Fall 1-31-2003

Robust and efficient video/image transmission

Xi Min Zhang
New Jersey Institute of Technology

ABSTRACT

ROBUST AND EFFICIENT VIDEO/IMAGE TRANSMISSION

by
Xi Min Zhang

The Internet has become a primary medium for information transmission. The unreliability of channel conditions, limited channel bandwidth and explosive growth of information transmission requests, however, hinder its further development. Hence, research on robust and efficient delivery of video/image content is demanding nowadays.

Three aspects of this task, error burst correction, efficient rate allocation and random error protection are investigated in this dissertation. A novel technique, called successive packing, is proposed for combating multi-dimensional (M-D) bursts of errors. A new concept of basis interleaving array is introduced. By combining different basis arrays, effective M-D interleaving can be realized. It has been shown that this algorithm can be implemented only once and yet optimal for a set of error bursts having different sizes for a given two-dimensional (2-D) array.

To adapt to variable channel conditions, a novel rate allocation technique is proposed for Fine-Granular Scalability (FGS) coded video, in which real data based rate-distortion modeling is developed, constant quality constraint is adopted and sliding window approach is proposed to adapt to the variable channel conditions. By using the proposed technique, constant quality is realized among frames by solving a set of linear functions. Thus, significant computational simplification is achieved compared with the state-of-the-art techniques. The reduction of the overall distortion is obtained at the same time. To combat the random error during the transmission, an unequal error protection (UEP) method and a robust error-concealment strategy are proposed for scalable coded video bitstreams.

ROBUST AND EFFICIENT VIDEO/IMAGE TRANSMISSION

by
Xi Min Zhang

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Electrical Engineering

Department of Electrical and Computer Engineering

January 2003

APPROVAL PAGE

ROBUST AND EFFICIENT VIDEO/IMAGE TRANSMISSION

Xi Min Zhang

Yun-Qing Shi, Dissertation Advisor — Date
Professor, New Jersey Institute of Technology

Nirwan Ansari, Committee Member — Date
Professor, New Jersey Institute of Technology

Constantine Manikopoulos, Committee Member — Date
Associate Professor, New Jersey Institute of Technology

Huifang Sun, Committee Member — Date
IEEE Fellow, Mitsubishi Electric Research Laboratories

Anthony Vetro, Committee Member — Date
Doctor, Mitsubishi Electric Research Laboratories

BIOGRAPHICAL SKETCH

Author: Xi Min Zhang

Degree: Doctor of Philosophy

Date: January 2003

Undergraduate and Graduate Education:

- Doctor of Philosophy in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2003
- Master of Engineering in Electrical Engineering,
  Nanyang Technological University, Singapore, 1999
- Bachelor of Science in Electrical Engineering,
  Xi'an Jiao Tong University, P. R. China, 1991.

Major: Electrical Engineering

Presentations and Publications:
Journal Publications:

X. M. Zhang and Yan Qiu Chen,
"Ray-Guided Global Optimization Method for Training Neural Networks,"
Neurocomputing , vol 30 (1-4), pp. 333-337, 2000.

X. M. Zhang, Yan Qiu Chen and Haroon A. Babri,
"Augmented TDNN for Frequency and Scale Invariant Sequence Classification,"
Neurocomputing, Accepted.

Y. Q. Shi and X. M. Zhang,
"A New Two-dimensional Interleaving Technique Using Successive Packing,"
IEEE Transactions on Circuits and Systems: Part I, vol. 49, no. 6, pp. 779-789, June 2002.

X. M. Zhang, A. Vetro, Y.Q. Shi, H. Sun,
"Constant quality constrained rate allocation for FGS coded video,"
IEEE Transaction on Circuits and Systems for Video Technology, Accepted.

X. M. Zhang, Y. Chen, N. Ansari and Y.Q. Shi,
"Mini-Max Initialization for Function Approximation,"
Neurocomputing, under review.

Conference Publications:

Yan Qiu Chen, X. M. Zhang and Haroon A. Babri,
"Training Neural Network using Ray-Guided Exploration Method,"
5th International Conference on Control, Automation,
Robotics and Vision (ICARCV98).

X. M. Zhang and Yan Qiu Chen,
"Frequency and scale invariant neural network for sequence classification,"
Proc., 2nd International Conf. on Information, Communications and Signal Processing, ICICS99, Singapore.

Y. Q. Shi and X. M. Zhang,
"Two-dimensional Interleaving by Using Successive Packing,"
38th Annual Allerton Conference on Communication, Control, and Computing, pp. 945-952, 2000.

X. M. Zhang, A. Vetro, H. Sun, Y. Q. Shi,
"Robust Decoding for Reduced Error Propagation of DC/AC Prediction Errors,"
IEEE 2nd Workshop and Exhibition on MPEG-4, pp. 91-94, 2001.

Z. Guo, X. M. Zhang, J. Savir, Y.Q. Shi,
"On Test and Characterization of Analog Linear Time-Invariant Circuits Using Neural Networks,"
Proc. Asian Test Symp., ATS'01, pp. 338-343, Nov. 2001.

X. M. Zhang, A. Vetro, Y. Q. Shi, H. Sun,
"Constant Quality Rate Allocation for FGS Video Coded Bitstreams,"
IS&T/SPIE symposium : Visual Communications and Image Processing 2002, pp. 817-827.

X. M. Zhang, Y.Q. Shi, A. M. Haimovich, H. Chen, A. Vetro, H. Sun,
"Successive Packing Based Interleaver Design for Turbo Codes,"
IEEE International Symposium on Circiuts and Systems (ISCAS) 2002.

X. M. Zhang, Y.Q. Shi,
"Equal Contribution Based Unequal Error Protection for FGS Coded Video,"
The 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI) 2002, Orlando, Florida.

X. M. Zhang, Y.Q. Shi, S. Basu,
"Basis array based successive packing approach for M-D interleaving,"
Fifteenth International Symposium on Mathematical Theory of Networks and Systems (MTNS) 2002, University of Notre Dame, Aug. 2002.

H. Kong, A. Vetro, H. Kalva, D. Fu, X. M. Zhang, J. Guo, H. Sun,
"A Platform for Real-Time Content Adaptive Video Transmission Over Heterogeneous Networks,"
SPIE ITcom, Aug. 2002, Boston.

To my beloved family.

ACKNOWLEDGMENT

Upon the completion of this dissertation, I would like to thank my advisor, Prof. Yun-Qing Shi, for his invaluable guidance, advice and encouragement through the years. His constant confidence in the face of research uncertainties and his persistent questioning of our methods has been and will always be inspiring to me.

During the period of my study, I was fortunate to have worked part-time under the supervison of Dr. Huifang Sun and Dr. Anthony Vetro. I am extremely grateful to them for introducing me to the world of standards and providing guidance on many things.

