TOWARD ENERGY-EFFICIENT ERROR CONTROL IN 3G BROADCAST VIDEO

KYUNGTAE KANG, HANYANG UNIVERSITY
JUNHEE RYU AND SUNGHYUN CHOI, SEOUL NATIONAL UNIVERSITY
KYUNG-JOON PARK, DAEGU GYEONGBUK INSTITUTE OF SCIENCE AND TECHNOLOGY
JUNBEOM HUR, CHUNG-ANG UNIVERSITY

ABSTRACT

Energy use is a key issue in battery-operated mobile devices. In order to extend battery life, mobiles showing video often allow a controlled drop in quality, which is tolerable when the shortcomings of a small screen are combined with the imperfections of visual perception. Both energy consumption and video quality are affected by the Reed-Solomon codes and interleaving levels used in 3G video broadcast services. We explore the effect of these elements of MAC-layer error control, and show how they can be manipulated to save energy while maintaining acceptable video quality through a controlled reduction in the number of parity symbols in the Reed-Solomon code and the level of interleaving.

INTRODUCTION

Recent advances in computing and communication have led to a vast increase in the use of applications running on mobile devices, including smart phones, portable MP3 players, and email pagers, as well as laptops, tablet PCs, and palm computers. In these battery-operated devices, energy is a critical resource, and users are often forced to trade functionality for battery life. The nature of this trade-off depends on the application. In the case of video broadcast services, a user may well prefer to see the whole of a broadcast at a lower quality, rather than experience the sudden termination of high-quality video because of a flat battery.

Various attempts [1] have been made to reduce the energy consumption of devices and thus extend battery life while maintaining an adequate performance. However, there has not been much work on the problem of reducing the energy consumed by mobile devices in providing video broadcast services. This motivates us to develop an energy management scheme specifically for mobile devices providing video broadcast services, which can guarantee a certain performance while extending the battery life. In particular, we are concerned with CDMA2000 broadcast and multicast services (BCMCS), for which the air interface specification [2] was recently baselined by the Third Generation Partnership Project 2 (3GPP2).

The broadcast medium access control (MAC) protocol in BCMCS specifies the procedures used for transmitting and receiving data over the broadcast channel, and also the method of forward error correction (FEC) which is used to correct the errors that unavoidably occur on a radio link; the error correction processes employ Reed-Solomon (RS) coding, which effectively reduces the error rate seen by the higher layers [2]. Both RS encoding at the transmitter and decoding at the receiver involve an error control block (ECB), which is a buffer in the form of a two-dimensional array.

During encoding, data is put into this array row by row, and then RS coding is applied to the columns; decoding is the reverse process. The ECB structure provides block interleaving [3] in which sequential data is rearranged in a way that counters the bursts of errors that frequently occur in radio communication with mobile devices. The height of the ECB is determined by the RS code in use, and its width determines the extent of this interleaving: a wider ECB increases time diversity, which improves performance in the presence of time-varying fading.

Within the context of the BCMS MAC protocol, Kang [4] analyzed the error recovery performance of RS coding and block interleaving for transmissions over a fading channel. The amount of data in a video, which all has to undergo RS decoding, makes the operation of the broadcast MAC layer a significant drain on the battery of a mobile device. Kang *et al.* [5] proposed a new energy model that describes how the energy consumed by RS decoding varies with the condition of a Rayleigh fading channel when the level of ECB interleaving is the maximum allowed by the CDMA2000 BCMCS standard. They also measured the actual energy consumption of the RS

