Jurnal Teknologi, 54(Sains & Kejuruteraan) Jan. 2011: 45–54

HYBRID ARQ TYPE I BASED ON CONVOLUTIONAL CODE

HAZILAH MAD KAIDI[1] & MUHAMMAD IBRAHIM[2]

Abstract. A Hybrid Automatic Repeat reQuest (HARQ), error control scheme based on a convolutional code for packet transmission over wireless channels was proposed. The analysis of the throughput and bit-error rate (BER) performance, according to the different constraint lengths (K=3 and K=4) and code rate (1/2 and 1/3) of convolutional codes on HARQ type I simulation scheme are presented. Certain error correction capability is provided in each (re)transmitted packet, and the information can be recovered from each transmission or retransmission alone if the errors are within the error correction capability. Simulation of HARQ is limited up to three retransmissions for each SNR in several iterations.

Keywords: *Terms*–Hybrid ARQ; ARQ; convolutional coding; error control

Abstrak. Hibrid Automatik Permintaan Ulangan (HARQ), sejenis kaedah mengawal kesilapan berdasarkan pada kod konvolutional di penghantaran paket merentasi saluran tanpa wayar telah dikemukakan. Analisis prestasi daya pemprosesan dan kadar-bit-silap (BER), merujuk kepada simulasi panjang kekangan yang berbeza (K=3 dan K=4) dan kadar kod (1/2 dan 1/3) di kod konvolutional pada HARQ jenis I telah dipersembahkan. Beberapa kebolehupaya kesilapan pembetulan disediakan pada setiap penghantaran semula paket dan maklumat yang boleh dibaiki semula dengan sendiri dari setiap penghantaran atau penghantaran semula jikalau kesilapan adalah berada di antara julat kebolehupaya pembetulan kesilapan. Simulasi HARQ adalah terhad kepada tiga penghantaran semula bagi setiap satu SNR dalam beberapa kali perulangan.

Kata kunci: Istilah– Hibrid ARQ; ARQ; pengkodan konvolusional; kawalan kesilapan

1.0 INTRODUCTION

Error-control codes are now used in almost the entire range of information communication, storage and processing systems. Rapid advances in electronic and optical devices and systems have enabled the implementation of very powerful codes with close to optimum error-control performance [1]. A major objective when designing an error control scheme for a particular application is to keep redundancy as low as possible and present throughput as high as possible [2]. Therefore, an error control scheme such as HARQ was created to enhance the performance.

Scope of this paper is to investigate and analyze the throughput and bit error rate error (BER) in HARQ Type I scheme based on Convolutional code with different

[1] UTM RAZAK School of Engineering and Advanced Technology, Universiti Teknologi Malaysia, Kuala Lumpur
[2] Faculty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam

constraint length and code rate. This scheme suggests that where possible an integration of the two techniques (ARQ and FEC) schemes might operate more efficient than either on FEC own. The FEC is used to correct small numbers of errors, while ARQ will send acknowledgment to retransmit the previous data. An advantage of the scheme is that the same packet has to be transmitted only if a packet has been actually lost or detected being in error.

This paper is organised as follows. In Section 2, the general description of the proposed FEC and HARQ scheme are presented and some of its characteristics are discussed in order to improve the FEC method. Section 3 presents numerical results and comparisons with simulations made in Matlab environment, whereby the symbol of throughput, BER and number of retransmissions are mainly considered. Section 4 concludes the paper.

2.0 SYSTEM MODELS

2.1 Convolutional Coding

Communication systems that use the FEC approach are not able to request a repetition of the transmission of coded information. Due to this, all the capability of the code is used for error correction. Convolutional code is an example of FEC beside Red Solomon and Turbo codes. At the receiver side, the redundant information can be used to detect and reconstruct a certain amount of data corruption [1].