I would like to thank Prof. Nirwan Ansari and Prof. Constantine Manikopoulos for their precious time spent in reviewing this work and for their valuable suggestions during its development.

I would also like to thank Prof. Yanqiu Chen for helping me to develop the necessary skills to be a successful researcher when I pursued the Master's degree in Singapore. Special thanks are extended to the New Jersey Center for Wireless Telecommunications for its support and efforts in providing a friendly research environment.

Most of all, I thank my parents for their understanding and support through all my life. Without their encouragement and confidence on me, I would certainly not have accomplished the many things that I am proud of today. I'd also thank my sister for her support and help over years. Finally, I would like to express my most sincere appreciation to my wife, Helen, for her love, patience and encouragement.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1
INTRODUCTION

The subject of communication involves the transmission of information from a source that generates the information to one or more destinations. Great advances have been achieved in various areas including source-encoding technique to increase the efficiency; channel coding technique to improve the error resilient capability; physical medium technique to increase the transmission bandwidth and etc. Recently, with advances in network technology, the Internet has become a primary medium for information transmission. On-line entertainment and live broadcast over the Internet are common now. However, robust and efficient signal transmission is still an objective due to the unreliability of channel conditions, limited channel bandwidth and explosive growth of information transmission requirements. One reason for this observation is Metcalf's Law which states that the value of a network service grows with the square of the number of users [1]. Research about robust and efficient delivery of video/image content over networks is demanding nowadays.

1.1 Motivation

Most of the well-known channel codes that have been devised for increasing the reliability in the transmission of information are effective when the errors caused by the channel are statistically independent. However, burst error usually happens in some channels such as the wireless channel characterized by multipath and fading, and magnetic recording channels. Such burst errors cannot be corrected by the error control coding technique designed for statistically independent errors. To solve this problem, interleaving techniques have been proposed to combat burst errors during the transmission. Interleaving is a permutation of the order of the encoded sequence prior to the channel transmission. In the receiver, the received sequence is unscrambled to restore the original ordering. Thus, the channel actions appear

to be independent. By using interleaving, the error burst is spread into different codewords. It is then corrected by the error correction code.

The conventional interleaving technique is effective to combat 1-D burst error [2]. But two-dimensional (2-D) or three-dimensional (3-D) error bursts occur during video/image transmission. Besides video and image transmission, the scenarios where M-D error bursts may occur also include magnetic/optical and holographic data storage [3], charge-coupled devices (CCDs), 2-D barcodes [4], and integrated circuits design [5], to name a few. In magnetic and optical storage system, if the effects of neighboring tracks is considered, then such a system is a 2-D channel. In the holographic storage system, if the effects of angular multiplexing is considered, then such a channel is a 3-D channel [6]. The design of Application Specific Integrated Circuits (ASIC) is usually required to improve the performance of computation-intensive applications. A lot of such applications consist of multi-dimensional (M-D) problems such as computer vision, high-definition television, medical imaging and remote sensing. Therefore, the reliability issue of M-D information has arisen as an important task, having both theoretical and practical significance.

In order to solve this problem, extending the strategy of one-dimensional (1-D) interleaving technique to the M-D situation in order to combat error bursts with some simple random-error-correction codes has become the most common approach [4]. However, it can be shown that combining the 1-D interleaving technique and some writing procedure to constitute a 2-D interleaved array may not be able to combat 2-D error bursts effectively. This does not come as a surprise because the 2-D nature has not been taken into account with the 1-D procedure. Hence, it is necessary to develop efficient M-D interleaving techniques to secure the reliability of M-D digital data. In order to reach this objective, some fundamental investigations such as the characteristics of M-D error bursts, optimality of M-D interleaving, and the selection of suitable codeword are needed.

Interleaver design plays an important role to improve the performance of turbo codes. It is well known that the optimal performance can be obtained by using pseudo-random interleaver design. However, it is difficult to analyze the corresponding pseudo-random algorithms such that the performance can be guranteed, and they are computational expensive to implement. Moreover, they are not optimal with short block length. Thus, good determinitic interleaver design algorithms are desirable.

A key concern in the delivery of video content over networks is the ability to adapt the outgoing traffic to meet constraints imposed by users and networks. For the transmission of video over fixed bandwidth channels, the video signal is often encoded at a constant bit-rate (CBR). However, CBR does have certain drawbacks. One drawback is that picture quality fluctuates. Another drawback of CBR is that it does not provide an efficient means of transmitting video over time-varying heterogeneous networks. In both cases, either to provide constant-quality or improved-quality video, or to fully utilize the link capacity, video coded with variable bit-rate (VBR) is often preferred.

To achieve optimal bit allocation in the VBR coding, many techniques have been proposed [7], [8], [9]. Overall, previous optimal rate allocation techniques have provided ways to minimize the overall distortion (rate) subject to rate (distortion) constraints. Using Lagrange multiplier techniques to find the optimal solution is the most common approach. In this approach, the closed-form rate-distortion (R-D) model is required. The performance of the model based methods for rate allocation depends heavily on the accuracy of the selected model. It is found that the commonly used exponential model is not suitable to accurately model the R-D properties of FGS enhancement layer data at low bit-rates. This is consistent with the classic theory on this subject [10].

Users require playback with minimal variation in quality, but dynamic network conditions often make this difficult to achieve with single-layer coding schemes. With scalable coding, a multimedia server may store one copy of high quality video bit stream and deliver only part of the bit stream depending on the client demand and channel condition. Recently, Fine-Granular Scalability (FGS) coding [11] and Fine-Granular Scalability Temporal (FGST) coding [12] have been proposed and adopted as amendments to the MPEG-4 standard [13]. The scalable coding with fine granularity is a radical departure from traditional scalable coding schemes. It provides an enhancement layer that is continually scalable. This is accomplished through a novel bit-plane coding method of DCT coefficients in the enhancement layer, which allows the enhancement layer bitstream to be truncated at any point. An important point to emphasize is that the standard itself does not specify how the rate allocation, or equivalently, the truncation of bits on a per frame basis is done. The standard only specifies how a truncated bitstream is decoded. Thus, an optimal rate allocation strategy for FGS coded bit-stream will be desirable.

