Evaluation of rate compatible block turbo codes for multimedia application in satellite communication network

Sooyoung Kim¹,*⁺⁺⁺⁺, Sunheui Ryoo²,§ and Do Seob Ahn³,§

¹ Division of Electronics and Information Engineering, Chonbuk National University, 664-14 Deokjin-dong, Jeonju, 561-756, Korea
² Seoul National University, Korea
³ ETRI, Korea

SUMMARY

In this paper, we evaluate the performance of the hybrid automatic repeat request (ARQ) scheme combined with adaptive coding in satellite communication systems. A cross-layer design which combines the ARQ scheme in the data link layer and adaptive coding and modulation in the physical layer is very important to maximize system throughput and/or performance. Due to the long round trip delay in satellite systems, the strategy to adopt may need to be different from that is used in the terrestrial systems. A rate compatible code is a desirable scheme for adaptive coding and hybrid ARQ. In this paper, we propose an adaptive coding scheme with hybrid ARQ using rate compatible block turbo codes, and demonstrate its performance on a mobile satellite channel for multimedia applications. The proposed rate compatible block turbo codes are a very effective tool to improve the performance of satellite systems using adaptive coding with hybrid ARQ. The simulation results presented in this paper reveal that the hybrid ARQ scheme combined with adaptive coding can improve the system performance by effectively compensating for the various channel attenuations. Copyright © 2006 John Wiley & Sons, Ltd.

Accepted 5 July 2006

KEY WORDS: satellite; communication; error correction; block turbo code; ARQ; adaptive
1. INTRODUCTION

The key feature of future fourth-generation (4G) wireless communication systems will be providing high-speed data transmissions and interactive multimedia applications beyond simple data transfer. For effective delivery of these services, which should be bandwidth-intensive, delay-sensitive, and loss tolerant, quality of service (QoS) metrics will be more stringent. This situation calls for QoS-specific technologies including cross-layer optimization. In this context, considerable research effort have been addressing the issue of cross-layer optimization and its effectiveness in providing an improved solution with respect to the trade-offs in multimedia quality, power consumption, implementation complexity, and spectrum utilization [1–3].

Traditionally, the resource management, adaptation, and protection strategies available in the lower layers are optimized without explicitly considering the specific characteristics of multimedia applications. In other words, multimedia compression and streaming algorithms do not consider the mechanisms provided by the lower layers for error protection, scheduling, resource management, and so no. This layered optimization leads to a simple independent implementation, but results in suboptimal performance [1]. Therefore, the cross-layer optimization framework is one of the most important topics which need to be considered to achieve high-quality multimedia transmission at low cost.

Mobile satellite systems will be fully integrated within 4G systems, playing important roles in the areas where terrestrial communications infrastructure is very expensive and impractical or satellite communications may be the only solution, due to the wide area coverage, reconfigurability, and multicast capabilities [4]. In this sense, maintaining the compatibility with terrestrial systems is another important issue. In this paper, we will discuss hybrid automatic repeat request (ARQ) and adaptive coding issue which is actively investigated in various terrestrial system standards.

The technical challenges facing future broadband satellite systems were extensively discussed in Reference [5], and these include onboard processing and switching, mobility and resource management, IP routing, and cross-layer designs. Radio resource management for satellite networks including some cross-layer solutions was discussed in a tutorial sense in Reference [4]. Because satellite bandwidth is a relatively scare resource, the adoption of highly efficient modulation and coding schemes to countermeasure various channel impairments is mandatory in the physical layer aspects. Adaptive usage of the resources offered by various modulation and coding schemes is another prerequisite for ensuring the efficiency and economy of the system, as is the case in many future communication standards including Digital Video Broadcasting via Satellite (DVB-S2) [6]. This requires closer interaction between the medium access control (MAC) and physical layers for optimum resource allocations.

Our previous studies discussed the adaptive usage of advanced modulation and coding schemes for future mobile satellite systems [7–9]. We proposed an adaptive radio resource allocation scheme for mobile packet services in a synchronous multibeam satellite system by considering the physical-MAC layer interaction, and showed various interesting results including the effectiveness of a system-level adaptation [7]. An adaptive multi-carrier code division multiple access (MC-CDMA) scheme has been proposed and its performance was demonstrated in a Ka-band mobile satellite channel [9].