 1536-1284/12/$25.00

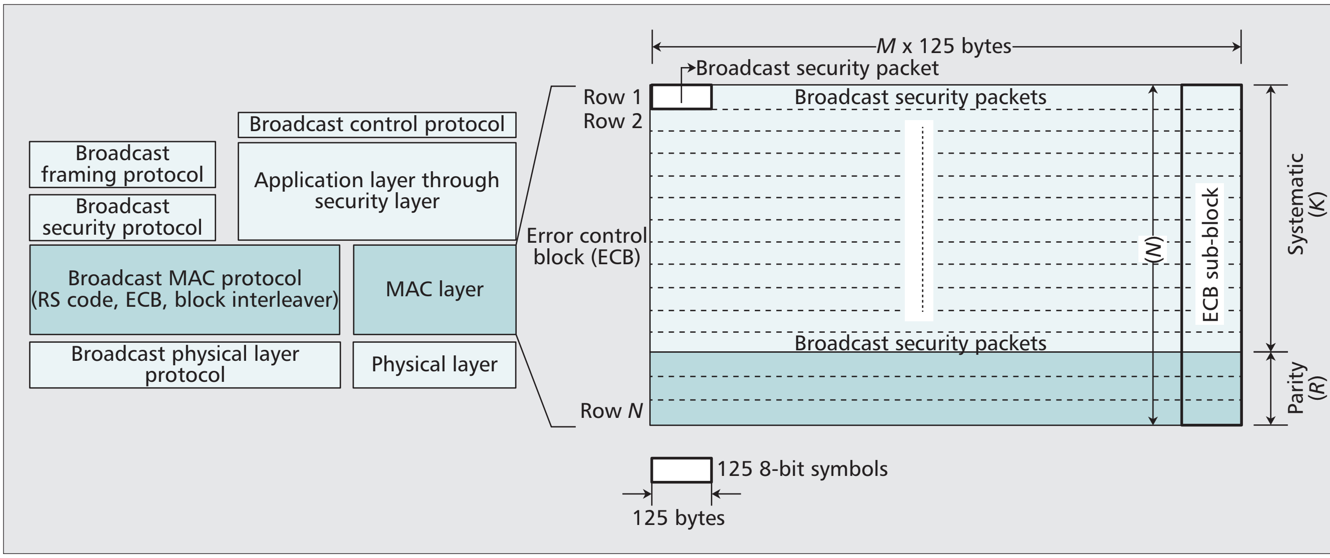


Figure 1. *Broadcast protocol suite and its broadcast MAC protocol components.*

decoding process running on a mobile device based on an ARM7TDMI core and a Samsung 128-Mb SDRAM array (K4S280832B), operating at a clock speed of 100 MHz, using the Seoul National University Energy Explorer (SEE) [6]. With the same equipment, Kang *et al.* [7] extended their energy model to determine the energy consumption of RS decoding under various levels of ECB interleaving, and they also investigated the detailed effect of interleaving on RS error correction.

These studies have provided a good understanding of the trade-off between performance and the energy used by the RS coding-based error recovery in BCMCS: the bottom line is that quality of service (QoS) can be improved by using an RS code with more parity or a higher level of block interleaving, at the cost of increased energy consumption. Consequently, error control in mobile devices needs to balance QoS against energy consumption in order to meet the user's preferences.

In this article, we report on a further extension to this line of work, and show how to determine the most energy-efficient operating range for a particular RS code and QoS requirement. We also propose a way of selecting an energy-efficient level of block interleaving by the use of a relative energy efficiency metric that takes account of the channel condition. This approach has been shown to reduce the overall energy consumption of mobile devices running a video application, while incurring only a minor reduction in the performance of error recovery. This is a significant improvement over the employment of the widest allowable ECB, which maximizes the amount of interleaving, without reference to the channel conditions and user preferences with regard to energy use and service quality.

The remainder of this article is organized as follows. We introduce 3GPP2 BCMCS. We characterize the use of energy by the RS decoding process and present energy values measured with the SEE. We apply this information to the investigation of the effect of user mobility and the level of interleaving on energy consumption. The results of this investigation allow us to introduce two methods of energy-efficient link control. Finally, we draw conclusions.

Introduction to 3GPP2 BCMCS

BCMCS Protocols

BCMCS provide point-to-multipoint transmission of multimedia data such as text, audio, still pictures, and video from a single source to a group of users in a specific area. BCMCS are built on unicast code-division multiple access (CDMA) high-rate packet data networks. The air interface [2] used by broadcast packet data within a CDMA2000 1xEV-DO system is made up of the group of protocols shown in Fig. 1, which is called the broadcast protocol suite.

The broadcast framing protocol specifies the way in which higher-layer packets are fragmented at the access network and, in part, how the access terminal determines higher-layer packet boundaries. It also provides a methodology for validating the integrity of higher-layer packets, so packets that have a high probability of containing erroneous data are discarded. The broadcast security protocol specifies how framing packets are encrypted. The broadcast MAC protocol defines the procedures used to transmit over the broadcast channel, which includes FEC and multiplexing to reduce the radio link error rate seen by the higher layers. The broadcast physical layer provides the channel structure for the broadcast channel. The broadcast control protocol defines procedures that control various aspects of the operation of the broadcast packet data system. Our primary concern in this study is the broadcast MAC protocol, which we now describe in detail.

MAC-Layer Error Control in BCMCS

The components of the broadcast MAC protocol are shown in Fig. 1. Unlike the equivalent protocol in the unicast CDMA2000-1xEV-DO standard, the broadcast MAC protocol does not specify automatic repeat requests (ARQs) for error control because a reverse link is not available to transport acknowledgment and negative acknowledgment (ACK/NAK) signals. Instead, FEC is specified.

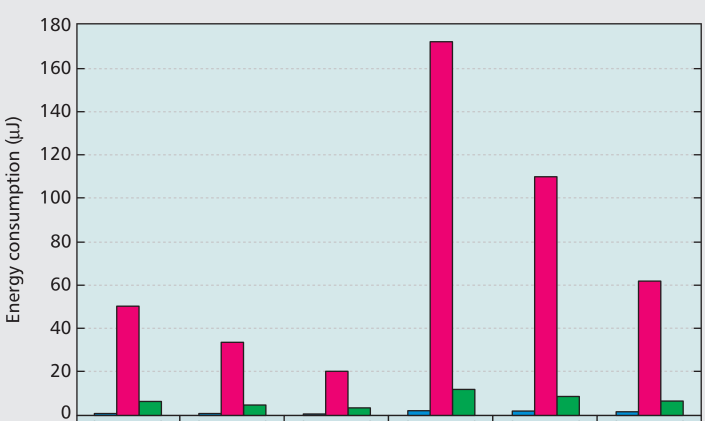
The broadcast MAC protocol receives packets from the security protocol and adds an outer FEC code that, in conjunction with the physical-layer turbo code, forms a product code. With a short codeword, the performance of turbo codes is only marginally better than that of convolutional codes, and the decoding of turbo codes is more complicated. Moreover, it is clear from the product-code literature that RS codes are a better choice for the outer FEC code due to their superior performance at lower error rates [4]. Hence an RS block code is specified as the outer code.