The compression standards such as JPEG2000, H.263, MPEG-2 and MPEG-4 significantly reduce the data rates of the image/video signal [14]. However, the compressed video signal by using these standards is extremely vulnerable to transmission errors. For example, in all of the above standards, variable length coding (VLC) is adopted to realize entropy coding. If any bit is damaged during the transmission, the following bits are either decoded wrong or cannot be decoded until the next resychronization bit is met. Thus, VLC causes the error propagation. Besides this, motion compensated coding is used to remove temporal redundency in video coding. In INTER mode, the loss of information in one frame has considerable impact on the quality of the following frames. Automatic Repeat on reQuest (ARQ) have been shown very effective to protect the signal from error during transmission [15]. However, retransmission of corrupted data frames introduces additional delay, which

is very critical for video applications. On the other hand, an effective forward error correction (FEC) technique can overcome this problem. If this technique can provide unequal error protection to the scalable coded video, it will be desirable. Although FEC or the other error protection techniques can protect the video during transmission, the bandwidth restriction limits the amount of error protection they can provide. In that case, the video decoder has to employ error concealment techniques to hide visible distortion [16].

1.2 Objectives

As what has been discussed in the last section, several areas in video/image transmission await further study and exploration. There are still problems to be solved and theories to be developed and refined. The objectives of this dissertation are to address the following areas:

- Effective M-D interleaving schemes.

- High quality turbo codes interleaver design method.

- Constant quality constrained optimal rate allocation for FGS coded bit-streams.

- FEC and error concealment for robust MPEG-4 video transmission.

Effective M-D interleaving schemes: Conventional 1-D interleaving does not work well to combat 2-D burst error. Although some M-D interleaving techniques have been proposed, they focus more on the theoretical analysis and have some deficiency on application. In this dissertation, it is aimed to propose a new M-D interleaving technique to combat M-D burst errors. This method should be optimal to error bursts with different size and easy to implement.

High quality turbo codes interleaver design method: An interleaver is the key component in both turbo encoder and decoder. It has been shown that the good

turbo codes performance can be obtained by using a pseudo-random interleaver. However, it is difficult to analyze the corresponding pseudo-random algorithms such that the performance can be guaranteed, and furthermore, they are computational expensive. In this dissertation, the research focuses on deterministic turbo codes interleaver design. It is aimed to design a family of interleavers that possess the following desirable features: prunability, adaptability to various criteria, and pseudo-random distribution.

Constant quality constrained optimal rate allocation for FGS coded bit-streams: The human vision system is very sensitive to the quality fluctuation among the consecutive video frames. This dissertation aims to find an optimal rate allocation strategy for Fine-Granular Scalability (FGS) coded bitstreams that can achieve constant quality reconstruction of frames under a dynamic rate budget constraint. In doing so, it also aims to minimize the overall distortion at the same time. To achieve this objective, a novel R-D labeling scheme to characterize the R-D relationship of the source coding process is needed. Due to the variable channel condition, this strategy should be able to adapt to the channel variation dynamically. Extending it to operate on multiple sources is desirable.

FEC and error concealment for robust MPEG-4 video transmission: Scalable video coding techniques have been used to adapt to the channel bandwidth variations. Due to different contribution to the overall quality, unequal error protection (UEP) is needed to provide graceful quality degradation when packets are damaged or discarded randomly. In this dissertation, the research focuses on the effective use of UEP to scalable encoded bitstreams, specifically, feed forward error correction (FEC) to FGS coded video bitstreams.

In the MPEG-4 standards, a new technique called DC/AC prediciton has been adopted to improve the compression efficiency. However, it also cause error propagation if the previous blocks are damaged. To conceal this kind of error propagation,

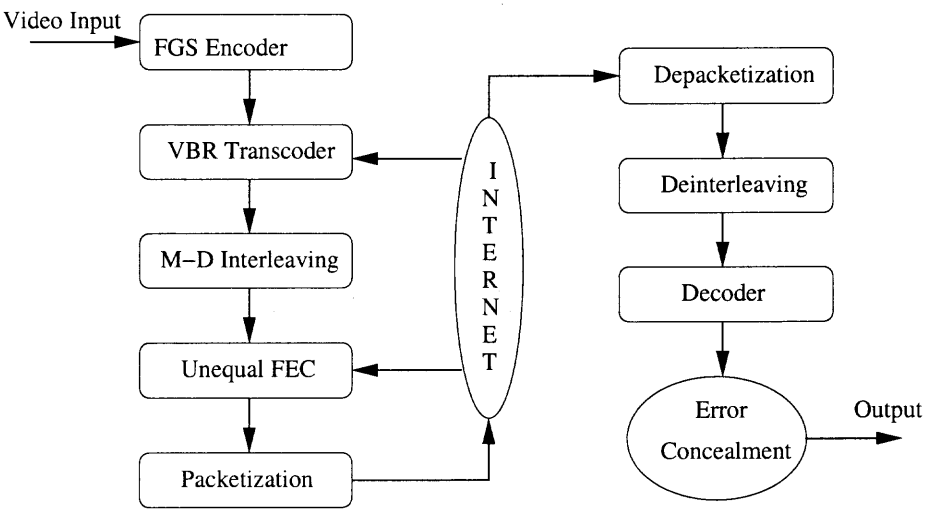


Figure 1.1 The overall framework of this dissertation research.

the smoothness of DC coefficient values and their gradients between the adjacent blocks to estimate the DC value of the damaged blocks will be used. The side effect of the concealment will also be considered.

In summary, the objective of this dissertation research focuses on three aspects of the robust and efficient video/image transmission: rate allocation of FGS coded video for efficient bandwidth utilization; M-D interleaving for combating bursts of errors; unequal error protection and error concealment for random error protection. This is described in Figure 1.1. The source signal is first encoded by FGS encoder. Then, based on the feedback from channel, the compressed bitstream is processed by the proposed VBR transcoder and the constant quality is obtained here. Before the transmission, the transcoded sequence is interleaved and the unequal error protection is added. At the receiver side, the received sequence is depacketized and deinterleaved according to the protocal between the server and client. It is then decoded. If the error is detected in some blocks, the error concealment strategy is applied to reduce the error propagation and improve the quality. It is noted that all of the above research works are conducted on the application layer.

1.3 Organization of the Dissertation

This dissertation comprises seven chapters. In this chapter, a brief introduction about some challenges in video/image transmission is presented. Next, the project objectives are stated together with scopes of research.

In Chapter 2, A novel successive packing based 2-D interleaving technique for combating 2-D burst error is proposed. The existing methods to deal with the M-D interleaving problem are first reviewed. Then, a novel method which use the successive packing of basis array to realize 2-D interleaving is proposed. The optimality of the proposed method is discussed. Some theoretical results are demonstrated and proved.

In Chapter 3, an effective interleaving scheme for M-D interleaving is proposed. To achieve this, a novel concept, basis interleaving array is first proposed. A general method to obtain a variety of basis interleaving arrays is presented. Based on the basis interleaving array and the successive packing method, an interleaving technique is then proposed by successive packing of basis interleaving array to generate the interleaved array with arbitrary size. It is shown that the proposed technique can spread any error burst of $m_0^k \times m_1^k$ within $m_0^n \times m_1^n$ array ($1 \leq k \leq n-1$) effectively so that the error burst can be corrected with some simple random error correcting-code (provided the error correcting-code is available). It is further shown that the technique is optimal for combating a set of arbitrarily-shaped error bursts. This algorithm needs to be implemented only once for a given M-D array, thereby reducing the computational cost.

