Efficient Error Recovery for Multimedia Data Transmission over 3G Cellular Broadcast Networks

Joonho Lee, Kyungtae Kang, Yongwoo Cho, Junu Kim and Heonshik Shin
School of Electrical Engineering and Computer Science
Seoul National University, Seoul, Korea 151–744
{jlee108, ktkang, xtg05, kesker, shinhs}@cslab.snu.ac.kr

Abstract

We analyze the execution time of MAC-layer Reed-Solomon decoding in cdma2000 1xEV-DO broadcast and multicast services (BCMCS), using different cache sizes and a simulated air channel. MPEG-4 video is widely used for wireless multimedia services; however, it requires successive frames within a specific time interval, which can be missed by a MAC-layer Reed-Solomon decoder due to its computation time. We propose static and dynamic proactive Reed-Solomon bypass (PRSB) schemes to alleviate this problem in mobile nodes. Our first scheme bypasses the Reed-Solomon decoding process to satisfy the MPEG-4 time constraint statically, despite varying channel conditions. The second, dynamic scheme corrects errors in a best-effort manner within this same time constraint, leading to a further quality improvement. Extensive simulation results show a dramatic quality improvement in video quality.

1 Introduction

A wireless radio channel has a much higher error-rate than that of expected on a wired link. Additionally, errors occur in clusters or bursts, with relatively long error-free intervals between them. Error control is therefore mandatory when data are sent over wireless networks to an end-user. Forward error correction is the scheme most commonly used when sending a broadcast video stream over a lossy network such as a wireless radio channel.

Broadcast and multicast services (BCMCS) are currently being standardized by various mobile wireless standards bodies such as 3GPP and 3GPP2 [1]. These services will allow users to receive a variety of content simultaneously on their handsets over cellular links. In BCMCS, Reed-Solomon (RS) coding is used as the method of forward error correction, and can efficiently conceal error clusters [2][3]. However, error correction must also consider the timing constraints which are characteristic of multimedia streams. A video frame is useful only if it arrives at the mobile node before its playback time. The current cdma2000 1xEV-DO chipset used for unicast transmission is not designed to minimize delay, and timing problems are therefore inevitable.

MPEG-4 specifies how a multimedia stream is transmitted, and is the data standard most widely used with BCMCS. In this study, we analyze the execution time of the RS forward error correction (FEC) scheme. A long decoding time makes it difficult to satisfy MPEG-4 time constraints if there are a lot of errors. If a frame does not arrive on time, the video quality degrades dramatically. Furthermore, once this time constraint has been exceeded, it causes a system reset, which leads to significant quality degradation of the video. We therefore propose two ways of satisfying the timing constraints of real-time multimedia applications: the static and dynamic proactive Reed-Solomon bypass (PRSB) schemes.

The remainder of this paper is organized as follows: In Section 2, we introduce the background to our research and, in Section 3, we illustrate the basic idea behind our proposed schemes, which is formulated in Section 4. The environment and the simulation results are then described in Section 5, followed by conclusions in Section 6.

2 Background

2.1 FEC for 3G cellular broadcast networks

Error control in BCMCS is provided by forward error correction (FEC) using Reed-Solomon encoding [2][3]. Its structure is shown in Fig. 1. At the access network, higher-layer packets are divided by the broadcast framing protocol, and the broadcast security protocol is used to encrypt the framing packets. The broadcast MAC protocol defines the procedures used to transmit over the broadcast channel and

Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06)
1550-445X/06 $20.00 © 2006 IEEE

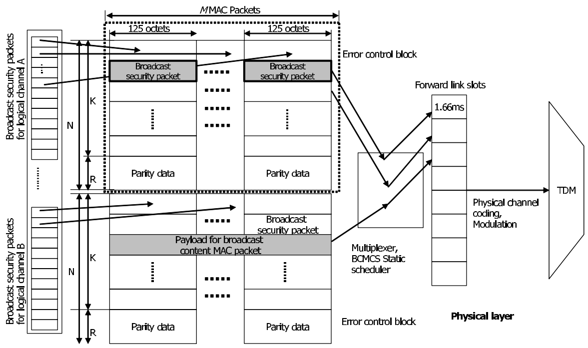


Figure 1. Error recovery structure for current BCMCS.

specifies an outer code which, when added to the physical layer turbo code, forms the product code. RS code is used as the wrapper code for cdma2000 BCMCS, and the size of broadcast MAC layer packets is fixed at 125 bytes. Once the protocol has been followed at the broadcast physical layer, an error control block (ECB) is transported as payload on one or more subchannels of this layer. Various data from multiple error control blocks will be multiplexed on to the broadcast physical channel.