Convolutional codes are usually described using two parameters: the code rate and the constraint length. The code rate, k/n, is expressed as a ratio of the number of bits into the convolutional encoder (k) to the number of channel symbols output by the convolutional encoder (n) in a given encoder cycle. The constraint length parameter, K, denotes the "length" of the convolutional encoder, i.e. how many k-bit stages are available to feed the combinatorial logic that produces the output symbols [3]. The code of a convolutional encoder is generated by combining certain sequences of the current bit and stored bits. The outputs are generated by an XOR or addition of a certain sequence of bits, which is show in Figure 1.

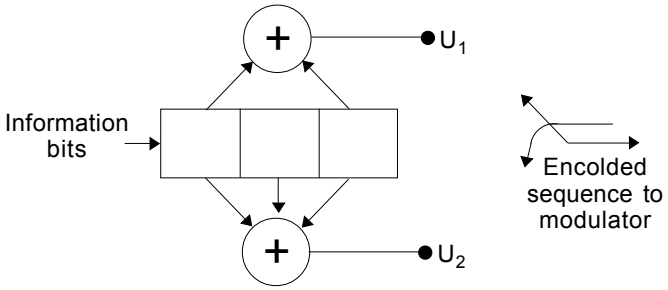

Figure 1 Rate 1/2 convolutional encoder

In specific example, considered the binary convolutional encoder with constraint length K = 3, k = 1, n = 2, which is shown in Figure 1. Suppose the information bits sequence is 101100111_2. Initially, the shift register is assumed to be all-zero state. The generator for first output is g_1=[101] which is representing octal value of 5. The second generator is connected to stage 1, 2 and 3. Hence g_2=[111] represent 7 in octal. The generators for this code are more conveniently given in octal form as (5, 7). All generators are the impulse responses from the encoder input to the two outputs. The output sequence will be 00 11 01 00 10 10 11 11 10 01_2 after all of the inputs have been presented to the encoder.

The convolutional code of the encoder diagram can also be represented by a state transition diagram. It is illustrated in Figure 2. Referring to example of Figure 1, this encoder may be regarded as a finite state machine. Let consider the input as previous, 101100111_2, and assumed the initial value of register contains are 0. The output sequence is 00 11 01 00 10 10 11 11 10 01.

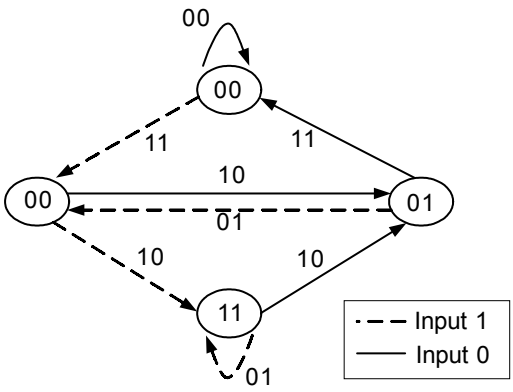

Figure 2 State transition diagram representation for rate 1/2 convolutional encoder

The encoder begins in state 00. If the first encoder bit is a 0, the encoder exits state 00 on the solid branch, output the two code symbols found on this branch (00), and return to state 00. If the first input bit were 1, the encoder would exit state 00 on the dashed branch, output the two code symbols found on this branch (11), and would enter state 10. In general, encoder inputs equal to 0 caused the encoder to move along the solid branches to the next state, again outputting two codeword symbols encountered along the branch. Convolutional coders can therefore be thought of as finite-state machines that changes states as a function of the input sequence [4]. The branch labels in this state transition diagram are calculated directly from the shift register representation.

Table 1 Simulation parameters

Code Rate	Contraint Length (K)	Generators (Octal)			d_{free}	No. of error corrected
		g_1	g_2	g_3		
1/2	3	5	7		5	2
	4	15	17		6	2
1/3	3	5	7	7	8	3
	4	13	15	17	10	4

For determine the number of errors that FEC can correct, some calculation is used at the decoder. The decoder does MLSD using the Viterbi algorithm. Equal gain combining is applied and a soft bounded distance decoder with distance $t = (d_{free}-1)/2$ is used to determine the need for a retransmission [5]. The number of errors can be corrected is tabulated in Table 1 regarding to the characteristic of convolutional code.

