQ schemes of [18] and [20] over an AWGN channel.

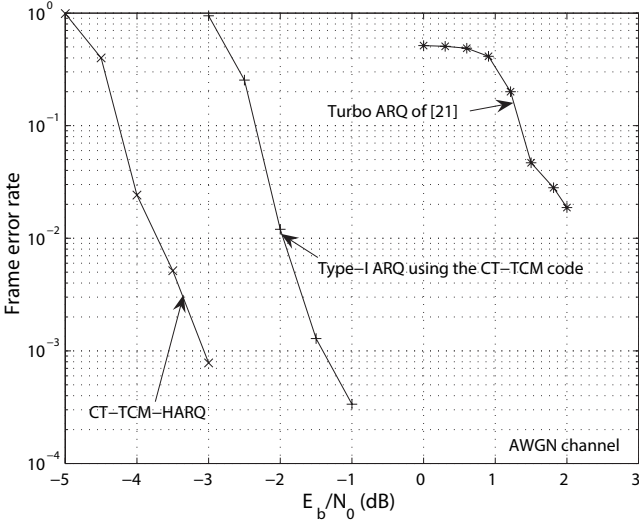


Fig. 7. FER of our scheme and the Turbo ARQ [21] with a rate 1/2 Turbo code and 1024-bit packet over an AWGN channel.

scheme of [21], even in the case that we use the simple Type-I ARQ with the CT-TCM code.

B. TCP Performance

We evaluate TCP performance over the CT-TCM-HARQ under the following parameters. The capacity of the wireless link is set to 2.048 M bits/s. Each TCP packet has 8192 bits. Let $D = 0.1$ s, $B = 20$, $M_r = 4$, unless they are specified.

We compare the TCP level performances between our proposed CT-TCM-HARQ and the TCM-based HARQ schemes reviewed in Section II. Since the previous comparisons of the link layer performance have demonstrated that the scheme in [20] outperforms the other schemes, here we present a comparison of the TCP level performance between our scheme and the scheme of [20]. Based on the symbol error rate (SER) results provided in [20], we can estimate the FER of the scheme of [20] by $FER=1 - (1 - SER)^{n_s}$, where n_s denotes the number of symbols per frame. Such estimation is

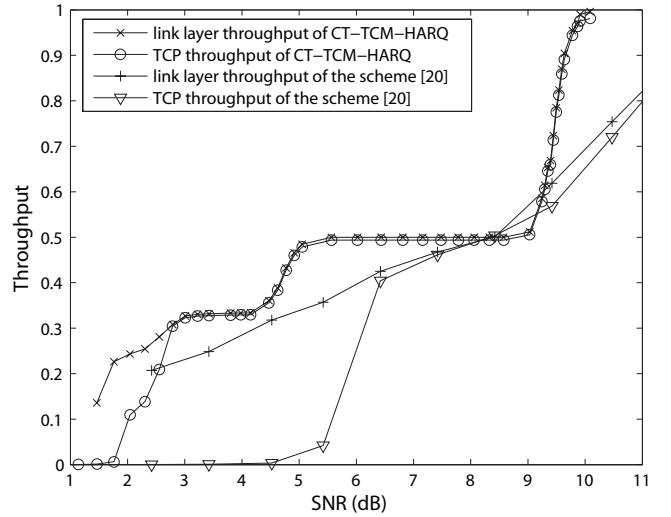


Fig. 8. TCP Throughput comparisons between our scheme and the scheme of [20].

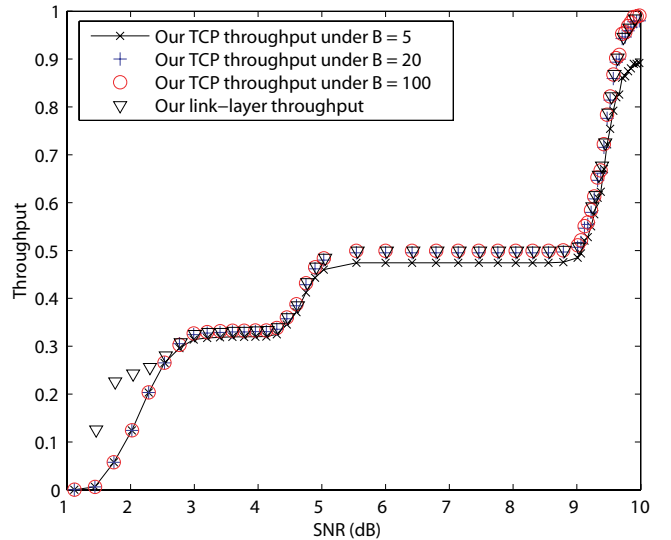


Fig. 9. TCP throughput under different buffer size B .