Encoded ECBs with the same RS parameters (N, K, R) are used by each logical channel. This logical channel has M MAC packets per ECB row. The N, K and R parameters represent the total number of octets, the number of security layer octets and the number of parity octets in an RS code word, respectively. This means that an RS decoder can recover up to R erased octets in each code word. RS coding is applied to each column of the ECB, and this data is then transmitted row by row to form one or more physical-layer packets. The ECB structure means that, if a packet fails to arrive, octets in the same position are lost from all affected RS code words. Decoding an RS code word requires the broadcast MAC protocol to receive at least K of the N octets in the code word. However, if all data are received without any errors, they are forwarded directly to the upper layer of the BCMCS protocol suite.

RS decoding involves two phases: error detection and correction. In the error correction phase, the Berlekamp algorithm is used to fix the error, and this algorithm consumes most of the decoding time. As the number of errors increases, the execution time of the Berlekamp algorithm grows and proportionally increases the execution time of RS decoding.

2.2 Real-time constraints imposed by MPEG-4 video streams

MPEG-4 is one of the best-known multimedia systems that is suitable for cdma2000 1xEV-DO BCMCS [2]. The transport stream system target decoder (T-STD) model of an MPEG-4 system provides a guideline for managing timing constraints and buffers in an MPEG-4 transport stream (TS) decoding process [4][5].

There are three types of decoders in the T-STD: video, audio, and systems. For proper decoding of an MPEG-4 TS, the system requires a transport buffer (TB) to store the de-multiplexed packetized elementary stream (PES), and a main buffer (B) to feed appropriate access units into each decoder [4][5]. These buffers are required to provide exact timing references to synchronize the system target decoding processes, which include de-multiplexing of each elementary stream (ES) received from the transport stream, decoding of the de-multiplexed ES, and presentation of decoded video and audio. By using the same $system\_clock\_frequency$ (SCF), these three decoders (system, video and audio) can be operated synchronously and will have a presentation unit available at the proper time. An SCF is transmitted to the system target decoder (STD) by means of the program clock reference (PCR), which is prepared during the system encoding process. The PCR then generates a decoding time stamp (DTS) and a presentation time stamp (PTS).

The timing of each process in the system decoder is determined by the dynamic PCR and the static $transport\_rate$. If $PCR(i)$, which is an input to the system decoder, has been corrupted, then the SCF will also be corrupt, as follows:

$$SCF_{corrupted} = \frac{PCR(i)}{t_{corrupted}(i) - \frac{i-i''}{transport\_rate}}. \tag{1}$$

This corruption of the SCF will sequentially corrupt all the timing references, resulting in a system crisis, because all the rates in the system decoder will be modified. Once a system crisis has occurred, the system decoder will stop all the decoding processes, initialize every buffer and timing count, and then obtain a fresh PCR from which to calculate the SCF, allowing the system decoder to be re-initialized. Thus, in an MPEG-4 system, correct timing reference requires that certain rules are followed.

According to the MPEG-4 specifications [4][5], the TS should be constructed so there is a time interval of no more than 100ms between the bytes containing the last bit of the $program\_clock\_reference\_base$ field in successive occurrences of the PCR in the transport stream packets containing the PCR Packet ID (PCR_PID) for each program:

$$|t(i) - t(i')| \leq 100\text{ms}, \tag{2}$$

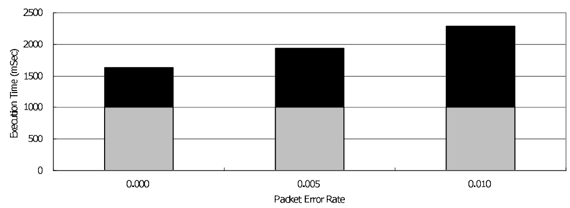


Figure 2. Execution time of RS decoding on a cacheless system.