The hybrid ARQ is a classic example of a cross-layer approach, where end to end data reliability is provided by a combination of the physical layer, data link layer, and transport layers [4]. The authors in Reference [2] conducted a numerical analysis of the cross layer
integration of adaptive modulation and coding (AMC) with ARQ over a terrestrial wireless channel, and revealed that retransmissions in the data link layer relieve stringent error control requirements at the physical layer. Figure 1 demonstrates the cross-layer structure of the ARQ in the data link layer and AMC in the physical layer. We will see this structure in more detail in the next section, by adding hybrid ARQ schemes.

The situation in a terrestrial system may not be directly applicable to a satellite system, due to the long round trip delay (RTD) incurred in the system. Our previous studies showed the advantages of applying rate compatible block turbo codes (RC-BTC) to adaptive coding (AC) in conjunction with hybrid ARQ for satellite communication systems [10, 11]. In this paper, we propose an improved AC scheme with ARQ using the RC-BTC by considering multimedia transmissions via a satellite network. We first simulate the BER performance of the RC-BTC on an AWGN channel, and select the optimum RC code sets for the proposed scheme. We show that the proposed scheme considerably reduces the retransmission delay and improves the throughput performance in the satellite channel. We also investigate the performances in various aspects, which are averaged over on a heavy rainy day at a mobile satellite terminal.

In Section 2, we introduce the basic concept of RC-BTC, and show a few examples. We describe the proposed AC scheme combined with hybrid ARQ using the RC-BTC in Section 3, and discuss advantages of the proposed scheme. We present various simulation results on a mobile satellite channel in Section 4. Finally we draw our conclusions in Section 5.

### 2. RATE COMPATIBLE BLOCK TURBO CODES

Pyndiah et al. first introduced block turbo codes, which are product codes combined with iterative decoding algorithms [12]. After their introduction and detailed investigation [13], these codes have proven to be applicable to many areas, including satellite communications and wireless MAN systems [14–16]. In a block turbo code, serial concatenation with a block interleaver is normally used. Although parallel concatenation is also possible for block turbo codes, serial concatenation is usually preferred, leading to the so-called product codes.
Concatenation in serial guarantees a large minimum Hamming distance even for relatively small block size. Figure 2 shows the procedure for constructing classical two-dimensional (2D) product codes using a \((n_1, k_1)\) block code and a \((n_2, k_2)\) block code. Figure 3 shows the configuration of a 3D product code constructed in exactly the same way as in Figure 2.

It is theoretically possible to construct \(m\)-dimensional product codes for \(m\) larger than 2. Therefore, in the \(m\)-dimensional product codes we have \((n_1 \times n_2 \times \ldots n_m)\) bits (symbols) of encoded block [17]. In an \(m\)-dimensional product code, we can divide it into several \(m\)-dimensional blocks of which axes lengths are information length, \(k\) or parity length, \((n-k)\). For example, in the case of a 2D product code, there are four 2D blocks including an information block and three parity blocks. Similarly, a 3D product code consists of eight 3D blocks including an information block and 7 parity blocks as shown in Figure 3. Generalizing this, we see that an \(m\)-dimensional product code consists of an information block and \(2m-1\) parity blocks. This is because the number of existing \(m\)-dimensional blocks is exactly the same as the number of ways of allocating the two different lengths of \(k\) and \((n-k)\) at each axis of the \(m\)-dimensional blocks [18].

![Figure 2. Procedure for constructing 2D product codes.](image)

![Figure 3. Information block and parity blocks in a 3D product code.](image)
Now, we can produce RC codes by making several combinations of these consisting blocks. To make it easy, we identify each $m$-dimensional block by indexing it with a binary number, $(b_1, b_2, \ldots, b_m)$ with the following rule. We assign 0 to $b_i$ if the length of the $i$th axis in the block is $k_i$, or 1 if it is $(n_i - k_i)$. For example, we can represent each block in the 3D product code of Figure 3 as follows: $I = (0,0,0)$, $P_1 = (1,0,0)$, $P_2 = (0,1,0)$, $P_3 = (1,1,0)$, $P_4 = (0,0,1)$, $P_5 = (1,0,1)$, $P_6 = (0,1,1)$, $P_7 = (1,1,1)$. By analysing the above binary representation rule in detail, we see that the information block is always represented by all zeros.