made under the assumption that the correlation in the SER process is negligible. This assumption is commonly justified for convolutional codes such as the one used in [20]. From the estimated FER results, we can derive the TCP throughput obtained by the scheme of [20]. The analytical results of TCP throughput obtained by the CT-TCM-HARQ and the scheme of [20] are plotted in Fig. 8. It demonstrates that significantly improved TCP throughput is achieved by our scheme. Realizing that the link layer throughput provides the capacity limit of TCP over the wireless link, we also include in Fig. 8 the link layer throughput of both schemes for a further comparison. We observe that the difference between the TCP throughput of the scheme of [20] and its link layer counterpart is larger than that of our scheme. The large difference in the scheme of [20] is mainly attributed to its higher link layer losses than that of our scheme. As a result, TCP over the scheme of [20] requires larger buffer size than $B = 20$ at the AP for more local retransmissions of the corrupted packets.

In Fig. 9, we plot the TCP throughput over the CT-TCM-

TABLE II
COMPARISONS OF TCP THROUGHPUT OBTAINED BY ANALYSIS AND SIMULATION.

Throughput v.s. SNR (dB)	1	2	3	4	5	6	7	8	9	10
Analytical results (B = 5)	0.5×10^{-5}	0.12	0.314	0.32	0.46	0.475	0.475	0.475	0.485	0.905
Simulation results (B = 5)	1.12×10^{-4}	0.09	0.318	0.324	0.462	0.48	0.48	0.48	0.49	0.92
Analytical results (B = 20)	0.5×10^{-5}	0.124	0.324	0.330	0.479	0.495	0.495	0.485	0.507	0.982
Simulation results (B = 20)	1.15×10^{-4}	0.109	0.323	0.329	0.478	0.494	0.494	0.494	0.505	0.979

HARQ under different buffer size B . The results demonstrate that high TCP throughput is achieved under large buffer size. However, we also observe that as the buffer size further increases from 20 packets to 100 packets, no significant improvement on TCP throughput is achieved. The reason for this effect is that the increase of B incurs the increase of queuing delay to a newly arrived TCP packet in the buffer and finally leads to the increase of the end-to-end RTT of a TCP packet. As a result, the TCP throughput cannot be further improved due to large RTT. Instead, it reaches the maximum link capacity (the link layer throughput is also included in Fig. 9, that provides fundamental limit to TCP throughput. The above discussion is based on the assumption that a TCP source can accurately estimate its RTT and uses the results to update its timeout threshold. Otherwise, large buffer size leads to undesirable longer packet delay or even timeouts [31], which degrade the end-to-end TCP throughput. These results provide arguments against the usage of large buffers as the small benefit obtained in TCP throughput may not justify the longer RTT. We also observe in Fig. 9 that increasing the AP buffer size cannot improve the TCP throughput when $\text{SNR} < 2.5$ dB. This is because very high packet loss rate when $\text{SNR} < 2.5$ dB, $q > 0.1855$, and the maximum size of $cwnd$ is constrained by $\sqrt{8/(3q)}$.

Finally, we verify our analytical results by simulation in OPNET. We run a FTP download application provided by the OPNET package over the TCP Reno through the wireless link. The inter-request time of FTP is exponentially distributed with mean 0.01 second, and the file size of FTP is 4000 bytes. Table II presents the comparisons between the analytical and simulated results for $B = 5$ and 20 TCP packets. In analysis, the values of g_j and e_j fed into the discrete-time Markov chain model in Fig. 3 are obtained by the simulation of the CT-TCM decoding algorithm [16]. The comparisons demonstrate strong agreements between the analysis and the simulation.

VI. CONCLUSION

The applications of TCM codes in HARQ schemes and their impacts on TCP over wireless channels have been investigated in this paper. A Markov model has been developed to describe the interaction between TCP and the underlying HARQ scheme in a wireless TCP system. By this model, we have obtained the analytical TCP throughput over the proposed CT-TCM HARQ and investigated the impacts on TCP due to underlying FEC schemes, link layer retransmission, maximum attempts, and packet buffer size at AP. Both analysis and simulations show that the CT-TCM HARQ provides faster increase of coding gain in retransmissions and higher link

layer and TCP layer throughput than other existing TCM-based ARQs.