In Chapter 4, the principle of successive packing (SP) is applied to the turbo codes interleaver design. It is shown that SP based interleaver design is optimal in the sense of pseudo-random distribution, prunability, and adaptability to various criterion. Simulation results demonstrate the superior performance of SP compared to other deterministic interleaver design technique.

In Chapter 5, an R-D labeling scheme is proposed to measure the real R-D pair of FGS coded bitstream. Piecewise interpolation is then used to approximate the R-D relationship. Based on this approximation, a sliding window based rate allocation approach is presented to realize constant quality among frames. The proposed method is applied to several sequences and the experimental results demonstrate the effectiveness of the proposed method.

In Chapter 6, two issues for robust MPEG-4 video transmission are approached: forward error correction (FEC) and error concealment. An effective contribution vector based unequal error protection method is proposed to protect the FGS coded video. A novel error concealment strategy to combat the error propagation caused by DC/AC prediciton in MPEG-4 video is also proposed.

Finally, in Chapter 7, a summary of the work is given. Directions for future work are suggested.

CHAPTER 2

SUCCESSIVE PACKING FOR TWO-DIMENSIONAL INTERLEAVING

The event in which Yahoo and a few other web sites were struck in February 2000 has lead to the following common belief that digital data safety is an issue of national security. This chapter deals with protection of multi-dimensional (M-D, $M \geq 2$) digital data. Specifically, how bursts (clusters) of errors taking place with M-D data can be corrected. What is proposed here is a novel method, called successive packing (SP), to two dimensional (2-D) interleaving, which can be potentialy extended to M-D interleaving.

Although some 2-D burst error correction codes have been developed [17, 18], extending the strategy of one-dimensional (1-D) interleaving technique to the M-D situation in order to combat error bursts with some simple random-error-correction codes has become the most common approach to the correction of error bursts. Some M-D interleaving techniques for combating M-D error bursts have been proposed [19, 4, 20, 21]. Among them, Almeida and Palazzo Jr. present some 2-D interleaving results for circularly shaped error bursts [19]. The United Parcel Service (UPS) combine the 1-D interleaving technique and some writing procedures to protect their 2-D barcode [4]. Adbel-Ghaffar study some theoretical aspects of 2-D interleaving [20]. The M-D interleaving technique reported in [21] is considered as the most comprehensive existing one. It defines an error burst as an arbitrarily-shaped, connected area (in the 2-D case) or a volume (in the 3-D case). For each given burst size, t_0, a specific algorithm is implemented, which can correct arbitrarily-shaped error burst of size t_0 and is optimal in the 2-D case. It is observed that when the burst size, t, increases, i.e., $t > t_0$, the algorithm with a set of new parameters has to be implemented once again in order to correct the larger error burst of arbitrary shape. When the burst size decreases, i.e., $t < t_0$, the interleaved array obtained

with respect to t_0 is no longer optimal. Because the size of error bursts is not known in advance (this is usually the case in reality), the application of the technique is somehow limited.

In contrast, the size of a given 2-D array is known in many applications. For instance, the size of an image or a video frame is normally known in advance. Motivated by these observations, a novel method, called successive packing (SP), is proposed to 2-D interleaving, as a different and complementary technique to the technique in [21]. This method can potentialy be extended to M-D interleaving. The main idea behind the proposed technique is to interleave the 2-D data neighboring in a given 2-D array as far away from each other as possible in both horizontal and vertical directions, and this is realized successively. It is this strategy that makes the proposed technique simple in both theoretical analysis and practical implementation, and, yet, powerful in interleaving.

The square arrays of $2^n \times 2^n$ are considered. It is shown that the proposed SP technique can spread any error burst of $2^k \times 2^k$ (with $1 \leq k \leq n-1$), $2^k \leq 2^{k+1}$ (with $0 \leq k \leq n-1$), and $2^{k+1} \times 2^k$ (with $0 \leq k \leq n-1$) effectively so that the error burst can be corrected with some simple random-error-correction code (provided the error correction code is available). It is further shown that the technique is optimal for correcting all the above-defined error bursts in the sense that the interleaving degree reaches its lower bound. Consequently, the algorithm needs to be implemented only once for a given 2-D array and is thereafter optimal for the set of error bursts having different sizes.

The chapter is organized as follows. In Section 2.1, 1-D interleaving technique is overviewed, the philosophy and application of interleaving techniques is presented. It is then shown that 1-D interleaving-based technique cannot work effectively in the case of 2-D barcodes, meaning that it is necessary to develop 2-D and, in general, M-D interleaving techniques. Next, in Section 2.2, the limitations of the existing

M-D interleaving techniques are commented, based on which, a novel SP approach is proposed to 2-D interleaving in Section 2.3. It is shown that the proposed approach can work well in the practical cases where the size of 2-D arrays is given and the size of error bursts is not known in advance. A performance comparison between the proposed method and some existing techniques is given. Section 2.4 provides a brief summary.

2.1 1-D Interleaving

2.1.1 Philosophy of Interleaving

1-D interleaving technique has been well written in some error correction coding texts, e.g., in [2]. The main idea is to mix up the code symbols from different codewords so that error bursts encountered in the transmission are spread across multiple codewords when the codewords are reconstructed at the receiving end. Consequently, the error occurring within one codeword may be small enough to be corrected by using some simple random-error-correction code. This can be seen clearly from a simple example that follows. Consider a code in which each codeword contains four code symbols. Furthermore, the code has what is known as random-error-correction capability. Without loss of generality, it is assumed that here one-random-error-correction capability, i.e., any single code symbol error occurring in one codeword can be corrected. Suppose there are 16 symbols, which correspond to four codewords. That is, code symbols from 1 to 4 form a codeword, from 5 to 8 another codeword, and so on. One of the 1-D interleaving procedures first creates a 2-D array of 4×4. The 16 symbols are then read into the 2-D array in a column-by-column (or row-by-row) manner. The interleaved symbols are obtained by writing the symbols out of the 2-D array in a row-by-row (or column-by-column) fashion. This process has been depicted in Figure 2.1 (a), (b), and (c). Now take a look at how this interleaving technique can combat error bursts. Note that an error burst