To make a code, we need the information block and more than one of the parity blocks which are adjacent to the information block. If two blocks are adjacent to the information block in an $m$-dimensional product code, then ($m$-1) positions of their binary representation should be in agreement. Therefore, we can make an RC product code by using these properties [18]. Figure 4 shows an example of an RC product code using a 3D product code and illustrates how we can form it by using various combinations of parity blocks. All of the patterns in Figure 4 can be seen as punctured codes obtained from the 3D product code.

In this paper, we consider only those combinations which can configure at least as 2D codes in order to use an iterative decoder. For example, eight variations in the code rate are possible if we use the same component code in each axis of the 3D code. We can make larger number of variations if we use a different component code in each axis, for example 18 variations in the case of 3D codes.

3. ADAPTIVE CODING AND HYBRID ARQ USING RC-BTC

3.1. Comparison of hybrid ARQ schemes for mobile satellite systems

We classify the hybrid ARQ schemes into three categories namely type-I, type-II and type-III schemes [19]. In this section, we summarize the characteristics of each hybrid ARQ scheme,
especially for mobile satellite applications. A type-I scheme uses the same channel code for each transmission. The performance of type-I hybrid ARQ relies heavily on the error correction capability of the code that is used. This scheme is not adequate for mobile satellite channel conditions because the fixed channel code cannot manage the limited channel resource efficiently as the satellite channel conditions change.

In type-II scheme, the very first transmission packet has only information bits or a few parity bits for error correction. As the retransmission steps increase, incremental redundancy bits are added. Because this type retransmits only the added parity blocks, it would be a good solution for prevailed line of sight (LOS) conditions, such as open areas. A type-III scheme uses a complementary code as an error correction code. Its self-decodability is advantageous for heavy shadowing conditions, such as urban areas.

The original type-I hybrid ARQ does not combine with the previous erroneous packet. In this paper, we consider the Chase combining scheme for type-I and type-III schemes. The basic idea in the Chase combining scheme is to combine multiple received copies of the coded packet weighted by the SNR prior to decoding [20]. The Chase combining method provides diversity gain via a satellite channel.

Now let us explain the above hybrid ARQ schemes using the RC-BTC. Table I shows the retransmission rule for each hybrid ARQ scheme. For example, in type-I scheme we use any fixed code for every (re)transmission. In other words, whenever the transmission fails, we always transmit the same code. Referring to Figure 4, we transmit Code 1 consisting of I + P1 + P2 in the first transmission in type-II scheme. If the first transmission fails, we send the P3 block in the second transmission. If this fails again, we send P4, P5, P6, and P7 in the next four retransmissions, respectively. In type-III scheme, we send Code 2, Code 4, Code 5, Code 7 and Code 8 in every retransmission in that order after sending Code 1 in the first transmission.

### Table I. The retransmission rule for the hybrid ARQ schemes using the 3D RC-BTC.

<table>
<thead>
<tr>
<th>No. of transmissions</th>
<th>Type-I</th>
<th>Type-II</th>
<th>Type-III</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Any code selected</td>
<td>I + P1 + P2</td>
<td>I + P1 + P2 (Code 1)</td>
</tr>
<tr>
<td>2</td>
<td>The same code</td>
<td>P3</td>
<td>I + P1 + P3 (Code 2)</td>
</tr>
<tr>
<td>3</td>
<td>The same code</td>
<td>P4</td>
<td>I + P1 + P4 (Code 4)</td>
</tr>
<tr>
<td>4</td>
<td>The same code</td>
<td>P5</td>
<td>I + P1 + P5 (Code 5)</td>
</tr>
<tr>
<td>5</td>
<td>The same code</td>
<td>P6</td>
<td>I + P1 + P6 (Code 7)</td>
</tr>
<tr>
<td>6</td>
<td>The same code</td>
<td>P7</td>
<td>I + P1 + P7 (Code 8)</td>
</tr>
</tbody>
</table>