An RS code is specified by a tuple (N, K, R), where K and R ($= N - K$) are respectively the number of information symbols and parity symbols. The values of K can be 28, 26, or 24 for $N = 32$ (alternatively $K = 14$, 13, or 12 for $N = 16$), providing RS codes of rate 7/8, 13/16, and 3/4, respectively. In BCMCS, an RS code is capable of recovering up to R erased symbols in each codeword because it is used as an erasure code [2]. In erasure decoding, the positions of the erased symbols are known in advance so that the entire capability of the code can be dedicated to recovering their values. Provided that the locations of the erasures are known accurately, and that other errors outside the erasures do not occur, erasure decoding outperforms error correction decoding, in which the positions of the errors are unknown. A cyclic redundancy check (CRC) is used to detect damaged packets in the physical layer, and the physical-layer protocol erases these packets and informs the RS erasure decoder of their locations.

For each logical channel, the access network forms an error control block (ECB) with the two-dimensional array structure introduced earlier. Each ECB consists of N rows and M columns, and each element of the array is a MAC packet of 125 bytes, as shown in Fig. 1. The broadcast security packets are filled into an ECB in rows. The access network then applies RS encoding along the columns of the ECB and transmits it row by row on the broadcast channel. Multiple ECBs are multiplexed before transmission to the physical layer. The width of the ECB is $125M$ bytes, and as mentioned earlier, this width sets the level of interleaving because it determines the temporal spacing between the symbols in each codeword during transmission. The resulting time diversity increases with M, and consequently a mobile device in a time-varying fading environment is still able to recover a substantial amount of corrupted data. The BCMCS standard specifies that the value of M for a given error control block should be 16 or less.

RS code	Energy consumption (μJ)					
	e_1		e_2		e_3	
	CPU	MEM	CPU	MEM	CPU	MEM
(16,12,4)	0.558	0.418	32.115	18.157	4.320	2.404
(16,13,3)	0.555	0.426	21.476	12.372	3.358	1.854
(16,14,2)	0.507	0.390	12.909	7.673	2.455	1.363
(32,24,8)	1.321	0.953	111.442	60.744	7.851	4.259
(32,26,6)	1.225	0.901	71.084	39.085	5.898	3.153
(32,28,4)	1.111	0.832	39.709	22.261	4.325	2.400

CPU: central processing unit, MEM: main memory



	(16,12,4)	(16,13,3)	(16,14,2)	(32,24,8)	(32,26,6)	(32,28,4)
e_1	0.9766	0.9812	0.8971	2.2746	2.1263	1.9424
e_2	50.2721	33.8475	20.5815	172.1866	110.1693	61.9694
e_3	6.7243	5.2114	3.8181	12.1098	9.0505	6.7251

Table 1. *Average energy consumption by each computational component per codeword.*

Energy Characterization of the Software RS Decoding Process

By virtue of recent improvements in algorithms and processors, software implementation of the RS decoding process can now run at relatively high data-rates, thus avoiding the inflexibility of special hardware. We will now analyze the operation of a software RS erasure decoder in some detail.

When an RS codeword is received, the decoder starts by checking for the existence of erasures. If there are erasures, and their number is within the capability of the RS code in use, then the decoder will attempt to correct them. The erased data can then be restored by an iterative erasure decoding process. The number of times that this last procedure needs to be executed is equal to the number of erasures in the received codeword, and hence the energy required for RS decoding increases with the error rate at the input to the decoder.

This summary of the RS erasure decoding process suggests that it can be logically separated into three computational components. The first component decides whether to try to correct the erasures in a codeword or not. The second component computes the syndromes and constructs erasure location polynomials that will be required in the correction loop. Finally, the third component runs in that loop and successively corrects all the erasures in the codeword. The first component is executed for every codeword received; the second component is called once for each codeword that contains any erasures; and the third component must be invoked for each erasure in a codeword. Thus, the total energy e_{cw} consumed in decoding an RS codeword can readily be predicted to be the sum of the energy each component is expected to use in dealing with the number of erasures, ν, contained in the codeword, as follows:

$$e_{cw}(v) = \begin{cases} e_1 & \text{if } v = 0 \text{ or } v > R) \\ e_1 + e_2 + ve_3 & (\text{if } 0 < v \leq R), \end{cases}$$

where e_i is the energy used by the ith component in decoding a codeword with a single erased symbol.