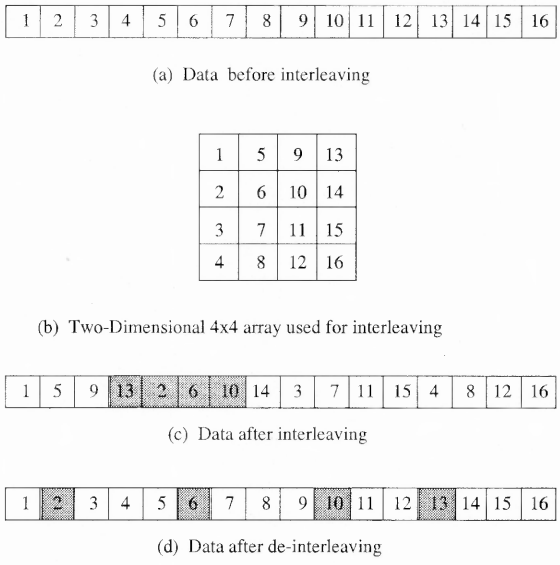


Figure 2.1 1-D interleaving procedure and its performance.

involving four consecutive symbols is shown in Figure 2.1 (c) with shades. When such an error burst takes place in the 1-D interleaved data, Figure 2.1 (d) demonstrates that the error burst can be spread effectively among four codewords in the de-interleaved array so that there is only one symbol in error for each of the four codewords. With the one-random-error-correction capability, it is obvious that no decoding error will result from the presence of the error burst. This simple example demonstrates the effectiveness of 1-D interleaving technique in combating 1-D error bursts.

2.1.2 1-D Interleaving: Not Effective for Combating 2-D Error Bursts

To enhance the reliability of M-D digital data, which is of crucial importance in the information age, codes that can correct M-D error bursts are desired. Here, the necessity of developing M-D interleaving techniques is demonstrated via an example that 1-D interleaving-based techniques cannot work well for correcting 2-D error bursts. It is known that 2-D barcodes are information storage media in which the source information is stored as a bit stream on a printed label. Examples of

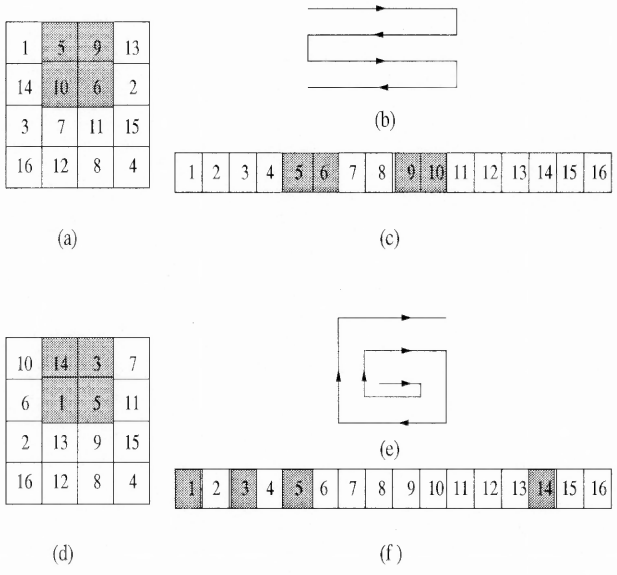


Figure 2.2 Boustrophedonic Pattern and Spiral Data Pattern.

2-D barcode applications include the bracelets, that are used in hospitals to carry patients' entire medical histories; the labels put on parts, that are used in automotive assemble processes to carry a unique identification number and other pertinent information applicable to production, tracking, and statistical process control [22]; and the labels that are used as portable data files that accompany packages in shipping industry [4]. When the United Parcel Service (UPS) developed its own 2-D bar codes, 1-D interleaving technique was used to combat 2-D bursts. That is, a sequence of code symbols is first 1-D interleaved. The 1-D interleaved symbols are then written into a 2- D array (printed on a 2-D label) according to some writing pattern. Specifically, there are two different patterns used by the UPS: the Boustrophedonic Pattern and the Spiral Data Pattern [4], which are shown in Figure 2.2 (b) and (e), respectively.

Now, consider again the scenario of the 16 code symbols discussed above. After the 16 symbols have been 1-D interleaved (refer to Figure 2.1 (c)), they are written into a 2-D array of 4×4 according to either the Boustrophedonic Pattern or the Spiral Data Pattern. The 2-D arrays of 4×4 obtained by applying these two writing patterns are shown in Figure 2.2 (a) and (d), respectively. Let's examine the performance of

these two procedures by checking if they can combat a 2-D error burst of 2×2. Take a look at an error burst of 2×2 shown in Figure 2.2 (a) with shades. Figure 2.2 (c) indicates that this 2×2 error burst has not been spread effectively so that there are two code symbols in error in the second codeword. That is, the 1-D interleaving technique plus the Boustrophedonic Pattern writing procedure cannot combat the 2×2 error bursts. The same observation may be obtained for the case of using the Spiral Data Pattern writing procedure, which has been shown in Figure 2.2 (d) and (f). In summary, combining the 1-D interleaving technique and some writing procedure to constitute a 2-D interleaved array may not be able to combat 2-D error bursts effectively. This does not come as a surprise because the 2-D nature has not been taken into account with the 1-D procedure. Therefore, it is necessary to develop efficient M-D interleaving techniques to secure the reliability of M-D digital data.

2.2 Existing 2-D and M-D Interleaving Techniques

While some 2-D burst error correction codes have been developed [17, 18], M-D interleaving technique followed by some simple random-error-correction code has become the most common approach to correction of M-D error bursts. Some M-D interleaving techniques for combating M-D error bursts have been proposed [19, 21, 23]. Some theoretical aspects of the task, in terms of the definitions of 2-D and M-D bursts, the optimality of interleaving, the existence of the optimal interleaving and so on have been studied in [20, 21]. It is noticed that the organization of raster-graphics memory encounters the same issue as that faced by the interleaving technique [24].

Since the objective is to propose a new interleaving technique for combating 2-D error bursts, the focus is on the existing M-D interleaving techniques, instead of 2-D burst error correction codes. Observing that what presented in [19] (dealing with an error burst of circular-shape) is only very briefly described and cannot be generalized; what presented in [20] is only some concepts and unproved theoretical results; and

that presented in [21] is the most recent and comprehensive M-D interleaving scheme, only the technique presented in [21] is described in this section.

Instead of defining an error burst as a rectangular area or a circular area, Blaum et al. defined a 2-D error bursts as an arbitrarily-shaped, connected area. Let us take a look at Figure 2.3, where all the code symbols (assigned to the elements of the 2-D array) marked with star form a 2-D error burst. Note that all of these symbols are connected to each other and the connectivity here is constrained to the horizontal and vertical directions. This definition can be generalized to the M-D case. The size of a burst is defined as the total number of code symbols contained in the burst. Hence, the size of the burst in Figure 2.3 is 7. In [21], some 2-D and 3-D interleaving schemes were proposed. It is shown that in the 2-D case, the lower bounds of interleaving degree (optimality) can always be achieved by the schemes. The optimality, however, cannot be guaranteed in general for the M-D ($M > 2$) case.