3.2. Adaptive coding with hybrid ARQ using the RC-BTC

Figure 5 shows a schematic diagram of the ARQ system with AMC. This is the cross-layer structure in Figure 1 seen from a communication link. We will apply the RC-BTC to this link for AC with hybrid ARQ. For the data packet from the data link layer, where the packet size is controlled for adaptive coding and/or hybrid ARQ in the physical layer, the encoder generates a product code. Then, in the buffer, the encoded block is divided into an information block and several parity blocks. We send a combination of these blocks corresponding to the given transmission by the control command from the data link layer for adaptive coding and ARQ.
The control command from the adaptive controller in the data link layer selects the most appropriate coding schemes to the estimated channel quality information (CQI), and the command from the ARQ controller determines the retransmission by the results from the packet error checking. In this paper, we concentrate on AC by using a fixed modulation scheme, BPSK, because the purpose of this paper is to investigate the behaviour of the hybrid ARQ scheme combined with AC using the RC-BTC, rather than to investigate the AMC itself.

Figure 6 shows an example of the channel response in a mobile satellite system, in which we have rain attenuation, shadowing, and small-scale fading. We divide the channel SNR according to the error control capability of the RC codes as shown in Figure 6. We select a target code, $C_i$, having the maximum throughput for the previous observation period of length $L$ under a current estimated SNR such that

$$\arg\max_{C_i} \left[ \sum_{k=0}^{L-1} \lambda \cdot S_{C_i} \right]$$

where $C_i$ is the targeted coding scheme among $D$ RC block turbo codes and $D$ is the number of available transmission codes. $S_{C_i}$ is the spectrum efficiency of the channel code $C_i$ and calculated by the value of $1 - P_{C_i}(\gamma)$, where $P_{C_i}(\gamma)$ is the bit error rate (BER) for the predicted SNR, $\gamma$. The weighting factor, $\lambda$, is inversely proportional to the slope of the SNR, $\Delta\gamma$. When the slope of the SNR slope increases, it means that the channel changes rapidly, in which case a smaller $\lambda$ decreases the effect of previous CQI [16].

For multimedia transmission, we consider a target BER of $10^{-5}$ which may correspond to the requirement of real time streaming services using MPEG-4 audio/video transmission. Figure 7
shows the BER performance of the RC-BTC in Table I, where we use the 3D product code consisting of the (16,11) extended BCH code. We first apply the AC scheme based on Equation (1). For example, at time $t_0$ in Figure 6, we send code $C_i$ which can satisfy the target performance in the SNR range of $SR_i$. The receiver sends the result of the transmission in the form of a positive acknowledgement (ACK) or negative acknowledgement (NACK). Upon receiving this acknowledgement, the transmitter decides the retransmission. The receiver
combines the transmitted packets if necessary to offer more powerful error correction to recover the information. We assume that the return channel for sending ACK and NACK is error free.

If we use a hybrid ARQ scheme without combining the AC scheme, the transmission will start with code $C_1$ and will fail until the $i$th retransmission. In this paper, we propose to use an AC scheme with hybrid ARQ by using the RC-BTC. By this way, we can reduce transmission failure at the first transmission, and thus reduce the number of retransmissions especially in deep fading conditions. Compared with the AC scheme without hybrid ARQ, the combined scheme is advantageous in the low SNR range. Retransmissions and power combining in hybrid ARQ can extend the operating range of mobile satellite systems resulting in power savings and increased capacity.

In particular, the proposed RC-BTC scheme is very efficient for hybrid ARQ in mobile satellite systems. This is because we can use exactly the same decoding algorithm in each component code of the RC block turbo codes and the decoding procedures of the component codes are independent of each other. This means that the decoding procedure can be implemented in parallel, and we can combine the extrinsic information from the decoding results of the previous retransmissions. This multiple use of extrinsic information and parallel processing can reduce the decoding delay and improve the decoding performance, and may be advantageous for satellite systems which are generally power-limited and have a long RTD.

4. SIMULATION RESULTS

We first demonstrate the performances of hybrid ARQ and AC schemes on a mobile satellite communication channel in terms of transmission efficiency and delay, and then will investigate the simulation results in more detail by considering multimedia transmission through the satellite network. We consider a geostationary satellite system and use the mobile satellite channel model in Reference [21]. The received signal level is characterized by an embedded Markov chain with three states including the LOS condition, the moderate shadowing condition, and the deep shadowing condition. These fading states are generated by the steady-state probabilities and state transition probabilities for a suburban environment. The state duration is modelled by an exponential distribution with an average duration of $t_f$, given by

$$t_f = \frac{d_c}{v_u}$$

where $d_c$ and $v_u$ are the fading correlation distance and user speed, respectively.