We measured the average energy consumption of the reference software implementation of the RS decoder provided by Henry Minsky, which is originally designed to correct both erasures and errors (the source code is available at http://rscode.sourceforge.net). Because an RS code is used as an erasure code in BCMCS, we modified the source code by removing the functions to construct the error location polynomial and also to find the roots of the error location polynomial for erasure-only decoding. The energy profiling was done by using the XEEMU simulator [8], which provides accurate power data for the ADI 80200EVB XScale board. This board uses the ARMv5TE instruction set, which is identical to that of the modem processors in some of the QUALCOMM chips found in smart phones. We assume that the core of our target processor operates at 400 MHz, has separate instruction and data caches, both of 32 kbytes, and a Micron 128-Mbyte SDRAM with a clock speed of 100 MHz. Our measurements of the energy required by each computational component to decode a codeword containing a single erased symbol (i.e., octet) are presented in Table 1.

We can observe three clear trends in Table 1. First, as the number of parity symbols is increased to improve the error correcting capability of the RS code, more energy is used in erasure correction. Second, the energy required by each component of the RS algorithm to decode a codeword increases with the number of parity symbols. Third, the second component consumes by far the most energy, and hence there is a huge difference in the energy needed to deal with a codeword with no erasures and one that contains a single erasure. Further erasures have a relatively small effect.

On the Effect of Mobility and Interleaving on Energy Consumption

Error Statistics in Mobile Environments

A lot of communication theory is based on a particular idealization of the communication channel, which is seen as limited in bandwidth and corrupted by additive white Gaussian noise (AWGN). However, in a mobile radio environment, we also have to pay considerable attention to fluctuations in the amplitude and phase of received signals due to multipath effects. The transmission environment can vary significantly as the channel condition changes over time. Variations in path loss are not only caused by changes in the distance between the transmitter and the receiver, but also by shadowing and multipath fading. These effects largely determine the error processes occurring in the wireless channel.

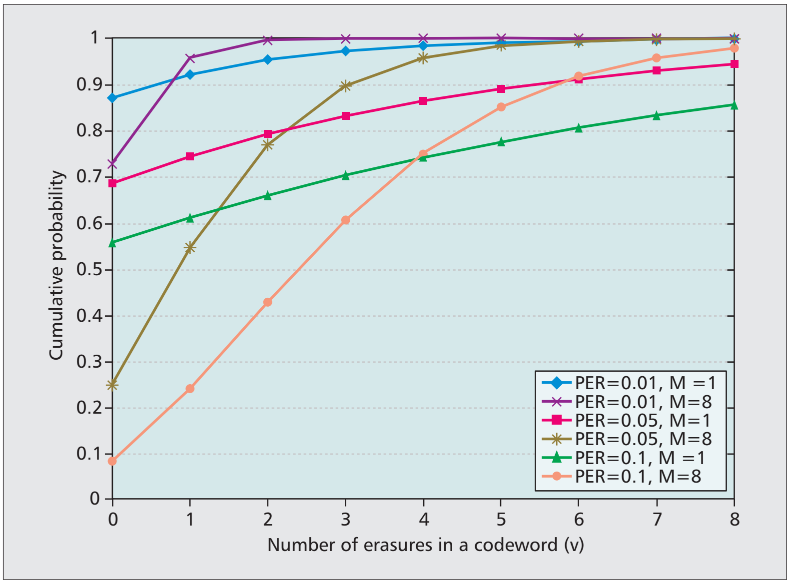


Figure 2. *Cumulative probability distributions of the number of erasures for a codeword of a mobile moving at 3.5 km/h.*

Several recent efforts to collect trace data for a variety of wireless networks, including 3G networks, have improved our understanding of error processes in wireless channels. In particular, we know that errors tend to occur in bursts in a fading channel, as opposed to the random patterns associated with an AWGN channel. A fading channel has a memory, making it natural to approximate it by a Markov model. Wang [9] demonstrated that such an approximation is satisfactory by investigating the accuracy of a first-order Markov process in modeling transmissions over a flat Rayleigh-fading channel.

When a receiver is moving, its velocity causes a Doppler shift in the frequency of the signal transmitted along each signal path. In a fading channel, the relationship between the velocity υ of a mobile receiver and the maximum Doppler frequency of the system, f_D, can be given as follows:

$$f_D = \frac{f_c \upsilon}{c},$$

where υ is the velocity of the receiver, c is the speed of the electromagnetic wave, and f_c is the carrier frequency.

Effect of Block Interleaving on Energy Consumption

Figure 2 shows the cumulative distribution function (CDF) for the number of erasures ν in a codeword, when $N = 32$, for a mobile receiver moving at a speed of 3.5 km/h with a reference channel data rate of 614.4 kb/s, a carrier frequency of 900 MHz, and a physical-layer packet length of 1024 bits. Focusing on the RS codes for which $N = 32$, we set the packet error rate (PER) of the forward data channel to 0.01, 0.05, or 0.1, and the value of M to 1 or 8.

As more block interleaving is achieved by increasing the value of M, bursts of erasures are spread more widely over multiple codewords, while the average density of erasures in each codeword during those bursts is reduced. Thus, the probability that a codeword contains at least one erasure is higher with the larger M value, e.g., $M = 8$ in Fig. 2. Conversely, when the value of M is small, some codewords contain a large number of erasures, because they correspond to error bursts, while many others contain no erasures. Hence, the probability that a codeword contains no erasures is higher, and the cumulative probability increases more slowly with the number of erasures, as shown in Fig. 2. Because the presence of any erasures in a codeword has more effect on energy consumption than their number, as already observed, the average energy used in RS decoding of a fixed amount of data increases with the value of M. This phenomenon is clearly shown in Fig. 3, which gives the average amount of energy required to decode a payload of one byte encoded with RS codes of (32, 24, 8) and (32, 26, 6) on an XScale-based mobile platform moving at 7 km/h, while varying the PER.