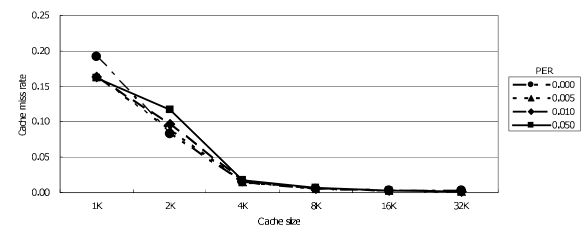


Figure 4. D-cache miss rate against cache size.

for all i and i', where i and i' are the indexes of the bytes containing the last bit of each consecutive $program\_clock\_reference\_base$ field in the transport stream packets of the PCR_PID for each program. There should be at least two PCRs, from the specified PCR_PID within a transport stream, between consecutive PCR discontinuities to facilitate phase locking and to permit extrapolation of byte delivery times. Therefore, an MPEG-4 system should have a maximum delay of 100ms from input to system decoder.

3 Analysis of current Reed-Solomon decoding technique

The one of the most widely used chipset is the QUALCOMM MSM5000 Mobile Station ModemTM (MSM), whose microprocessor is the ARM7TDMI core. We therefore used an ARM7TDMI testbed.

The decoding times for different values of the packet error rate (PER) were measured without a cache. We found that values of PER at the physical layer have a linear relationship with execution time, as shown in Fig. 2. In this figure, grey and black areas indicate the timing deadline of the upper-layer service and the excess time required for RS decoding, respectively. When the PER is 0.01, decoding takes more than 2000ms. This is an intuitively plausible result because more errors means more time spent running the Berlekamp algorithm. From this measurement, we see that RS decoding without a cache exceeds the time constraint of an MPEG-4 system. The issue is therefore reduced to determining the best cache size for RS decoding.

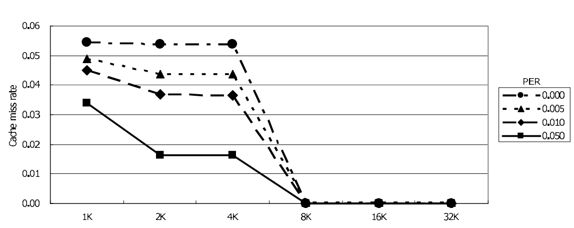


Figure 3. I-cache miss rate against cache size.

We measured the miss rates of the instruction and data caches while varying their size between 2k and 32k. The results shown in Figs. 3 and 4 indicate that cache size and miss rate are inversely proportional up to a cache size of 8k, but after that there is negligible improvement. The cache miss rate is stabilized if both instruction and data cache are larger than 16k.

These two measurements suggest that, as the size of the caches increases, the execution time of RS decoding is reduced. We conducted an experiment to measure the execution time of RS decoding with instruction and data caches both sized at 16k, because the cache miss ratio stabilizes when both caches reach 16k. The results, shown in Fig. 5, indicate that execution time and PER have a linear relationship, which we can use to decide whether RS decoding can be performed within the MPEG-4 time constraint. With a PER of 1%, the execution time is around 40ms, growing to 300ms when the PER is 4.5%. From this graph, we can see that RS decoding can be completed on time up to a PER of 1%, at which point decoding takes a little less than 100ms.

4 Static and dynamic PRSB schemes

4.1 Static PRSB scheme

Timing is the main constraint on an MPEG-4 system, especially the 100ms ceiling on the PCR interval. Thus, we propose a static PRSB scheme based on the idea of anticipating whether RS decoding can be completed before this deadline, using the linear relation between PER and execution time in a mobile node. This scheme omits RS decoding if it cannot meet the MPEG-4 system timing constraint. The pseudocode is shown in Fig. 6. Let us denote P_{max} as the maximum PER for which RS decoding can be completed in

Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06)
1550-445X/06 $20.00 © 2006 IEEE

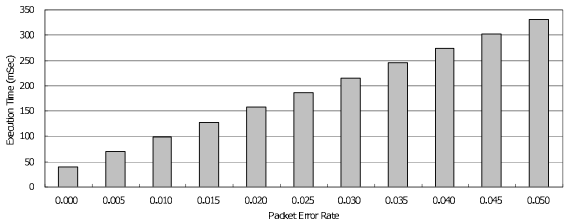


Figure 5. RS execution time against PER.

time. In the previous section, we saw that this P_{max} threshold is 1%. When an ECB block arrives at the RS decoding layer, the mobile node also receives the PER from the physical layer. If the received PER, P, is greater than P_{max}, the node will omit RS decoding. Otherwise, it will perform RS decoding and correct the errors.