The key idea of the Blaum et al. approach is related to the concept of the Lee-spheres and the closed tiling (or packing). For more information of these two concepts, interested readers may refer to [25]. Linking the Lee spheres with the odd burst sizes and creating some spheres for the even burst sizes, Blaum et al. use the spheres as building blocks to construct interleaved arrays. They have shown that if one labels each element in the building block with a distinct code symbol and uses the building block to closely (meaning no uncovered elements) tile (meaning no overlapping between blocks) a large enough 2-D area (or M-D volume), then one can produce an interleaved array. In this interleaved array, each element in any arbitrarily shaped connected subsets consisting of t elements is labeled with a distinct code symbol. All code symbols of the same kind form a codeword. Consequently, the error burst of size t can be corrected with one-random-error-correction codes. That is, the closely tiling of the 2-D array (or M-D volume) with the building block

generates a 2-D (or M-D) interleaved array. The tiling process will be defined in Section 2.3.

Though it can effectively spread arbitrarily-shaped 2-D error bursts of size t, the above characterization of the technique does reveal some limitations on the other hand. Firstly, the technique is based on the size of an error burst, t. For combating error bursts of size equal to or less than a specific t_0, one needs to implement a corresponding algorithm to construct an interleaving code. When the size t increases, i.e., $t > t_0$, one needs to implement the algorithm with a new set of parameters to construct another interleaving code. That is, the interleaved array constructed for a specific t_0 may not be able to correct an error burst of size t with $t > t_0$. Since in reality, e.g., in the application of 2-D barcodes, *the size of error bursts may not be known in advance.* This means that the implementation of the technique may be inconvenient.

Secondly, when the actual size of a burst, t, is less than t_0, with which the interleaving algorithm is applied, the technique is no longer optimal. This can be justified as follows. In [21], the optimality means that the interleaving degree reaches its lower bound. As mentioned before in the 2-D case, the interleaving degree, associated with an interleaving scheme designed for some burst size t, is guaranteed to reach its lower bound. Furthermore, it is known that the lower bound of the interleaving degree is a monotonically increasing function of the burst size t. Specifically, the lower bound is $t^2/2$ for even t and $(t^2+1)/2$ for odd t. Therefore, with respect to the implementation of the interleaving scheme designed for a burst size t_0, when the actual size of a burst, t, is smaller than t_0, the achieved interleaving degree with t_0 is larger than the lower bound corresponding to t. That is, the interleaving scheme designed for a burst size t_0 is not optimal for a smaller burst size, t. The definition of interleaving degree will be given below.

0	0	0	0	0	0	0
0	*	*	*	0	0	0
0	0	*	*	*	0	0
0	0	0	*	0	0	0

Figure 2.3 An error burst of size 7.

In many applications, *the size of a 2-D or M-D array is known.* For instance, a digital (watermarked) image may be known to have a size of 512 by 512 pixels. Under the circumstances, one may wonder if it is possible to develop a 2-D interleaving technique, which is optimal for all (if possible) or (at least) for many of possible burst sizes. Therefore, it can be implemented only once for a given 2-D array. Motivated by these observations, a novel M-D interleaving technique is proposed below.

2.3 Proposed Successive Packing Technique

To facilitate the description of the proposed SP technique, some definitions are given, and the optimality is discussed prior to presenting the SP technique.

2.3.1 Definitions: M-D Bursts, Tiling, Interleaving Degree

Definition 2.3.1. *Let C be an M-D code of $n_1 \times n_2 \cdots \times n_m$ over GF(q). A codeword of C is an M-D array of $n_1 \times n_2 \cdots \times n_m$, with each element of the M-D array assigned with a code symbol.*

Note that $GF(q)$ denotes Galois field, which is a field with a finite number, q, elements. The simplest field is the binary field, $GF(2) = \{0, 1\}$.

Definition 2.3.2. *In 2-D arrays, the neighbors of element (x, y) are denoted by*

$$(x+1, y), (x-1, y), (x, y+1), (x, y-1)$$

In 3-D arrays, the neighbors of element (x,y,z) are denoted by

$$(x+1, y, z), (x-1, y, z), (x, y+1, z),$$

$$(x, y-1, z), (x, y, z+1), (x, y, z-1)$$

provided those elements exist.

Notice that Definition 2.3.2 can be extended to higher dimensional arrays. For M-D arrays, the total number of neighbors is 2M.

Definition 2.3.3. *In M-D arrays, a burst is a subset of the given M-D array, B, in which any element has at least one neighbor contained in B. Its size is defined as the number of elements in B.*

Note that in what follows the treatment is presented with respect to one-random-error-correction codes for the sake of brevity of the discussion unless otherwise stated. The results can be straightforwardly extended to r-random-error-correction codes with $r > 1$. The strategy of interleaving to combat bursts of M-D errors is similar to that in the 1-D situation. Loosely speaking, with interleaving, the elements in an M-D array are rearranged so that error bursts occurred in the interleaved M-D array are separated as far away from one another in the de-interleaved array as possible. Hopefully, there is only one error in each codeword in the de-interleaved M-D array. Hence, M-D error bursts can be corrected.

Definition 2.3.4. *Let F be an M-D array, and A a large enough M-D array. If A is a union of some translated versions of F, and all the translated versions of F in A are mutually disjoint, then A is said to be a tiling with F.*

Conceivably, if each element in F is assigned a distinct code symbol, and consider all the code symbols of the same kind a codeword, then the tiling of A with F becomes an interleaved array.

Definition 2.3.5. *Let A be a tiling with F and each element in F is assigned a distinct code symbol, the size of F, denoted by γ_F, i.e., the number of distinct code symbols contained in F, is called interleaving gain. The total number of distinct code symbols used in interleaving is normally referred to as interleaving degree, denoted by D.*

Equivalently, the interleaving degree can be defined as the total number of codewords used in interleaving because all the code symbols of the same kind constitute a codeword. It is seen that if A is a tiling with F, each element in F is marked with a distinct code symbol, and if the code symbol assignment in all the translated version of F is the same, the interleaving degree is equal to the interleaving gain.

2.3.2 Optimality: Lower Bound of Interleaving Degree

Interleaving generally means mixing up code symbols so that each element in an error burst can be spread into a different codeword (with respect to one-random-error-correction codes). Therefore, normally the larger the size of an error burst, the larger the number of codewords (the larger the interleaving degree) required. Optimality has something to do with the lower bound of the interleaving degree required.

