Efficient Error Control for Scalable Media Transmission over 3G Broadcast Networks

Kyungtae Kang, Joonho Lee, Yongwoo Cho and Heonshik Shin

School of Computer Science and Engineering
Seoul National University, Seoul, 151-744, Korea
E-mail: {ktkang, jlee108, xtg05, shinhs}@cslab.snu.ac.kr

Abstract. Broadcast and mobile phone technologies have now combined to provide wireless multimedia services. 3GPP2 has introduced the Broadcast and Multicast Services (BCMCS) architecture in a 3G wireless network. BCMCS are capable of supplying multimedia content, which requires successive frames to arrive within a specific time interval. We analyze the execution time of Reed-Solomon decoding, which is the MAC-layer forward error correction scheme used in cdma2000 1xEV-DO BCMCS, under different air channel conditions. The results show that the time constraints of MPEG-4 cannot be guaranteed by Reed-Solomon decoding when the packet loss rate (PLR) is high, due to its long computation time on current hardware. To alleviate this problem, we propose three error control schemes. Our static scheme bypasses Reed-Solomon decoding at the mobile node to satisfy the MPEG-4 time constraint when the PLR exceeds a given boundary. Our second, dynamic scheme corrects errors in a best-effort manner within the time constraint, instead of giving up altogether when the PLR is high. The third, video-aware dynamic scheme fixes errors in a similar way to the dynamic scheme, but in a priority-driven manner which improves the quality of the final video. Extensive simulation results show the effectiveness of our schemes compared to the original FEC scheme.

Keywords: cdma2000 1xEV-DO, broadcast, error control schemes, Reed-Solomon, MPEG-4 FGS.

1 Introduction

Broadcast services are now widespread over wireless networks, making it possible to watch television on handheld devices. These broadcast and multicast services (BCMCS) [1][2] are currently being standardized by various mobile wireless standards bodies, and commercial operations have just begun. It is highly probable that multimedia broadcasting will become a 'killer application', in the wireless network environment.

Wireless networks are much more prone to errors than wired networks. In order to reduce the packet loss rate (PLR), some form of error correction mechanism is necessary to remove and restore the original information. Forward error correction (FEC) has been widely suggested for video broadcast applications,

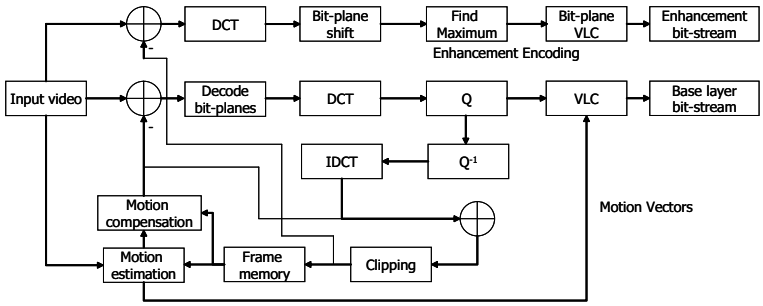


Fig. 1. FGS encoder block diagram.

due to the combination of strict delay requirements and the semi-reliable nature of video streams. In current BCMCS, Reed-Solomon (RS) coding is used for FEC [3][4]. However, RS coding is designed to fix errors accurately but not efficiently, and this causes several problems in BCMCS.

In our study, MPEG-4 [5] is the target multimedia application. To ensure an uninterrupted supply of images, the MPEG-4 standard specifies a strict timing constraint, but the RS decoder takes no account of this. We propose three solutions to this problem, realized as static, dynamic and video-aware dynamic error control schemes (ECS). All of the schemes bypass RS decoding if there is not enough time to correct the errors.

2 Multimedia in BCMCS

2.1 MPEG-4 FGS overview

To cope with the growing requirement for flexible stream transport, an MPEG working group has introduced a scalable video coding scheme called fine granularity scalability (FGS). The FGS video coding scheme in MPEG-4 not only provides an effective method of video compression, but also adapts its bit-rate flexibly to changing network conditions.