A shortcoming of the static proactive Reed-Solomon bypass scheme is that RS decoding may sometimes not be performed even if there is sufficient marginal time within the PCB interval. Static PRSB causes all the data to bypass RS decoding if the PER is greater than 1%, even though the errors might be fixed in a few milliseconds, and the MPEG-4 time constraint satisfied.

4.2 Dynamic PRSB scheme

The dynamic PRSB scheme does not cut off at P_{max}. Instead, it will correct as many errors as it predicts, based on the relationship between execution time and the number of errors, within the time constraint. The MPEG-4 time constraint of 100ms is denoted by T_{max} and the remaining time is denoted by T, as shown in Fig. 7. We define the function $f_{(n)}$ to be the execution time of RS decoding with n errors. Initially, T is set to T_{max}. Whenever the RS decoder performs error correction, it will calculate $f_{(n)}$ for the current sub-block and compares it with T. If T is less than $f_{(n)}$, the decoder skips the current sub-block and starts on the next one. This procedure continues until it reaches last sub-block or there is no time left to fix any more errors.

To determine $f_{(n)}$, we measured the time required to correct one-sub block of the ECB. We repeated this measurement to find a worst-case value, which provides a maximum bound on the timing of RS decoding. These results, set out

```
Pmax = threshold_PER;
P = received_PER;
if (P < Pmax){
       while (there_is_a_sub_block) {
              do_RS_decoding();
       }
}
```

Figure 6. Pseudocode of static PRSB.

```
Tmax = threshold_Time;
T = Tmax
while (there_is_a_sub_block || T > f(0)) {
       n = number_of_errors;
       if (f(n) < T) {
              do_RS_decoding();
              T = T - f(n);
       }
}
```

Figure 7. Pseudocode of dynamic PRSB.

in Fig. 8, show that the execution time and the number of errors are proportional, so that $f_{(n)}$ is linear.

The execution time of RS decoding is dependent on the CPU and can be measured using a variety of tools. Thus, dynamic PRSB can be applied to any environment if this execution time is known.

5 Performance evaluation

5.1 Experimental environment

We will now analyze how many errors can be corrected within the MPEG-4 time constraint. We generated a template data-stream with errors that model those of an actual air channel, by injecting errors generated by a channel error model. In this study, we used the simple threshold model suggested by Zorzi [6] to simulate the behavior of data errors which arise in transmission over fading channels. This model can be represented as a binary Markov process in which the receiver is deemed to have received a data bit when the fading envelope of that bit is more than some threshold value. If the fading envelope is below the threshold, receipt fails. A first-order two-state Markov process can simulate the error sequences generated by data transmission on a correlated Rayleigh fading channel: these errors occur in clusters or bursts with relatively long error-free intervals between them. Fading in the air channel is assumed to have a Rayleigh distribution.

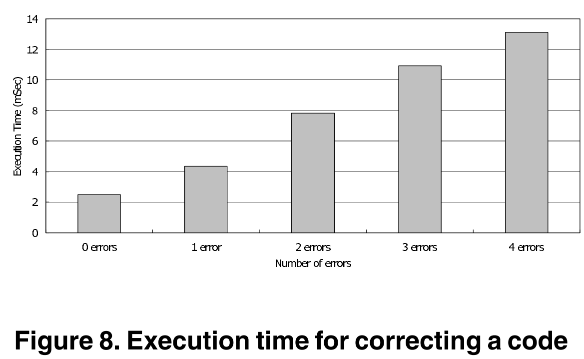


Figure 8. Execution time for correcting a code word without errors.

Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06)
1550-445X/06 $20.00 © 2006 IEEE

By choosing different values for the physical-layer bit error-rate ($\varepsilon_{physical}$) and for $f_d NT$, we can model different degrees of correlation in the fading process of radio channels. The value of $f_d NT$ determines the correlation properties, which are related to the mobile speed for a given carrier frequency. When $f_d NT$ is small, the fading process has a strong correlation, which means long bursts of errors (slow fading). Conversely, the occurrence of errors has a weak correlation for large value of $f_d NT$ (fast fading). In these experiments, we set the value of $f_d NT$ to 0.02, which represents moderate fading.