Lemma 2.3.1. *Let A be an M-D array, the interleaving degree should be bounded below by $D \geq \gamma$, where γ is the interleaving gain.*

From Definition 2.3.5, the proof of this Lemma is obvious. Note that although Lemma 2.3.1 presents the lower bound theoretically, it doesn't necessarily mean that there exists a method to realize it. In fact, when $M > 2$, this lower bound cannot be obtained in general as shown in [25], [21].

Theorem 2.3.1. *Given a burst B in M-D array A, if there exists an interleaving method that can spread the burst B effectively, then the interleaving degree must*

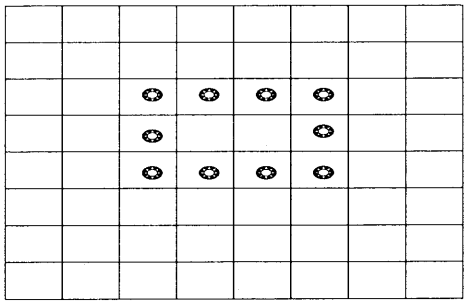


Figure 2.4 A building block F of special shape.

satisfy the following formula

$$D_A \geq \gamma_B \tag{2.1}$$

where γ_B is the number of elements in B, by effectively *it is meant that each element of the burst B is spreaded into a distinct block with the same size.*

Consider an M-D array A, which is a tiling with F, and F is a smallest subset of A containing entirely the error burst B. Then, obviously $\gamma_F \geq \gamma_B$. Therefore, the proof of Theorem 2.3.1 can be completed by using Lemma 2.3.1. It should be noted that when an M-D array A is given, for a building block F of some shape, there may not exist a tiling of A with F. An example supporting this observation is presented in Figure 2.4. Under the circumstances, the building block depicted in Figure 2.4 cannot be used for tiling because some unused elements will result, violating the definition of tiling. Hence, some redundant elements must be added to the building block such that the new block structure can produce a tiling of A. Therefore, the result of $D > \gamma_F$ is obtained.

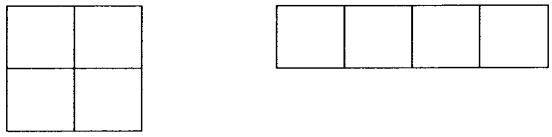
2.3.3 Successive Packing for 2-D Interleaving

It is noted that an initial version of the proposed interleaving technique was presented in [26]. Afterwards a more rigorous presentation, called successive partition, was given in [27]. However, in both of these works, the authors were in ignorance of the existing M-D interleaving techniques [21]. Furthermore, the authors' initial works

only consider the case of 1-D codewords. Except for some graphical examples, the burst error correction capability of the presented interleaving techniques has not been theoretically analyzed and proved; the optimality issue has never been addressed.

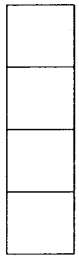
In this subsection, the proposed successive packing technique is presented for 2-D interleaving. The successive packing technique deals with 2-D codewords (1-D codewords are considered as a special case). Its burst error correction capability is explicitly claimed and vigorously proved. Its optimality is discussed and proved. These results were partially presented in [28].

2.3.3.1 Codewords and 1-D Sequence of Code Symbols. Given that digital images, video frames, charge-coupled devices (CCDs), and 2-D bar-codes are all in the form of 2-D arrays, without loss of generality, square arrays of $2^n \times 2^n$ are considered here. The utilization of $2^n \times 2^n$ arrays will be further justified later. In general, the codewords in the 2-D case are of 2-D in nature as defined in Definition 2.3.1. 1-D codewords, either row-type, or column-type, or other-type, can be considered as special cases of 2-D codewords. The proposed technique is able to handle 2-D codewords because all the code symbols in the 2-D codewords are first organized into a 1-D sequence of code symbols in the successive packing. Without loss of generality, the quartering indexing scheme is described below to provide an illustration. That is, a square array of $2^n \times 2^n$ is viewed as consisting of four quadrants, each quadrant itself consists of its own four quadrants, the process repeats itself until at a certain level where all four quadrants are of 2×2. This can be considered as 2-D successive doubling. These 2×2 arrays are the fundamental structure. When the quartering indexing scheme is applied, each element in the array has a pair of subscripts. The first subscript represents the index of the 2×2 array in which the code symbol is located, while the second subscript represents the index of the code symbol within the 2×2 array. To convert the quartering index, $s_{i,j}$, into the 1-D index, s_k, the following operation: $k = 4i + j$, may be used.

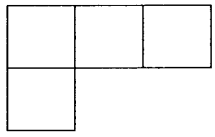


(a) Square codeword

(b) Row codeword



(c) Column codeword



(d) Bent line codeword

Figure 2.5 Four different types of 2-D codewords having four code symbols.

Comment: Quartering indexing is not the only choice for the proposed interleaving technique. Actually, codewords can be of any shape. Several shapes having four code symbols are shown in Figure 2.5. Obviously, for any given shape of 2-D codewords, it is always possible to label the code symbols into a 1-D sequence with a possibly more complicated book keeping scheme.

2.3.3.2 Successive Packing. Now the proposed interleaving technique is presented in 2-D case in a general and compact way, which is expected to be able to generalize to M-D case straightforwardly.

Procedure 2.3.1. *The 2-D interleaving using the successive packing proceeds as follows. Consider a 2-D array of* $2^n \times 2^n$ *for 2-D interleaving.*

When $n = 0$, *the interleaved array of* 1×1 *is the original array itself. That is,*

$$S_1 = [s_0] \tag{2.2}$$

where s_0 *represents the element in the array, and* S_1 *the array. It is noted that the subscript in the notation* S_1 *representing the total number of elements in the interleaved array. Hence, when* $n = 1$, *i.e., for a* 2×2 *array, the interleaved array*

is denoted by S_4; when $n = 2$, the interleaved array is S_{16}. In general, for a given n, the interleaved array is denoted by $S_{2^{2n}}$.

The procedure is carried out successively. Given interleaved array S_i, the interleaved array of S_{4i} can be generated according to

$$S_{4i} = \begin{bmatrix} 4 \times S_i + 0 & 4 \times S_i + 2 \\ 4 \times S_i + 3 & 4 \times S_i + 1 \end{bmatrix} \tag{2.3}$$

where the notation of $4 \times S_i + k$ (with $k = 0, 1, 2, 3$) represents a 2-D array that is generated from S_i. This means that $4 \times S_i + k$ has the same dimensionality as S_i. Furthermore, each element in $4 \times S_i + k$ is indexed in such a way that its subscript equals to the four times of that of the corresponding elements in S_i plus k. The corresponding element means the element occupying the same position in the 2-D array. It appears from Equation 2.3 that S_{4i} is derived from S_i by packing S_i four times. This explains why the term successive packing is used.