REFERENCES

- [1] Y. Tian, K. Xu, and N. Ansari, "TCP in wireless environments: problems and solutions," *IEEE Commun. Mag.*, vol. 43, no. 3, pp. S27–S32, Mar. 2005.
- [2] H. Balakrishnan *et al.*, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Trans. Networking*, vol. 5, no. 6, pp. 756–769, Dec. 1997.
- [3] A. Bakre and B. R. Badrinath, "I-TCP: indirect TCP for mobile hosts," in *Proc. 15th International Conference on Distributed Computing Systems*, pp. 136–143, May 1995.
- [4] H. Balakrishnan *et al.*, "Improving reliable transport and handoff performance in cellular wireless networks," *Wireless Networks*, vol. 1, no. 4, pp. 469–481, Dec. 1995.
- [5] T. Goff *et al.*, "Freeze-TCP: a true end-to-end enhancement mechanism for mobile environments," in *Proc. IEEE INFOCOM'2000*, Tel Aviv, Mar. 2000, pp. 1537–1545.
- [6] K. Ramakrishnan and S. Floyd, "The addition of explicit congestion notification (ECN) to IP," RFC 3168, Sept. 2001.
- [7] K. Xu, Y. Tian, and N. Ansari, "TCP-Jersey for wireless IP communications," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 4, pp. 747–756, May 2004.
- [8] C. P. Fu and S. C. Liew, "TCP Venet: TCP enhancement for transmission over wireless access networks," *IEEE J. Sel. Areas Commun.*, vol. 21, pp. 216–228, Feb. 2003.
- [9] R. G. Mukhtar, S. Hanly, M. Zukerman, and F. Cameron, "A model for the performance evaluation of packet transmissions using Type-II hybrid ARQ over a correlated error channel," *Wireless Networks*, vol. 10, no. 1, pp. 7–16, Jan. 2004.
- [10] A. Chockalingam, M. Zorzi, and V. Tralli, "Wireless TCP performance with link layer FEC/ARQ," in *Proc. IEEE ICC'99*, Vancouver, BC, June 1999, pp. 1212–1216.
- [11] C. F. Chiasserini and M. Meo, "A reconfigurable protocol setting to improve TCP over wireless," *IEEE Trans. Veh. Technol.*, vol. 51, no. 6, pp. 1608–1620, Nov. 2002.
- [12] A. Ghosh *et al.*, "Broadband wireless access with WiMax/802.16: current performance benchmarks and future potential," *IEEE Commun. Mag.*, vol. 43, no. 2, pp. 129–136, Feb. 2005.
- [13] S. L. Goff, A. Glavieux, and C. Berrou, "Turbo-codes and high spectral efficiency modulation," in *Proc. IEEE ICC'94*, New Orleans, LA, May 1994, pp. 645–649.
- [14] P. Robertson and T. Woz, "Bandwidth-efficient turbo trellis-coded modulation using punctured component codes," *IEEE J. Sel. Areas Commun.*, vol. 16, pp. 206–218, Feb. 1998.
- [15] C. Fragouli and R. D. Wesel, "Turbo-encoder design for symbol-interleaved parallel concatenated trellis-coded modulation," *IEEE Trans. Commun.*, vol. 49, pp. 425–435, Mar. 2001.
- [16] P. Li *et al.*, "Low complexity concatenated two-state TCM schemes with near capacity performance," *IEEE Trans. Inf. Theory*, vol. 49, no. 12, pp. 3225–3234, Dec. 2003.
- [17] R. H. Deng, "Hybrid ARQ scheme using TCM and code combining," *Electron. Lett.*, vol. 27, no. 10, pp. 866–868, 1991.
- [18] A. Banerjee, D. J. Costello Jr., and T. E. Fuja, "Bandwidth efficient hybrid ARQ schemes using Turbo codes," in *Proc. ISIT 2000*, Sorrento, June 2000, pp. 188.
- [19] M. P. Schmitt, "Hybrid ARQ scheme employing TCM and packet combining," *Electron. Lett.*, vol. 34, no. 18, pp. 1725–1726, 1998.
- [20] J. X. Yu *et al.*, "Hybrid-ARQ scheme using different TCM for retransmission," *IEEE Trans. Commun.*, vol. 48, no. 10, pp. 1609–1613, Oct. 2000.