2.2 Hybrid ARQ Scheme

Hybrid ARQ schemes can be classified into three categories: Type-I, type-II and type III hybrid schemes. Stop-and-wait (SW), go-back-N (GBN) and selective-repeat (SR) modes of basic ARQ schemes can be applied to that types of hybrid ARQ schemes. The stop-and-wait (SW) ARQ scheme is the simplest ARQ scheme, and was implemented in early error-control systems. The SW asn HARQ schemes are the simplest scheme compared to another schemes [6].

Hybrid ARQ type I is known as packet combining technique is typically based on a block code that is designed partly to correct a small number of errors and partly to detect a large number of errors. It uses error detection and correction codes for each transmission and retransmission. Previously received packets with incorrect errors are discarded [6]. Certain error correction capability is provided in each (re)transmitted packet, and the information can be recovered from each transmission or retransmission alone if the errors are within the error correction capability [7].

In a type-II scheme the first transmission usually includes information bits and a limited amount of redundant bits. If a retransmission is needed, additional redundant bits are sent which are combined with the previously received redundant bits. [8]. However, HARQ Type III, each retransmission is self-decoded able. Chase combining (also called HARQ Type III with one redundancy version) involves the retransmission by the transmitter of the same coded data packet. The decoder at the receiver combines these multiple copies of the transmitted packet weighted by the signal-to-noise ratio (SNR) of the received signal for each attempt [9].

Throughput and BER equations in this simulation are applied as below [9];

$$Throughput = \frac{\sum Accepted\ -\ errorfree\ -\ message\ -\ bits}{\sum Transmitted\ -\ coded\ -\ bits} \quad (1)$$

$$BER = \frac{\sum No\ -\ of\ -\ bit\ -\ errors}{\sum Transmitted\ -\ bits} \quad (2)$$

Figure 3 shows the process involved in the simulation of the Hybrid ARQ convolutional coding system. The variables such as number of bits to be transmitted, the generator polynomial and the range of E_b/N_0 need to be initialized. The data generated are encoded by convolutional encoder after creating the check bits at error detection such as CRC generator. The message is modulated by a modulator before it is transmitted through the Additive White Gaussian Noise (AWGN) channel. The AWGN noise is generated randomly and added to the message to simulate the interference that co-exists in transmission medium. AWGN is chosen in this study because it produces simple and tractable mathematical models which are useful for gaining insight into the underlying behavior of a system before other phenomena are considered.

At the receiver, the signal will be demodulated and decoded by Viterbi decoder in order to recover the original message. The bit error rate (BER) is calculated by comparing the decoded data with the original data. Then, CRC detector is used for detecting errors. At the receiving end, the additional bits are used to check if the message needs retransmission. If the receiver detects that there are errors in the received bit stream, the receiver asks the sender to retransmit the data. Negative acknowledgement (NACK) sends to the transmitter to request for retransmission of the previous data. The positive acknowledgment (ACK) sends where there are no errors being detected.