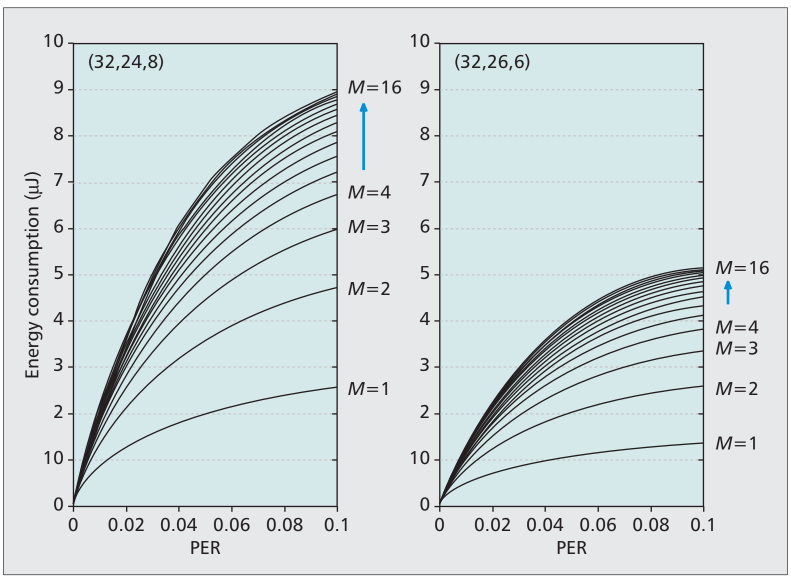


Figure 3. *Energy consumed in a mobile by RS decoding for varying values of the PER and of* M*, when the mobile is moving at 7 km/h.*

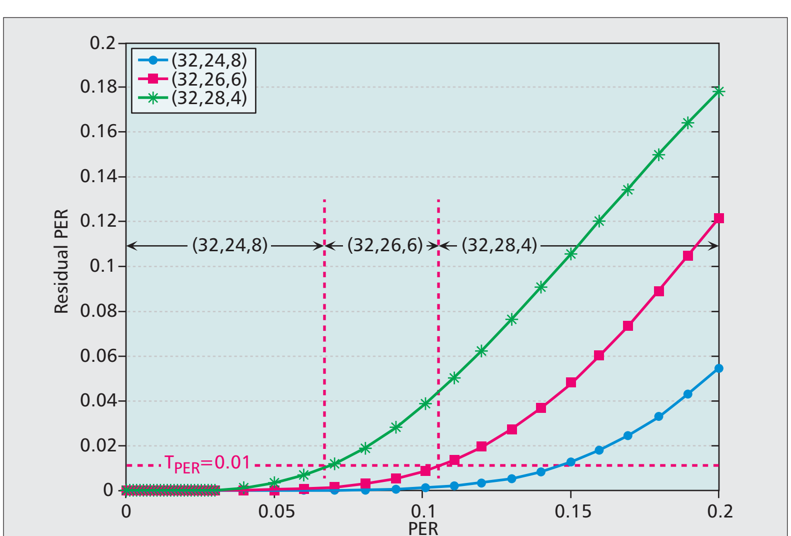


Figure 4. *Error recovery performance against channel PER.*

Towards an Energy-Efficient MAC Protocol

Selection of RS Codes for Energy Efficiency

Motivation — Table 1 and Fig. 3 show that more energy is needed for error correction if an RS code with more parity information is used, because the complexity of computing the syndromes and erasure evaluator polynomials increases with the amount of parity information. Our observation that RS codes with fewer parity symbols achieve a significant energy saving at the cost of error recovery performance suggests that the best choice of RS code for a video application is the one that saves the most energy while still guaranteeing the QoS requirement.

Energy-Efficient RS Code with Guaranteed QoS — Figure 4 shows the residual PER after RS decoding (i.e., the remaining PER at the output of RS decoding) when M is set to 16, and the PER of the forward data channel varies between 0 and 0.2. Here, the velocity of the mobile is around 5 km/h, the carrier frequency is 900 MHz, and the reference channel data rate is 614.4 kb/s, which is one of the forward-link data rates specified in CDMA2000. Figure 4 clearly shows that RS codes with more parity symbols have a greater error recovery capacity, as we would expect, but decoding them takes more energy, as shown in Fig. 3. Thus, if we use the RS code with the minimum amount of parity that still guarantees the required service quality instead of that with the maximum parity, we can save a significant amount of energy.

For example, the leftmost of the three intervals marked in Fig. 4 contains PERs for which a (32, 28, 4) code is the most energy-efficient, assuming the required bit-level QoS, denoted as T_{PER}, is 0.01; for PERs above 0.065 or so, a (32, 26, 6) code is the most efficient (middle interval); and when the channel condition is extremely bad and the PER of the channel exceeds 0.12, it is necessary to select the (32, 24, 8) code to reduce the residual PER as far as possible (rightmost interval).