The FGS scheme encodes video frames into two layers, distinguished by the priority of the information that they contain, using bit-plane coding of discrete cosine transform (DCT) coefficients [7]. The layer containing information that is essential to the decodability of the whole video stream is called the base layer. The second layer is called the enhancement layer and includes more detailed information for improving the quality of the decoded video. The enhancement layer is obtained by bit-plane DCT coefficient coding of the differences between the original picture and the picture degraded by image transformations, as shown in Fig. 1. Because bit-plane coding considers each quantized DCT coefficient as a binary rather than a decimal number, each bit-plane of the enhancement data stream has its own level of significance, from the MSB (most significant bit) to the LSB (least significant bit). This makes it simple to truncate the enhancement

layer to a reduced number of bits. After receiving the whole of the base layer, the quality of the video increases proportionally as more of the enhancement layer is received.

However, this arrangement leaves the base-layer stream very sensitive to channel errors and, if the decoder finds any errors in the base layer, the enhancement layer of the current frame is discarded, whether it is correct or not. Should they go undetected, errors in the base layer will propagate to the start of the next group of pictures and cause serious drifting problems in the enhancement layers of subsequent frames. The enhancement layer is more tolerant, as we have seen, and in any case errors in that layer cannot degrade the video quality below the lower bound provided by the base layer.

To utilize this layered approach effectively, the more important base layer should be provided with more protection against errors than the less important enhancement layer. But the error recovery scheme in the current BCMCS standard makes no allowance for the different value of the two layers, and each is equally likely to be damaged during transmission.

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

An MPEG-4 system should have a maximum delay of 100ms from input to system decoder [5]. The corruption of this timing constraint will sequentially corrupt all the timing references and cause 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 allow itself to be re-initialized.

Guidelines for managing timing constraints and buffers in an MPEG-4 transport stream (TS) decoding process, and the rules that must be followed in order to satisfy the timing reference, are described elsewhere [9].

3 BCMCS error recovery technique

3.1 Reed-Solomon coding in BCMCS

The ECB structure is designed to allow efficient recovery from bursts of errors, by the way that such bursts are interleaved spatially within the ECB. Let M be the number of MAC packets in each row of the ECB (see Fig. 2). As M increases, the time-diversity also increases and thus a mobile node which is in a time-varying shadow environment is still able to recover a substantial amount of corrupted data. The organization of BCMCS is such that the value of M for a given ECB has to be less than or equal to 16.

In the context of MPEG-4 FGS video, the contents of an ECB are as shown in Fig. 2. Assuming the data rate of the base and enhancement layers of the video flow are b_B and b_E respectively, the number of packets allocated to each layer will be in the ratio $b_B : b_E$. The base-layer and enhancement-layer packets are segregated within the ECB in such a manner that all the MAC packets

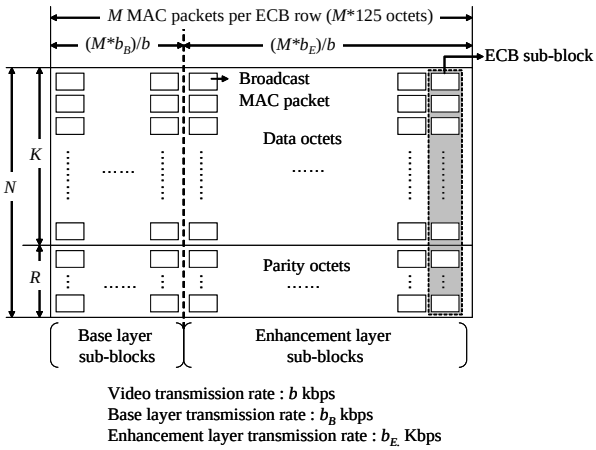


Fig. 2. Logical structure of an ECB.

in each ECB sub-block correspond to the same layer. If the values of $(M \times b_B)/b$ and $(M \times b_E)/b$ (see Fig. 2 again) are both integers and are less than the value of M, then all sub-blocks can be allocated to either the base-layer or the enhancement-layer section without any overlap. This segregation helps to prioritize the recovery of base-layer packets when only limited recovery is possible due to the timing constraints of MPEG-4.

3.2 Analysis of the current Reed-Solomon decoding technique

One of the most widely used chipset solutions that is able to support a cdma2000 1x system is the QUALCOMM MSM5000 Mobile Station $Modem^{\mathrm{TM}}$ (MSM), which uses the ARM7TDMI core as its microprocessor. We therefore used an ARM7TDMI testbed.