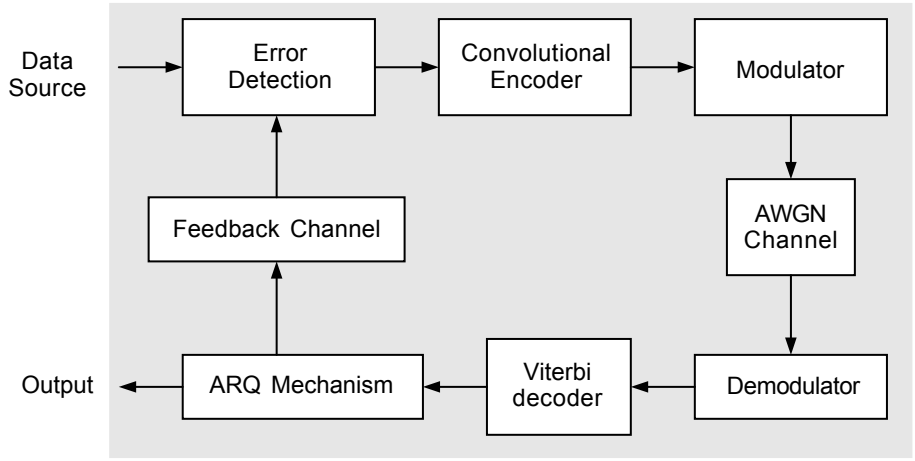

Figure 3 Simulated hybrid ARQ on convolutional coding transceiver block diagram

3.0 SIMULATION AND RESULTS

The HARQ Type I concept is implemented when the retransmission occurred. The retransmission repeated at the CRC generator and the process above is continued. Retransmission is limited for three times in this simulation. Retransmitting data is discarded when the iteration system detects more than three retransmissions. Then, the new information data is generated. This process continues until the iteration reaches ten times of simulations. In error correction technique, redundant bits are use to calculate syndrome in order to check for errors. If syndrome is equal to zeros value, there is no errors is detected in the system. If the syndrome equals to non-zeros value, retransmission of the same data is required.

Table 2 Number of erasures in transmitted and received sequence of symbols

Transmitter	Receiver
• Send information bits	■ Check local correctness and/or sequence of number received symbols and send ACK/NACK to the transmitter
• Compute parity symbols	■ Check whether an information sequence has to be declared as finally error free or still in error
• Count retransmissions from receiver (not greater than 3)	■ Calculate syndromes for an information sequence and correct erasures
• Clear buffer with parity symbols and retransmission flag and start to retransmit the sequence of symbols again to encoder	■ As long as the number of parity symbols is within 0-3 it operates like previously
• Transmit a new information sequence if no retransmission detected (no error)	■ If there were NACKs generated than three, discard all received symbols and clear the parity buffer

The simulation repeats until all the constraint length values (K = 3 and K = 4) have been tested. After all processes have been completed, all the output data (BER) and throughput produced are collected and plotted onto different graphs. These processes are applied on both code rate of 1/2 and code rate 1/3. All the results for BER and throughput performance are shown in Figure 4, 5, 6, 7, Table 3, Table 4 and Table 5.

Referring to the results in Figure 4 and Figure 5, code rate 1/3 gives the lower bit-error rate compared to rate 1/2. On the other hand, constraint length K = 4 presents lower bit-error rate instead of K = 3. The improvement of K=4 is most noticeable around E_b/N_o=3.5 dB for rate ½ in Figure 4. In Figure 5, it shows that K=4 is almost at lower E_b/N_o from 0 dB until up to 6 dB.

The corresponding throughput is depicted in Figure 6 and Figure 7. The equivalent results perform whereas throughput for K = 3 and rate ½ is keeps gradually higher until 3.5 dB only. Starting 3.5 dB onward, K = 4 achieve higher throughput. Nevertheless,