The performance of RS was measured using the SEE (SNU Energy Explorer) [7]. In this simulation, the ARM7TDMI core and the SEC 128Mbit SDRAM array (K4S280832A) were operated at 200MHz and 100MHz respectively. The target application is assumed to be a multimedia service and the proportion of CPU time used by RS decoding is assumed to be 5%. We used a bit-rate of 250kbps for processing the multimedia stream. During data transmission, physical packets are modulated by QPSK, which has a transmission rate of 1228.8kbps. Thus, the amount of data N that can be fitted into one time slot is 256 bytes.

5.2 Experimental results

We constructed a simulation to measure the peak signal to noise ratio (PSNR) [8] of the original Reed-Solomon scheme. This is shown in Fig. 9. We see that the average PSNR is 16.01dB. This is because RS decoding tries to correct every error and sometimes that requires no much computation that decoding overruns PCB interval.

As mentioned in Section 2, the SCF becomes damaged if the PCR does not arrive on time as a result of transgressing the input interval limitation of the MPEG-4 system decoder. Once the SCF has been damaged, it will successively damage the DTS and PTS. This leads to failure of the entire system and will cause re-initialization. There are many fluctuations of the PSNR but, once it drops to around 4dB, it stays at this level for a while. This corresponds to system re-initialization after a break due to violation of the MPEG-4 time constraint. Thus, we see how frequent system resets degrade the overall quality of a video.

We conducted an experiment to explore how much quality improvement can be achieved by our static and dynamic PRSB schemes. The results are shown in Fig. 10. We see that the average PSNR is 30.42dB with static PRSB and 32.84dB with dynamic PRSB. These results are 14.41dB and 16.83dB better than the original scheme, respectively.

With static PRSB, there are similar fluctuations to those experienced with the original scheme but, after a drop in PSNR, there is a rapid recovery to the average level. This is due to the omission of RS decoding. Since static PRSB skips the RS decoding phase if the PER threshold is exceeded, it will not cause system re-initialization, which improves the overall video quality.

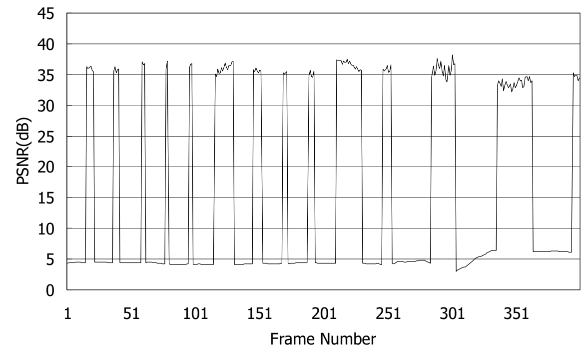


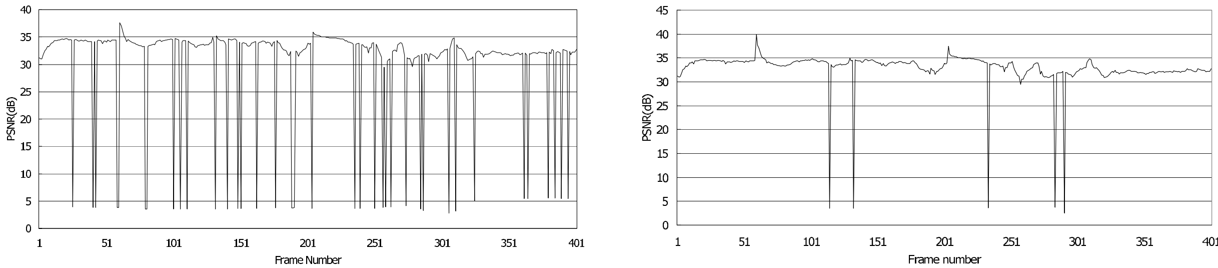
Figure 9. PSNR without either PRSB scheme (PER = 5%).

With dynamic PRSB, PSNR drops much less frequently. Furthermore, the improvement in average PSNR is 2.42dB greater than it is with the static scheme. Using static PRSB, the PER of the recovered data is the same as that of the input, because RS decoding is bypassed when the PER exceeds the threshold. But, because dynamic PRSB tries to recover as many errors as possible, its input PER is 5%, but falls to 0.8% after best-effort recovery. This shows the advantage of making an attempt at error recovery.