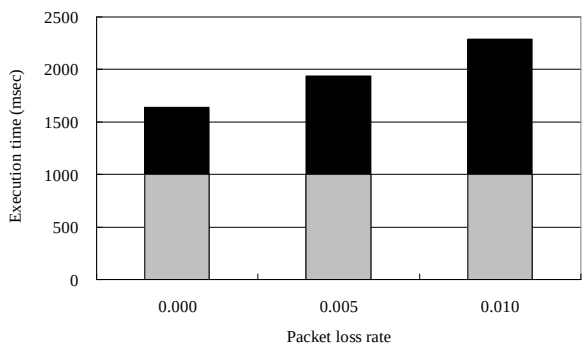
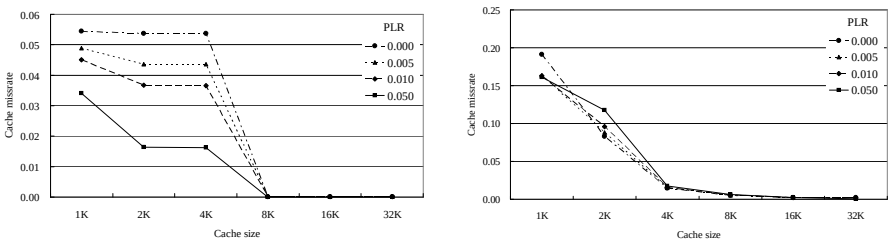


Fig. 3. Execution time of RS decoding on a cacheless system.



(a) I-cache (b) D-cache

Fig. 4. Cache miss rate against cache size.

The decoding times for different values of the packet loss rate (PLR) were measured without a cache. We found that values of PLR at the physical layer have a linear relationship with execution time, as shown in Fig. 3. In this figure, the gray bars show the timing deadline for processing the application layer multimedia service, and the black bars indicate the excess time required for RS decoding. ¿From this measurement, we see that RS decoding without a cache exceeds the time constraint of an MPEG-4 system, and the issue is therefore reduced to determining the best cache size.

We measured the miss rates of both the instruction and data cache while varying their size between 2k and 32k. The results shown in Figs. 4(a) and 4(b) 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 stabilizes if both the instruction and data cache are larger than 16k.

These two measurements suggest that, as the caches become bigger, the time required for RS decoding is reduced. We conducted an experiment to measure the execution time of RS decoding with instruction and data cache both sized at 16k, because the cache miss ratio has stabilized when both caches are this big. The results, shown in Fig. 5, indicate that PLR and execution time have

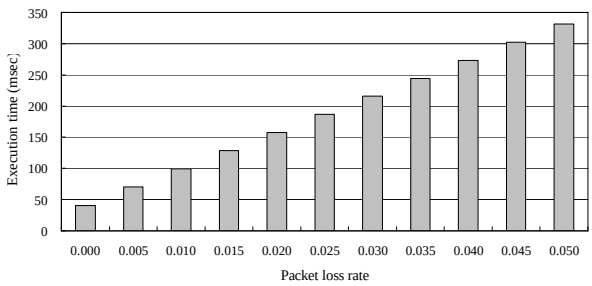


Fig. 5. RS execution time against PLR.

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

(a) Static ECS.

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

(b) Dynamic ECS.

Fig. 6. Pseudocode of the static and dynamic ECS.

a linear relationship, which we can use to decide whether RS decoding can be performed within the MPEG-4 time constraint. ¿From this graph, we can see that RS decoding can be completed on time up to a PLR of 1%, at which point decoding takes a little less than 100ms.

4 Proposed error recovery scheme

4.1 Static ECS

Timing is the main constraint on an MPEG-4 system, especially the 100ms ceiling on the PCR interval. Thus, we begin by proposing a static error control system based on the idea of anticipating whether RS decoding can be completed before this deadline, using the linear relation between PLR and execution time in a mobile node. This scheme omits RS decoding altogether if the system cannot meet the MPEG-4 timing constraint. Its pseudocode is shown in Fig. 6(a). Let P_{max} be the maximum PLR for which RS decoding can be completed in time. By the time that an ECB block arrives at the RS decoding layer, a mobile node will also have received the PLR of the physical layer. We will call the packet loss rate of data arriving at a mobile node through the physical layer the *given PLR*; and we will call the PLR after decoding the *recovered PLR*. We will denote the given PLR by P and, if it is greater than P_{max}, the node will omit RS decoding. Otherwise, it will perform RS decoding and correct the errors.