- [21] K. R. Narayanan and G. L. Stuber, "A novel ARQ technique using the turbo coding principle," *IEEE Commun. Lett.*, vol. 1, no. 2, pp. 49–51, Mar. 1997.
- [22] P. Li and K. Y. Wu, "Concatenated tree codes: a low complexity, high performance approach," *IEEE Trans. Inf. Theory*, vol. 47, pp. 791–799, Feb. 2001.
- [23] Q. Huang, S. Chan, P. Li, and M. Zukerman, "A hybrid ARQ scheme based on CT-TCM codes," *IEEE Commun. Lett.*, vol. 9, no. 7, pp. 664–666, July 2005.
- [24] T. Donald *et al.*, "Insensitivity results in statistical bandwidth sharing" in *Proc. 17th International Teletraffic Congress*, Salvador da Bahia, Brazil, Sept. 2001, pp. 125–136.
- [25] L. Kleinrock, *Queueing Systems, Volume I: Theory*. New York: Wiley, 1976.
- [26] L. Kleinrock, *Queueing Systems, Volume II: Computer Applications*. New York: Wiley, 1976.
- [27] W. R. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*. Reading, MA: Addison-Wesley, 1994.
- [28] H. L. Vu and S. Hanly, "A study of TCP performance and buffer occupancy over a fading wireless link," in *Proc. IEEE GLOBECOM'01*, San Antonio, TX, Nov. 2001, pp. 3478–3482.
- [29] OPNET University Program: <http://www.opnet.com/services/university/>
- [30] A. Banerjee, D. J. Costello Jr., and T. E. Fuja, "Comparison of different retransmission strategies for bandwidth efficient hybrid ARQ schemes using Turbo codes," in *Proc. ICPWC*, Hyderabad, India, Dec. 2000, pp. 548–552.
- [31] G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing router buffers," in *Proc. SIGCOMM*, Portland, Oregon, USA, Aug. 2004, pp. 281–292.



City University of Hong

Sammy Chan (M'87) received his B.E. and M.Eng.Sc. degrees in electrical engineering from the University of Melbourne, Australia, in 1988 and 1990, respectively, and a Ph.D. degree in communication engineering from the Royal Melbourne Institute of Technology, Australia, in 1995. From 1989 to 1994, he was with Telecom Australia Research Laboratories, first as a research engineer, and between 1992 and 1994 as a senior research engineer and project leader. Since December 1994, he has been with the Department of Electronic Engineering, City University of Hong Kong, where he is currently an associate professor.



Telecom - Royal Society

Li Ping (S'87-M'91-SM'05) received his Ph.D. degree at Glasgow University in 1990. He lectured at Department of Electronic Engineering, Melbourne University, from 1990 to 1992, and worked as a research staff at Telecom Australia Research Laboratories from 1993 to 1995. He has been with the Department of Electronic Engineering, City University of Hong Kong, since January 1996 where he is now a chair professor. His research interests are mixed analog/digital circuits, communications systems and coding theory. Dr. Li Ping was awarded a British Telecom - Royal Society Fellowship in 1986, the IEE J J Thomson premium in 1993 and a Croucher Senior Research Fellowship in 2005.



Moshe Zukerman (M'87-SM'91-F'07) received his B.Sc. in Industrial Engineering and Management and his M.Sc. in Operation Research from Technion - Israel Institute of Technology and a Ph.D. degree in Electrical Engineering from The University of California Los Angeles in 1985. He was an independent consultant with IRI Corporation and a post-doctoral fellow at UCLA during 1985-1986. During 1986-1997 he served in Telstra Research Laboratories. In 1997 he joined The University of Melbourne where he is responsible for promoting and expanding telecommunications research and teaching in the Electrical and Electronic Engineering Department. He served on the editorial board of the *Australian Telecommunications Research Journal*, *Computer Networks*, and *IEEE Communications Magazine*. He also served as a Guest Editor of *IEEE Journal on Selected Areas in Communications* for two issues: Future Voice Technologies and Analysis and Synthesis of MAC Protocols. Presently, he is serving on the editorial board of the *IEEE/ACM Transactions on Networking*, the *International Journal of Communication Systems* and *IEEE Journal on Selected Areas in Communications*. Dr. Zukerman has over 250 publications in scientific journals and conference proceedings.



Qian Huang received her Ph.D. degree in communication engineering from Shanghai University in 2002. She worked as a design engineer in Lucent Technologies (China) Co. Ltd., Shanghai, China, between August and October in 2002. In 2003, she worked as a research associate in Department of Electronic Engineering, City University of Hong Kong. From November 2003 to February 2004, she was a visiting research fellow with the Department of Electrical and Electronic Engineering, The University of Melbourne. She has been with the

Department of Electronic Engineering, City University of Hong Kong, since March 2004, where she is now a research fellow. Her research interests are in wireless communication, wireless TCP and tele-traffic modeling in overlay network.