Ka-band link is vulnerable to rain attenuation, and thus we also generated dynamic rain attenuation [16], so that the transmitted data would experience dynamic behaviours of rain attenuation, shadowing and small-scale fading. We generate rain attenuation samples with an interval of about a second. Figure 6 shows a part of the channel response generated by the channel model, and Table II shows the main simulation parameters. We implemented transmission the link shown in Figure 5, which has rain attenuation, mobile fading, and an AWGN in the channel.

For the RC-BTC code, we used a 3D product code consisting of the $(16,11)$ extended BCH code in each axis. We limit the maximum number of transmissions to six, which makes it possible to construct full 3D product codes, as shown in Table I. The selective repeat retransmission method can be an efficient solution for such a system. In this paper, we do not
consider the overhead introduced by the retransmissions, because we are only concentrating on the comparison of the performance of the (hybrid) ARQ schemes with that of the proposed RC-BTC.

We define the transmission efficiency, $T_e$ as the ratio of the number of the correctly decoded information bits to the number of bits transmitted in total. That is,

$$T_e = \frac{\sum_{i=1}^{N} K_i \delta_i}{\sum_{i=1}^{N} N_i}$$

where $K_i$ is the number of information bits for the $i$th transmitted packet and $N_i$ is the total number of coded bits (re)transmitted to recover the $K_i$ information bits. $N$ is the number of simulation runs we performed to estimate the packet error rate. $\delta_i$ is the indicator of the transmission results, and is defined by

$$\delta_i = \begin{cases} 1 & \text{if } K_i \text{ information is transmitted successfully} \\ 0 & \text{otherwise} \end{cases}$$

The number of information bits, $K_i$ will be the size of the information block, and is fixed, regardless of the retransmission scheme. On the other hand, $N_i$ varies according to the channel conditions and the hybrid ARQ schemes. $N_i$ is the sum of the transmitted bits at each retransmission, and can be expressed as follows:

$$N_i = \sum_{j=1}^{r_i} N_{ij}$$

where $N_{ij}$ is the number of coded bits for the $j$th retransmission for the $i$th transmitted packet and $r_i$ is the number of retransmissions for the $i$th packet. For easy comparison, we normalized $T_e$'s with the maximum transmission efficiency of Code 1, which is

$$NT_e = \frac{T_e}{T_{e,\text{Code1}}}$$

Table II. Simulation Parameters.

<table>
<thead>
<tr>
<th>Simulation parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Downlink carrier frequency</td>
<td>20 GHz</td>
</tr>
<tr>
<td>Modulation scheme</td>
<td>BPSK</td>
</tr>
<tr>
<td>Information bit rate</td>
<td>5 Mbps</td>
</tr>
<tr>
<td>Round trip delay</td>
<td>0.5 s (bent-pipe)</td>
</tr>
<tr>
<td>Frame duration</td>
<td>20 ms</td>
</tr>
<tr>
<td>Fading correlation distance ($d_{corr}$)</td>
<td>5 m</td>
</tr>
<tr>
<td>Mobile speed</td>
<td>70 km/h</td>
</tr>
<tr>
<td>Rain attenuation (maximum depth)</td>
<td>$-35$ dB</td>
</tr>
<tr>
<td>Rain attenuation (fading rate)</td>
<td>0.56 dB/s</td>
</tr>
<tr>
<td>Simulation time</td>
<td>24 h in rainy days</td>
</tr>
<tr>
<td>Required BER</td>
<td>$10^{-3}$</td>
</tr>
<tr>
<td>ARQ scheme</td>
<td>Selective repeat</td>
</tr>
<tr>
<td>RC-BTC code</td>
<td>3D code with the (16,11) code</td>
</tr>
</tbody>
</table>

and

\[ T_{\text{max}}^{e_{\text{Code1}}} = \frac{k_1 k_2 k_3}{k_3(n_1 k_2 + k_1(n_2 - k_2))} \]  

(7)

For our example code, \( T_{\text{max}}^{e_{\text{Code1}}} \) is about 0.52.