There are many applications that do not require fastidious control of the PER. In such applications our scheme can save a significant amount of energy while guaranteeing the required QoS level for the service.

Selection of the Level of ECB Interleaving for Energy Efficiency

Motivation — We have already shown that there is a significant difference between the energy consumption of the RS decoder with no erasures and with at least one erasure in each codeword, because building the erasure evaluator polynomials used to correct erasures is a complicated process. This suggests that it is more energy-efficient to concentrate a burst of errors into a small number of codewords, rather than to disperse it more widely. Hence, by making the ECB as narrow as possible without incurring a significant reduction in error recovery performance, we can significantly reduce energy consumption by increasing the number of error-free codewords.

Energy-Efficient ECB Interleaving — The way in which energy consumption varies with the width of the ECB and the PER of the forward data channel suggests that energy can be saved if we choose the lowest level of interleaving that still provides an acceptable level of performance. To select this adequate level of interleaving, we could use a method similar to the one, described earlier.

An alternative approach involves an energy-efficiency metric [10] that expresses the ratio between error-recovery rate and energy consumption for a given interleaving level $M = M'$, normalized to maximum interleaving, i.e., $M = 16$. In these terms, the relative energy efficiency is defined as follows:

$$\xi_{M=M'} = \frac{\left(1 - E\left[\epsilon_r \middle| M = M'\right]\right)}{E\left[e_{\mathrm{cw}} \middle| M = M'\right]} \frac{E\left[e_{cw} \middle| M = 16\right]}{\left(1 - E\left[\epsilon_r \middle| M = 16\right]\right)}$$

where $E[\varepsilon_r | M = M']$ is the expected residual PER after RS decoding and $E[e_{\mathrm{cw}} | M = M']$ is the average energy used in decoding an RS codeword when the value of M is M'. Thus, $\xi_{M=M'}$ is a measure of the effectiveness with which the additional energy supplied reduces the residual PER (ε_r), as M changes from 16 to M'.

Figure 5 shows the relative energy efficiency of RS decoding at each value of M for PERs of 0.01, 0.03, and 0.05 with the RS code of (32, 24, 8) when a mobile is moving at 5 km/h and 50 km/h, with a reference channel data-rate of 614.4 kb/s and a carrier frequency of 900 MHz. In this example, for PERs of 0.01, 0.03, and 0.05, the relative energy efficiency of a mobile which moving at 5 km/h begins to saturate around 1 when M is 5, 7, and 8, respectively, after which there is hardly any further benefit in using more energy. At the faster speed of 50 km/h, the relative energy efficiency reaches the saturation level of 1 sooner: when M is respectively 2, 2, and 3, for the same PERs. This is because the channel errors are much less bursty at 50 km/h, so compensating for them requires less interleaving (i.e., lower values of M).

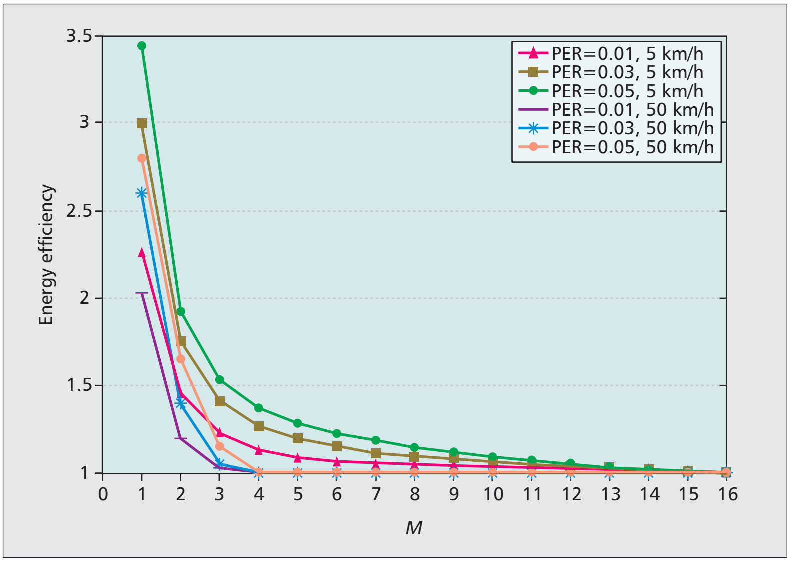


Figure 5. *Energy efficiency against* M *for channel PERs of 0.01, 0.03, and 0.05, and mobile speeds of 5 km/h and 50 km/h, using an RS code of (32, 24, 8).*

Similar selections can be made for a (32, 26, 6) code, but in this case the saturation point occurs at a higher level of interleaving, because the performance of error recovery is reduced when there are fewer parity symbols, and thus additional energy can be used effectively to improve performance at higher values of M. By selecting the level of interleaving in this manner, significantly less energy is used than with the maximum level of interleaving ($M = 16$), as shown in Table 2. Using the recommended value of M achieves a significant reduction in the energy consumed by RS decoding, with negligible effect on the outcome of RS error correction. The numbers in parentheses in Table 2 show the drops in RS error correction performance (the increases in residual PER) that occur when we use the recommended values of M. As the