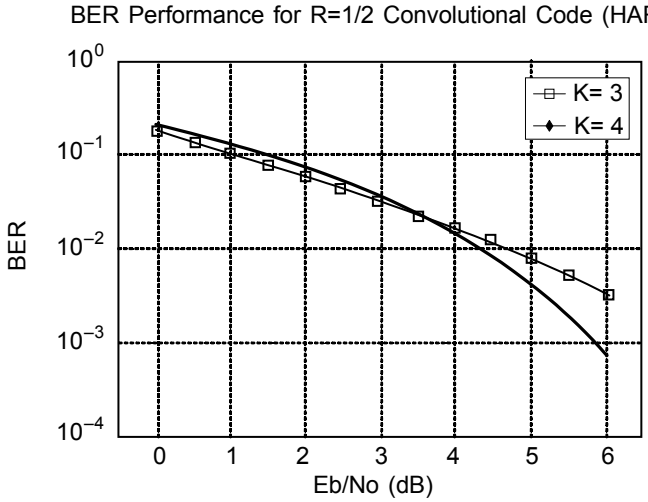

Figure 4 Bit error rate for coderate 1/2, constraint length 3 and 4

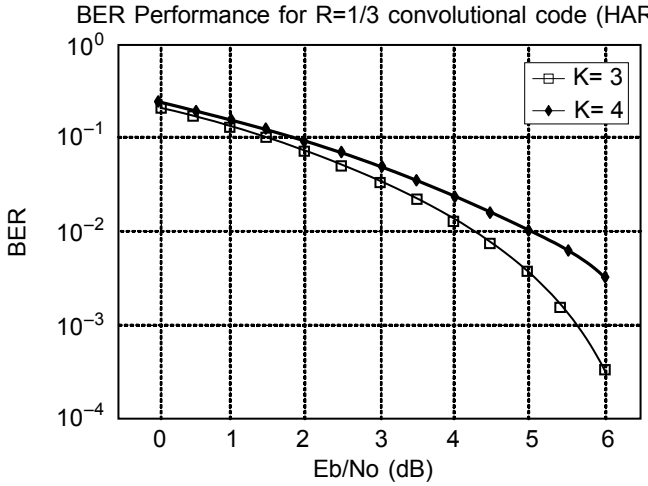

Figure 5 Bit error rate for coderate 1/3, constraint length 3 and 4

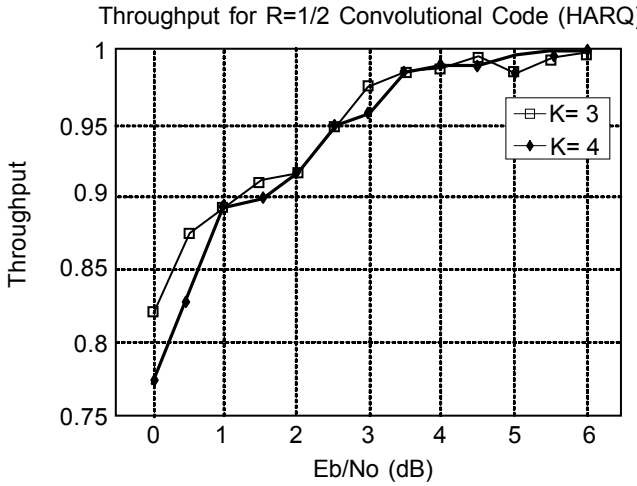

Figure 6 Throughput for rate 1/2, constraint length 3 and 4

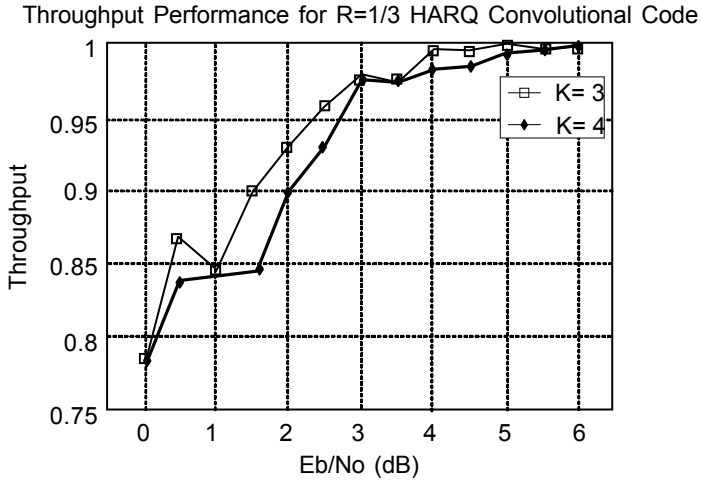

Figure 7 Throughput for rate 1/3, constraint length 3 and 4

Table 3 BER for coderate 1/2

E_b/N_o (dB)	K = 3			K = 4		
	BER	No. of Errors	No. of feedback	BER	No. of Errors	No. of feedback
0	0.1793	36	3	0.2258	41	3
2.0	0.0853	11	2	0.0869	15	2
4.0	0.0148	2	1	0.0134	4	1
6.0	0.0023	2	1	0.0007	2	1