A shortcoming of this static scheme is that, when $P > P_{max}$, no use is made of the remainder of the PCB interval. The static ECS causes all the data to bypass RS decoding if the given PLR is greater than P_{max}, even though some of the errors might be fixed within the allowable time interval, and the MPEG-4 time constraint satisfied.

4.2 Dynamic ECS

Our dynamic error control system does not cut off at P_{max}. Instead, it will correct as many errors within the time constraint as it predicts as it can, based on the relationship between execution time and the number of errors. We denote the MPEG-4 time constraint by T_{max}, and the time remaining by T, as shown in

Fig. 6(b). 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 compare 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 the last sub-block, or insufficient time remains to fix even a single error.

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 time for RS decoding. These results, set out in Fig. 7, show that the execution time and the number of errors are linearly proportional, making $f(n)$ a linear function.

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

4.3 Video-aware dynamic ECS

Our video-aware dynamic error control system uses same technique as the dynamic one, except that it differentiates between sub-blocks in the base and enhancement layer. In MPEG-4 FGS, there is a fixed ratio between the number of base and enhancement sub-blocks, but we know that base-layer sub-blocks are more important. In this third scheme, base-layer sub-blocks are corrected first, followed by enhancement-layer sub-blocks if time remains.

The pseudocode for this ECS is shown in Fig. 8. We will denote the sub-block to be corrected by F, with the rest of the notation following that for the dynamic ECS. Initially, T is set to T_{max}, as in the dynamic scheme. The algorithm runs while any sub-blocks remain and there is also time in hand. It looks first for a base-layer sub-block, using the routine $find\_base\_sub\_block()$. If there are any uncorrected base sub-blocks, these will be corrected first; otherwise (i.e. $find\_base\_sub\_block()$ returns $NULL$), the algorithm will begin to correct enhancement-layer sub-blocks. In this scheme the parameter of the function

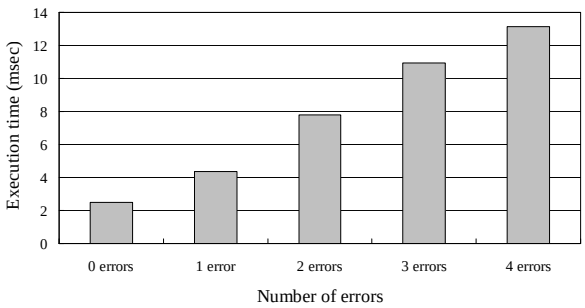


Fig. 7. Execution times for correcting a code word with different numbers of errors.

```
T_max = threshold_Time;
T = T_max
while (there_is_a_sub_block || T > f(0)) {
       F = find_base_sub_block();
       if (F == NULL)
              F = find_enhancement_sub_block();
       n = number_of_errors;
       if (f(n) < T) {
              do_RS_decoding(F);
              T = T - f(n);
       }
}
```

Fig. 8. Pseudocode of the video-aware dynamic ECS.

do_RS_decoding() is a sub-block to be corrected; this modification is necessary to ensure that the correct sub-block is selected.