RS code	Mobility	PER					
		0.01		0.03		0.05	
		M′	Energy saving	M′	Energy saving	M′	Energy saving
(32, 24, 8)	5 km/h	5	18.4% (0)	7	15.2% (2.04E-10)	8	12.8 (2.00E-10)
	50 km/h	2	8.3% (0)	2	6.9% (1.34E-10)	3	4.8 (1.18E-10)
(32, 26, 6)	5 km/h	6	15.1% (3.10E-11)	8	13.4% (7.34E-8)	10	9.3 (5.68E-8)
	50 km/h	2	7.2% (0)	3	5.8% (4.62E-8)	3	4.5 (4.11E-8)
(32, 28, 4)	5 km/h	8	11.8% (4.92E-7)	10	7.2% (4.68E-7)	12	6.5% (3.90E-7)
	50 km/h	3	5.7% (8.36E-7)	3	3.8% (1.09E-6)	4	2.6% (1.18E-6)

Table 2. *Recommended values of* M (M′) *for energy efficiency, and resulting energy savings and increases in residual per (the numbers in parentheses), achieved by recommended values of* M *compared to* M = *16.*

We suggest that the ECB is made as small as possible without incurring significant performance reduction, and we have shown that this reduces the overall energy consumption of mobile devices running a video application with only minor reduction in playback quality.

MPEG decoder has its own error resilience, such modest increases have little influence on the playback quality of MPEG videos.

Concluding Remarks

In this article, we have explored the energy consumption of mobile devices receiving high-data-rate broadcast services in a 3G cellular network, focusing on error recovery by RS decoding in the MAC layer. The energy consumption of the RS decoding process is mainly determined by three computational components: the first of these decides whether to try correcting any errors or not, and the other two build the syndromes and perform the erasure loops that make the required corrections.

We have found that the choice of RS code affects the energy consumption, which increases with the number of parity symbols in the code. Our analysis of the energy consumed by RS decoding suggests an energy-efficient operating range for each RS coding scheme, while guaranteeing the required bit-level QoS under varying channel conditions. Experimental results show that a significant amount of energy can be saved by selecting the RS code appropriate to the channel condition while maintaining the target QoS.

The RS error correction scheme uses a data interleaving mechanism to increase error recovery performance. This can be adjusted in BCMCS by changing the width of the ECB. A wider ECB allows more effective recovery from bursty errors, but also increases energy consumption, memory requirement, and service delay. We therefore suggest that the ECB is made as small as possible without incurring significant performance reduction, and we have shown that this reduces the overall energy consumption of mobile devices running a video application with only minor reduction in playback quality. This is a significant improvement over the use of an ECB of the maximum width without regard to the channel conditions.

Acknowledgments

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (2010-0022076). The corresponding author is J. Hur.

References

[1] G. Anastasi *et al.*, "A Power-Aware Multimedia Streaming Protocol for Mobile Users," *Proc. IEEE Int'l. Conf. Pervasive Services*, July 2005, pp. 371–80.
[2] CDMA2000 High Rate Broadcast-Multicast Packet Data Air Interface Specification, 3GPP2 Std. C.S0054 Rev. 1.0, Feb. 2004.
[3] K. Kang and L. Sha, "An Interleaving Structure for Guaranteed QoS in Real-Time Broadcasting Systems," *IEEE Trans. Computers*, vol. 59, no. 5, May 2010, pp. 666–78.
[4] K. Kang, "Probabilistic Analysis of Data Interleaving for Reed-Solomon Coding in BCMCS," *IEEE Trans. Wireless Commun.*, vol. 7, no. 10, Oct. 2008, pp. 3878–88.
[5] K. Kang, Y. Cho, and H. Shin, "Energy-Efficient MAC-Layer Error Recovery for Mobile Multimedia Applications in 3GPP2 BCMCS," *IEEE Trans. Broadcasting*, vol. 53, no. 1, Mar. 2007, pp. 338–49.
[6] I. Lee *et al.*, "Web-based Energy Exploration Tool for Embedded Systems," *IEEE Design and Test of Computers*, vol. 21, no. 6, Nov. 2004, pp. 572–86.
[7] K. Kang and H. Shin, "Modeling the Energy Consumption of Reed- Solomon Decoding with Interleaving on Fading Channels," *IEEE Trans. Wireless Commun.*, vol. 7, no. 11, Nov. 2008, pp. 4100–04.
[8] Z. Herczeg *et al.*, "Energy Simulation of Embedded XScale Systems with XEEMU," *J. Embedded Computing*, vol. 3, no. 3, 2009, pp. 209–19.
[9] H.S. Wang, "On Verifying the First-Order Markovian Assumption for A Rayleigh Fading Channel Model," *IEEE Trans. Vehic. Tech.*, vol. 45, no. 2, May 1996, pp. 353–57.
[10] K. Kang and H. Shin, "Reduced Data Rates for Energy-Efficient Reed- Solomon FEC on Fading Channels," *IEEE Trans. Vehic. Tech.*, vol. 58, no. 1, Jan. 2009, pp. 176–87.