Now we turn our attention to the performance of static and dynamic PRSB with different values of PER. We experimented with PERs of 1%, 3%, 5% and 7%, and Fig. 11 shows how they altered the PSNR. With a PER of 1%, there is not much change in the PSNR, because RS decoding can be performed within the time constraint; but there is significant improvement if the PER is greater than 3%. A PER of 1% is the threshold for RS decoding and, beyond this error rate, our schemes improve video quality dramatically. These results indicate that, if quality is important, the system should be prevented from re-initialization, and our schemes significantly benefit the PSNR by satisfying this requirement.

6 Concluding remarks

In BCMCS, FEC is used as the error correction algorithm for broadcasts, instead of ARQ (automatic repeat request). We have developed a realistic environment to measure the performance of the RS algorithm which is used for FEC in cdma2000 1xEV-DO. We have explored the considerations necessary for implementing cdma2000 1xEV-DO BCMCS, by simulating RS decoding in the most widely



(a) With the static PRSB scheme (PER of 5%) (b) With the dynamic PRSB scheme (PER of 5%)

Figure 10. PSNR of a sample picture.

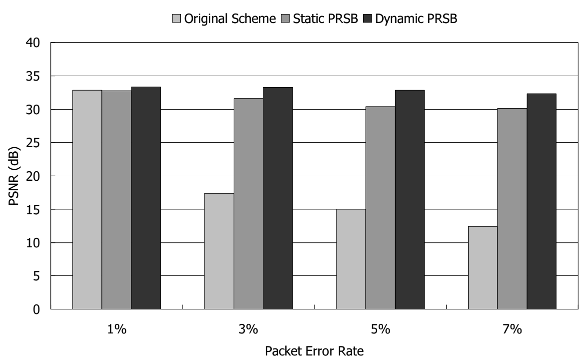


Figure 11. Comparison of PSNR with different values of PER.

used environment for a cellular service implementation. We measured the execution times of an RS decoder when there are between 0 and 4 errors, and the results of these tests suggested that several errors can be corrected within the MPEG-4 time interval.

If a packet delivery is delayed, the entire system has to be re-initialized, which dramatically degrades the quality of service. Therefore, it is critical to synchronize the timing in multimedia applications that use cdma2000 1xEV-DO BCMCS. The static PRSB scheme does this by skipping RS decoding for packets whose PER is higher than a threshold. This avoids the necessity of restarting the entire decoding process, although the quality of the multimedia service degrades temporarily as the error recovery rate decreases. Dynamic PRSB takes the same approach but also corrects as many errors as it can within 100ms. Our simulation results show that both of these schemes greatly increase the PSNR. Furthermore, dynamic PRSB can be applied to any CPU if the execution time of RS decoding on that device is analyzed.

Correcting errors in the base-layer of a video is more beneficial than correcting errors in the enhancement layer. In future work, we plan to identify the layer to which each error belongs. We can then correct base-layer errors preferentially, and achieve further quality improvement.

References

[1] J. Wang, R. Sinnarajaj, T. Chen, Y. Wei, E. Tiedeman, and QUALCOMM. Broadcast and multicast services in cdma2000. *IEEE Communications Magazine*, 45(2):76–83, February 2004.

[2] 3GPP2. cdma2000 high rate broadcast-multicast packet data air interface specification. Standard C.S0054 v1.0, 3GPP2, 2004.

[3] P. Agashe, R. Rezaiifar, P. Bender, and QUALCOMM. cdma2000 high rate broadcast packet data air interface design. *IEEE Communications Magazine*, 42(2):83–89, February 2004.

[4] ISO/IEC. Information technology - generic coding of moving pictures and associated audio information. Standard ISO/IEC 13818, International Standards Organization, 2000.

[5] ISO/IEC. Information technology - coding of audio-visual objects. Standard ISO/IEC 14496, International Standards Organization, 2004.

[6] M. Zorzi, R. R. Rao, and L. B. Milstein. Error statistics in data transmission over fading channels. *IEEE Transactions on Communications*, 46(11):1468–1477, November 2004.

[7] I. Lee, Y. Choi, Y. Cho, Y. Joo, H. Lim, H. G. Lee, H. Shim, and N. Chang. On the statistics of block errors in bursty channels. *IEEE Design and Test of Computers*, 21(6):572–586, November-December 2004.

[8] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley, 1992.

Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06)
1550-445X/06 $20.00 © 2006 IEEE