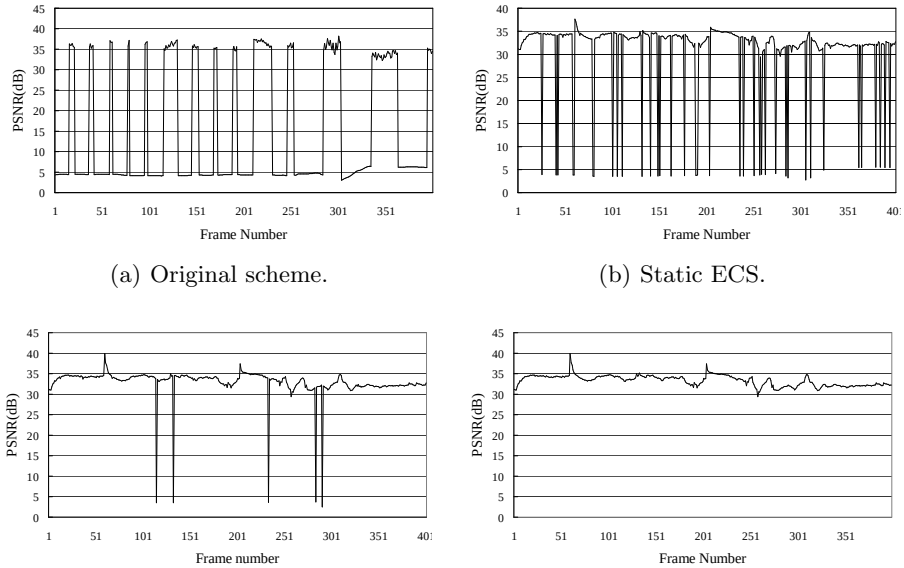
5 Performance evaluation

5.1 Experimental environment

We will first see how many errors can be corrected within the MPEG-4 time constraint. To do this 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 [11][12] to simulate the behavior of data errors which arise in transmission over fading channels. Fading in the air channel is assumed to have a Rayleigh distribution.

By choosing different values for the physical-layer packet loss rate and for $f_d N_{BL} T$ (which is the Doppler frequency normalized to the data-rate with block size N_{BL}, where f_d is the Doppler frequency, equal to the mobile velocity divided by the carrier wavelength), we can model different degrees of correlation in the fading process of radio channels. The value of $f_d N_{BL} T$ determines the correlation properties, which are related to the mobile speed for a given carrier frequency. When $f_d N_{BL} T$ 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 $f_d N_{BL} T$ (fast fading). In these experiments, we set the value of $f_d N_{BL} T$ to 0.02, which correspond to moderate fading.

The performance of Reed-Solomon coding was measured using the SEE (SNU Energy Explorer) [13]. 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 Foreman video sequence encoded with a bit-rate of 220kbps, running at 30 frames per second. The stream is handled by our reference MPEG-4 FGS codec, which is derived from the framework of the European ACTS Project



(a) Original scheme.

(b) Static ECS.

(c) Dynamic ECS.

(d) Video-aware dynamic ECS.

Fig. 9. PSNR of original and proposed schemes (given PLR = 5%).

Mobile Multimedia Systems (MoMuSys) [14], but modified for our experiments. The video streams consists of a base layer with a bit-rate of 110 kbps, and an enhancement layer with a bit-rate of 110 kbps. Physical packets were modulated using QPSK and transmitted at 1228.8kbps. Thus, the amount of data, N_{BL}, that can be fitted into one time slot is 256 bytes.

5.2 Experimental results

We conducted a simulation to measure the PSNR [15] of the original Reed-Solomon scheme, which is shown in Fig. 9(a). We found the average PSNR of the original scheme to be 16.01dB. Without any ECS, the decoder performs RS error recovery for every block. This causes so much computation that the PCB time interval is eventually exceeded.

We notice from Fig. 9(a) that, once the PSNR drops, it stays at a low level for a while. This is caused by a system reset. If the arrival of the PCR is delayed, the SCF will be damaged. This leads successively to damage to the DTS and PTS and eventually causes the entire system to be re-initialized, which results in degradation of the video quality, as explained elsewhere [9].

We have measured the PSNR achieved by the three proposed schemes. Fig. 9(b) shows the results achieved by the static ECS. Compared to the original scheme, it is noticeable that, once the PSNR has dropped to around 3dB, it quickly recovers. This is because the static ECS skips RS decoding if it cannot

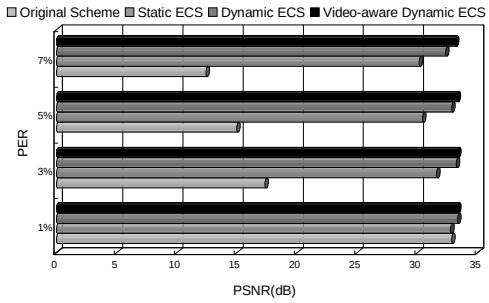


Fig. 10. Comparison of PSNR of original and proposed schemes.

meet the MPEG-4 time interval, which prevents re-initialization and results in instant recovery. The corrected average PSNR is 30.42dB, which is 14.41dB higher than that of the original scheme.

Fig. 9(c) shows the results achieved by the dynamic ECS, which exhibit fewer fluctuations than those of the static ECS, because the former does as much RS decoding as it can in the time available. Reduced fluctuation leads to smoother video and a higher-quality experience for a user who is watching the screen of a mobile node. Dynamic ECS achieves an improved PSNR of 32.84dB.