Biographies

KYUNGTAE KANG [M] is an assistant professor in the Department of Computer Science and Engineering at Hanyang University, Korea. He received a B.S. degree in computer science and engineering, followed by M.S. and Ph.D. degrees in electrical engineering and computer science, from Seoul National University (SNU), Korea. Prior to his tenure at Hanyang University, he spent four years as a postdoctoral research associate at the University of Illinois at Urbana-Champaign (UIUC). His research interests lie primarily in systems, including operating systems, mobile systems, sensor systems, distributed systems, and real-time embedded systems. His most recent research interest is in the interdisciplinary area of cyber-physical systems. He is a member of the ACM.

JUNHEE RYU [S] received his B.S. degree in computer engineering from Korea Aerospace University in 2003, and his M.S. degree in computer science and engineering from SNU in 2005. He is currently working toward a Ph.D. degree in computer science and engineering at the same university. His research interests include file systems, storage systems, and network systems.

JUNBEOM HUR (jbhur@cau.ac.kr) received his B.S. degree in computer science from Korea University in 2001, and M.S. and Ph.D. degrees in computer science from the Korea Advanced Institute of Science and Technology (KAIST) in 2005 and 2009, respectively. He was at UIUC as a postdoctoral researcher from 2009 to 2011. He is currently an assistant professor in the School of Computer Science and Engineering at Chung-Ang University, Korea. His research interests include information security, mobile computing, and wireless network security.

KYUNG-JOON PARK [M] received his B.S. and M.S. degrees from the School of Electrical Engineering and Ph.D. degree from the School of Electrical Engineering and Computer Science, SNU. He is currently an assistant professor in the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, Korea. He was a postdoctoral research associate at the Department of Computer Science, UIUC from 2006 to 2010. He was with Samsung Electronics, Suwon, Korea, as a senior engineer from 2005 to 2006, and was a visiting graduate student in the Department of Electrical and Computer Engineering, UIUC in 2001 and 2002. His current research interests include modeling and analysis of cyber-physical systems and design of medical-grade protocols for wireless healthcare systems. He is currently serving on the editorial boards of *Wiley Transactions on Emerging Telecommunications Technologies*. He served as lead guest editor for a special issue on cyber-physical systems in *Computer Communications*. He has also served as a TPC member of numerous international conferences on wireless networking. He is a recipient of the Gold Prize in the Samsung InsideEdge Thesis Competition.

SUNGHYUN CHOI [SM] is a professor at the School of Electrical and Computer Engineering, SNU. Before joining SNU in September 2002, he was with Philips Research USA, Briarcliff Manor, New York, as a senior member of research staff and a project leader for three years. He was also a visiting associate professor at the Electrical Engineering Department, Stanford University, California, from June 2009 to June 2010. He received his B.S. (summa cum laude) and M.S. degrees in electrical engineering from KAIST in 1992 and 1994, respectively, and received his Ph.D. from the Department of Electrical Engineering and

Computer Science, University of Michigan, Ann Arbor in September 1999. His current research interests are in the area of wireless/mobile networks with emphasis on wireless LAN/MAN/PAN, next-generation mobile networks, and cross-layer approaches. He has authored/coauthored over 150 technical papers and book chapters in the areas of wireless/mobile networks and communications. He coauthored (with B. G. Lee) *Broadband Wireless Access and Local Networks: Mobile WiMAX and WiFi* (Artech House, 2008). He holds about 80 patents, and has tens of patents pending. He has served as General Co-Chair of COMSWARE 2008, a Program Committee Co-Chair of ACM Multimedia 2007, IEEE WoWMoM 2007, and COMSWARE 2007, and a Co-Chair of IEEE GLOBECOM 2011 Wireless Networking Symposium. He has also served on program and organization committees of numerous leading wireless and networking conferences including ACM MobiCom, IEEE INFOCOM, IEEE SECON, IEEE MASS, and IEEE WoWMoM. He is also currently serving on the editorial boards of *IEEE Transactions on Mobile Computing* and IEEE Wireless Communications, and served as an editor of *ACM SIGMOBILE Mobile Computing and Communications Review*, *Journal of Communications and Networks*, *Computer Networks*, and *Computer Communications*. He has served as a guest editor for *IEEE Journal on Selected Areas in Communications*, *IEEE Wireless Communications*, *Pervasive and Mobile Computing*, *ACM Wireless Networks*, *Wireless Personal Communications*, and *Wireless Communications and Mobile Computing*. From 2000 to 2007, he was an active contributor to the IEEE 802.11 WLAN Working Group. He has received a number of awards including the Young Scientist Award awarded by the President of Korea (2008); IEEK/IEEE Joint Award for Young IT Engineer (2007); Shinyang Scholarship Award (2011), the Outstanding Research Award (2008) and Best Teaching Award (2006) from the College of Engineering, SNUn; the Best Paper Award from IEEE WoWMoM 2008; and Recognition of Service Award (2005, 2007) from ACM. He was a recipient of the Korea Foundation for Advanced Studies (KFAS) Scholarship and the Korean Government Overseas Scholarship during 1997-1999 and 1994-1997, respectively. He is a member of ACM, KICS, IEEK, and KIISE.