Figure 8 shows the normalized transmission efficiencies, \( N T_e \), of the three hybrid ARQ schemes using the proposed RC-BTC listed in Table I, and compares them with that of the same 3D BTC without any retransmission over the mobile satellite communication channel. As we expected, we cannot use the code without the hybrid ARQ techniques in the low \( E_s/N_0 \) ranges. In other words, we have to compensate for rain attenuation, shadowing and other fading effects using an efficient hybrid ARQ scheme under such low power conditions. The type-I scheme using Code 1 or Code 8 produces better efficiency than the ones without retransmissions, before the efficiency values are saturated. Over the saturation range, the hybrid ARQ schemes incur extra cost for the additional retransmissions. For the fixed error protection codes without incremental redundancy, the lower rate Code 8 has better performance than Code 1 in the low \( E_s/N_0 \) range, while its efficiency is limited in the higher \( E_s/N_0 \) range. This is due to its higher error correction capability being at the expense of the code rate (bandwidth efficiency).

Type-II and type-III schemes using the proposed RC-BTC can take advantages of each individual component code. We send the highest rate code at the first transmission. If this fails, the lower rate code is retransmitted. Because the retransmitted packet in the case of the type-II hybrid ARQ contains only incremental redundancy blocks, it produces better throughput efficiency than the other hybrid ARQ schemes in the relatively high \( E_s/N_0 \) range. However, under severe fading conditions, type-I schemes with Code 8 outperforms the other schemes due to its high error correction capability and the diversity gain from the Chase combing.
Now we will demonstrate the performance of the proposed scheme that combines ARQ and AC schemes. By this way, we can further improve the performance. In this method, we send the most appropriate code to the estimated $E_s/N_0$ value at the first transmission instead of the highest rate code. By doing this, we can reduce the failure at the first transmission, and thus reduce the number of retransmissions, especially under severe fading conditions. Figure 9 shows the enhancement of the throughput obtained by combining the AC scheme with hybrid ARQ. In Figure 9, C-ARQ stands for the conventional ARQ scheme without incremental redundancy.

In Figure 10, we compare the delay performance in terms of the number of transmission attempts per information packet over the mobile satellite channel. In general, the number of transmissions decreases with increasing $E_s/N_0$ and is inversely proportional to $NT_e$. A lower rate code gives rise to a better error correction capability and thus requires a lower number of retransmissions at the cost of a limited maximum efficiency. As in the efficiency comparison, we can increase the delay performance by combining hybrid ARQ with the AC scheme. Figure 11 compares the delay performance of the various hybrid ARQ and/or AC schemes, and it shows that the combined scheme reduces the number of retransmission attempts, $n_T$ by about 15–75% more than the hybrid ARQ only scheme in the $E_s/N_0$ range of less than 6 dB.

Figure 12 shows $NT_e$s of the hybrid ARQ schemes according to $n_T$. We can see that type-II hybrid ARQ scheme has better $NT_e$ value than the other types of hybrid ARQ schemes. However this evaluation does not take power efficiency into consideration. Referring to Figure 10, type-II needs a much greater number of retransmissions at a given $E_s/N_0$ than the other hybrid ARQ schemes.

Figure 13 shows the delay versus transmission efficiency performances of the hybrid ARQ type II with AC scheme that shows the best performance in Figures 9 and 11, as compared to the

![Figure 9. Normalized transmission efficiency comparison of hybrid ARQ and adaptive coding schemes.](image-url)
Figure 10. Delay comparison of hybrid ARQ schemes.

Figure 11. Delay comparison of hybrid ARQ and adaptive coding schemes.
AC only and AC with conventional ARQ (C-ARQ) schemes. The AC scheme itself cannot compensate for deep fading in the low SNR range, even though we use the best code for the estimated SNR. On the other hand, the AC with hybrid ARQ scheme can improve the

Figure 12. Normalized transmission efficiency versus the number of transmission attempts.