Fig. 9(d) shows the results for video-aware dynamic ECS, which exhibit the fewest fluctuations, with an average PSNR of 33.29dB. Fig. 10 compares the average PSNR of the original and the three new schemes. It shows that, when the given PLR is 1%, there is not much difference between the schemes: if there are few errors, all of them can usually be corrected within the MPEG-4 time interval, and so all the schemes have a similar PSNR. But, as the number of errors increases, the benefit of using one of our schemes becomes more noticeable. For all the error-rates that we tested, the relative ranking of our three error control schemes remains the same: best-effort recovery has better performance than the static scheme, but recovering higher-priority packets is best.

6 Conclusion

We have developed a realistic environment to measure the performance of the RS algorithm which is used for forward error correction in cdma2000 1xEV-DO BCMCS. We measured the execution times of an RS decoder when there are between 0 and 4 errors per sub-block, and the results of these tests suggest a guideline to control errors in an ECB so as to satisfy the MPEG-4 time interval.

If the delivery of a packet 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 ECS does this by skipping RS decoding for packets whose PLR 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. Our dynamic ECS takes the same approach, but also corrects as many errors as it can within the allowed time interval. Finally, the video-aware dynamic ECS corrects base-layer errors preferentially within the time interval, because correcting errors in the base layer of a video is more beneficial than correcting errors in the enhancement layer, and thus achieves a further quality improvement. Our simulation results show that these schemes greatly increase the PSNR compared to the original error correction scheme.

References

1. J. Wang, R. Sinnarajaj, T. Chen, Y. Wei, E. Tiedemann, and QUALCOMM, "Broadcast and multicast services in cdma2000," *IEEE Communications Magazine*, vol. 42, no. 2, pp. 76-82, February 2004.
2. 3GPP2, X.P0019 v 0.1.3, Broadcast-Multicast Services (BCMCS) Framework Draft Document, August 2003.
3. P. Agashe, R. Rezaiifar, P. Bender, and QUALCOMM, "cdma2000 high rate broadcast packet data air interface design," *IEEE Communications Magazine*, vol. 42, no. 2, pp. 83-89, February 2004.
4. 3GPP2, C.S0054 v1.0, cdma2000 High Rate Broadcast-Multicast Packet Data Air Interface Specification, February 2004.
5. ISO/IEC 14496-2, Coding of Audio-Visual Objects - Part2, May 2004.
6. W. Li, "Overview of fine granularity scalability in mpeg-4 video standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 301-317, March 2001.
7. W. Li, F. Ling, and H. Sun, "Bitplane coding of dct coefficients," ISO/IEC JTC1/SC29/WG11, MPEG97/M2691, October 1997.
8. W. Li and Y. Chen, "Experiment result on fine granularity scalability," ISO/IEC JTC1/SC29/WG11, MPEG99/M4792, March 1999.
9. Y. Cho, K. Kang, and H. Shin, "Proactive Reed-Solomon Bypass (PRSB): A Technique for Real-Time Multimedia Processing in 3G Cellular Broadcast Networks," *Proc. 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, vol. 4, pp. 532-538, September 2005.
10. R. Parry, "cdma2000 1xEV-DO [for 3G communications]," *IEEE Potentials*, vol. 21, no. 4, pp. 10-13, October-November 2002.
11. M. Zorzi and R. R. Rao, "On the statistics of block errors in bursty channels," *IEEE Transactions on Communications*, vol. 45, pp. 660-667, June 1997.
12. M. Zorzi, R. R. Rao, and L. B. Milstein, "Error statistics in data transmission over fading channels," *IEEE Transactions on Communications*, vol. 46, no. 11, pp. 1468-1477, November 1998.
13. I. Lee, Y. Choi, Y. Cho, Y. Joo, H. Lim, H. G. Lee, H. Shim, and N. Chang, "Web-based energy exploration tool for embedded systems," *IEEE Design and Test of Computers*, Vol. 21, Issue. 6, pp. 572-586, November-December, 2004.
14. A. Pearmain, A. Carvalho, A. Hamosfakidis, and J. Cosmas, "The momusys mpeg-4 mobile multimedia terminal," *Proc. 3rd ACTS Mobile Summit Conference*, pp. 224-229, June 1998.
15. R. C. Gonzalez, and R. E. Woods, *Digital Image Processing*, Addison-Wesley, 1992.