Table 4 BER for coderate 1/3

E_b/N_o (dB)	K = 3			K = 4		
	BER	No. of Errors	No. of retransmission	BER	No. of Errors	No. of retransmission
0	0.2185	46	4	0.2156	38	3
2.0	0.1057	19	2	0.0736	14	2
4.0	0.0203	4	1	0.0091	3	1
6.0	0.0031	2	1	0.0004	2	1

Table 5 The lowest BER achieved by FEC and HARQ

Code Rate	FEC		HARQ	
	K=3	K=4	K=3	K=4
R = 1/2	0.0200	0.0500	0.0010	0.0004
R = 1/3	0.0900	0.0065	0.0008	0.0002

in Figure 7 shows that K = 4 slightly increase almost higher than K = 3 for code rate 1/3. Thus, can be concluded that as expected there are improvements to be gained by saving the inner extrinsic information.

The results of bit-error rate, number of bit errors and number of retransmission for average of 10 iterations are tabulated in TABLE 3 and TABLE 4. These tables show the effectiveness of retransmission in HARQ to correct errors in different code rate. In TABLE 5, the lowest bit error rate is tabulated by HARQ compared to FEC scheme. The FEC results are referred from [10] simulation which is analyzed the BER performance for FEC mechanism only. Indeed, when comparing both code rates (rate ½ and 1/3), it can be seen that good performance in HARQ mechanism which has achieved the lowest bit-error rate compared to FEC in both different code rates.

4.0 CONCLUSION

In this paper, hybrid ARQ type I schemes has analyzed with stop and wait ARQ scheme. With retransmission provided by the hybrid ARQ schemes, data transmission over channels has been investigated. Simulation results for a data transmission system have shown good improvement as a function of throughput and BER compared to FEC. Hybrid ARQ scheme is quite attractive and can be used in any data communication system.

REFERENCES

[1] Jorge Castineira Moreira, Patrick Guy Farrell. 2006. *Essential of Error-Control Coding*. John Wiley & Sons Ltd.

[2] Igor Grellneth, Marian Oravik. 2000. *Analysis of Throughput Performance of a Modified Hybrid ARQ Scheme for Reliable Transport Services*. Slovak Technical University, Bratislava Slovakia.

[3] Sklar, Bernard. 2001. *Digital Communications Fundamentals and Applications.* 2nd Edition. Prentice-Hall International Inc.

[4] John G. Proakis and Masoud Salehi. 2008. *Digital Communications.* 5th Edition. McGraw Hill.

[5] Elisabeth Uhlemann, *Hybrid ARQ Using Serially Concatenated Block Codes for Real-Time Communication - An Iterative Decoding Approach.* Chalmers University of Technology, Sweeden.

[6] Qinqing Zhang, Salemm A. Kassim. 1999. *Hybrid ARQ with Selective Combining for Fading Channels*, IEEE Journal on selected areas in Communications. 17(5).

[7] Hang Liu and Magda El Zarki. 1997. *Performance of H.263 Video Transmission Over Wireless Channels using Hybrid ARQ.* IEEE Journal on selected areas in Communication. 15(19).

[8] Stephen B. Wicker. 1995. *Error Control Systems for Digital Communication and Storage.* Prentice Hall.

[9] Devendra Sharma, *Introduction to High speed downlink packet access (HSDPA),* 2006. Available: http://www.3g4g.co.uk/TutorialDS/

[10] Meor Mohd Azreen b. Meor Hamzah. 2007. BER Performance of Convolutional Codes, MSc Thesis Dissertation, Universiti Yeknologi MARA, Malaysia.