Figure 13. Delay-transmission efficiency performance at various $E_s/N_0$ values.
performance by using a suitable retransmission schemes. Figure 13 also shows that the proposed hybrid ARQ with the AC scheme requires the least SNR at a given delay and transmission efficiency value. For example, at the \( n_T \) value of 2 and \( N_T \) value of about 0.34, the required \( E_s/N_0 \) for the proposed AC with hybrid ARQ scheme is \(-5\) dB, while that of the AC with C-ARQ scheme is about \(-3.5\) dB. This indicates that the proposed scheme requires less power at a similar performance level in terms of both the efficiency and delay. If we use the AC only scheme, we need an \( E_s/N_0 \) value of \(-3.6\) dB in order to achieve an \( N_T \) of about 0.34. If we fix the \( E_s/N_0 \) value, we can see the advantage of the AC with hybrid ARQ scheme more clearly. At an \( E_s/N_0 \) value of \(-5\) dB, the \( N_T \) of the AC with hybrid ARQ scheme is about 0.34. On the other hand, the AC with C-ARQ scheme achieves just one third of the \( N_T \) even with a greater number of the \( n_T \) value. There is no \( N_T \) value for the AC only scheme at the same \( E_s/N_0 \) value. This is the case for almost all of the ranges in Figure 13.

The above simulation results can be interpreted as the reduction of power burden in physical layer due to a cross-layer optimization. The performance enhancement by the AC scheme should be accompanied by a precise CQI or a complex equipment to estimate CQI in the physical layer. In addition, the long RTD in satellite systems prevent us from adapting to fast time-varying channel conditions. The retransmissions by the ARQ scheme may be effective solution to this.

We will investigate this in more detail by comparing the performances of the AC only scheme and the AC scheme with hybrid ARQ type II. At this time, we assume imperfect CQI, and we predict the SNR with a simple first order FIR filter using the previous values. In order to maximize the efficiency, we select the optimum code set. For these two candidate schemes, we use five Codes in Table I including, Code 1, 2, 4, 7, and 8. This is because the BER performances of Code 5 and Code 7 are almost the same as shown in Figure 7.

The maximum fading depth in Ka-band mobile satellite link with rain attenuation reaches up to about 35 dB, we assume that our AC with hybrid ARQ scheme would compensate 5 dB of attenuation. We also assume that the remaining 30 dB of attenuation can be compensated by an adaptive modulation schemes and power control without loss of generality. We simulated the performance of two schemes with the Monte Carlo approach by using about \( 8.5 \times 10^9 \) bits of random binary data. We generated dynamic rain attenuation combined with mobile fading for 24 h in a heavy rainy day using the parameters in Table II.

Table III shows the performances of the AC only scheme and AC with hybrid ARQ type II scheme in terms of BER, \( N_T \), and \( n_T \). We averaged these performance meters over the whole observation time range. As we can see if we increase the number of retransmissions about 20%, we can increase the transmission efficiencies about 30% as well as increase the BER performance

<table>
<thead>
<tr>
<th>Scheme</th>
<th>AC only</th>
<th>AC+Hybrid ARQ type II</th>
</tr>
</thead>
<tbody>
<tr>
<td>BER</td>
<td>( 1.41 \times 10^{-2} )</td>
<td>( 1.19 \times 10^{-6} )</td>
</tr>
<tr>
<td>( N_T )</td>
<td>0.70</td>
<td>0.92</td>
</tr>
<tr>
<td>( n_T )</td>
<td>1.0</td>
<td>1.2</td>
</tr>
</tbody>
</table>
tremendously. If we consider delay-sensitive multimedia transmissions through satellite networks, the proposed RC-BTC code can be used efficiently to improve the performances.

5. CONCLUSIONS

Retransmission and power combining in hybrid ARQ can extend the operating range of mobile satellite systems, resulting in power saving and increased capacity. In this paper we propose an RC-BTC scheme for adaptive coding and hybrid ARQ schemes, and demonstrate the performance on mobile satellite channel. In the proposed scheme, by virtue of BTC structure, we can use the same decoding algorithm at every RC code. The multiple usage of extrinsic information and capability of parallel decoding in RC-BTC can reduce the decoding delay and also improve the decoding performance, and may give a great merit to a satellite system which is generally power-limited and have long RTD. Simulation results on rainy mobile satellite channel show that if we combine AC and ARQ schemes with the proposed RC-BTC, the performance can be highly increased with just small increase in delay.

ACKNOWLEDGEMENTS

This work was supported in part by the Korean Ministry of Information and Communication.

REFERENCES