According to the above rule, it can be obtained that

$$S_4 = \begin{bmatrix} 4 \times S_1 + 0 & 4 \times S_1 + 2 \\ 4 \times S_1 + 3 & 4 \times S_1 + 1 \end{bmatrix} = \begin{bmatrix} s_0 & s_2 \\ s_3 & s_1 \end{bmatrix} \tag{2.4}$$

Similarly, S_{16} is derived as follows.

$$\begin{aligned} S_{16} &= \begin{bmatrix} 4 \times S_4 + 0 & 4 \times S_4 + 2 \\ 4 \times S_4 + 3 & 4 \times S_4 + 1 \end{bmatrix} \\ &= \begin{bmatrix} s_0 & s_8 & s_2 & s_{10} \\ s_{12} & s_4 & s_{14} & s_6 \\ s_3 & s_{11} & s_1 & s_9 \\ s_{15} & s_7 & s_{13} & s_5 \end{bmatrix} \end{aligned} \tag{2.5}$$

2.3.3.3 Main Results. Prior to presenting the main result contained in Theorem 2.3.2, the following prerequisite is introduced.

Definition 2.3.6. *Let A be a 2-D array of $2^n \times 2^n$. Each element of A, $s_{i,j}$, is re-indexed as s_k with k being a function of i and j. (For instance, for the quartering index described above, $k = 4i + j$.) After having 1-D indexing, consider a division of*

A into 2^m blocks with $1 \leq m \leq 2n-1$. That is, each block thus generated contains $K = 2^{2n-m}$ elements. The first block contains elements having the subscripts from 0 to $2^{2n-m}-1$. In general, for any element having an index k satisfying $2^{2n-m}(d-1) \leq k < 2^{2n-m}d$, $1 \leq d \leq 2^m$, this element is said belonging to the dth block. The element having index $2^{2n-m}(d-1)$ is referred to as the beginning element of the dth block. All the elements belonging to the same block are referred to as the K-equivalent elements, where $K = 2^{2n-m}$.

According to the above definition, it is seen that s_{2l}, s_{2l+1} are 2-equivalent elements (when $m = 2n-1$); s_{4l}, s_{4l+1}, s_{4l+2}, s_{4l+3} are 4-equivalent elements (when $m = 2n-2$). It is obvious that K_1-equivalent elements are also K_2-equivalent elements if $K_1 < K_2$.

Definition 2.3.7. *Consider two bursts, B_1 and B_2, in an interleaved 2-D array. If these two bursts have the same size and shape, and each element in a burst (e.g.,B_1) is either an element of another burst (e.g.,B_2) or a K-equivalent element of an element of another burst (B_2), and vice versa, then these two bursts, B_1 and B_2, are called the K-equivalent bursts.*

In the remainder of this chapter, when discussing the error burst correction, each block defined in Definition 2.3.6 may be considered as a 2-D codeword. This implies that a codeword consists of a set of different code symbols. This way to characterize codewords is seemingly different from the previous way given immediately following Definition 2.3.4 (i.e, a codeword is defined as a collection of all the code symbols of the same kind). However, it is understood that the difference lies only in the way of characterization. The essence of codeword remains the same. The new way is necessary since it is needed to discriminate code symbols within a codeword in the following discussion of the SP technique for 2-D interleaving. An error burst (in the interleaved array) is said to be spread and can be corrected with

one-random-error-correction codes if each element in the burst has been spread in the de-interleaved array into a distinct codeword. From this point of view, it is easy to see that given two equivalent bursts, if one has been interleaved, the other must have also been interleaved.

To present Theorem 2.3.2, the following lemma is provided first. With this powerful lemma, the proof of Theorem 2.3.2 will be drastically simplified.

Lemma 2.3.2. *Let A be an array of $2^n \times 2^n$ obtained by using the successive packing procedure. Then all square bursts of $2^k \times 2^k$ in A with $k < n$ are K_1-equivalent to one another, where $K_1 = 2^{2n-2k}$; all rectangular bursts of $2^k \times 2^{k+1}$ in A with $k < n$ are K_2-equivalent to one another, where $K_2 = 2^{2n-2k-1}$, and all bursts of $2^{k+1} \times 2^k$ in A with $k < n$ are also K_2-equivalent to one another.*

The proof of this lemma is contained in Appendix A. Now it is the time to present the following theorem.

Theorem 2.3.2. *Consider a 2-D array of $2^n \times 2^n$, partition it into 2^m blocks for an integer m satisfying $1 \leq m \leq 2n-1$ according to Definition 2.3.6. When m is even and $m = 2k$, any square burst of $2^k \times 2^k$ in the interleaved array, A, obtained by using the successive packing is spread in the de-interleaved array so that each element of the burst falls into a distinct block of size 2^{2n-2k}. When m is odd and $m = 2k+1$, any rectangular burst of $2^k \times 2^{k+1}$ or $2^{k+1} \times 2^k$ in A is spread in the de-interleaved array such that each element of the burst falls into a distinct block of size $2^{2n-2k-1}$.*

Meaning of Theorem 2.3.2: Theorem 2.3.2 claims that in a 2-D array of $2^n \times 2^n$, A, generated with the successive packing technique, any square error burst of $2^k \times 2^k$ with $1 \leq k \leq n-1$ and any rectangular error burst of $2^k \times 2^{k+1}$ or $2^{k+1} \times 2^k$ with $0 \leq k \leq n-1$ can be spread so that each element in the burst falls into a distinct block in the de-interleaved array, where the block size, K, is 2^{2n-2k} for the burst of $2^k \times 2^k$, and $2^{2n-2k-1}$ for the burst of $2^k \times 2^{k+1}$ or $2^{k+1} \times 2^k$. This indicates that,

if a distinct code symbol is assigned to each element in a block (refer to Definition 2.3.6) and all the code symbols associated with an individual K-equivalent class form a distinct codeword, then this technique guarantees that the burst error can be corrected with a one-random-error-correction code, provided the code is available. (Note that the code capable of correcting one code symbol error within a codeword of two code symbols does not exist in reality. Therefore, though the error burst of $2^{n-1} \times 2^n$ or $2^n \times 2^{n-1}$ can be effectively spread in the de-interleaved arrays as described above, in fact, they cannot be corrected with a one-random-error-correction code.) Furthermore, the interleaving degree equals to the size of the burst error, hence, minimizing the number of codewords required in an interleaving scheme. In other words, with the successive packing technique, the interleaving degree obtains the lower bound (the interleaving gain). In this sense, the successive packing interleaving technique is optimal. If a coding technique has a strong random-error-correcting capability, say, it can correct one error in every codeword of size four, then any error bursts of $2^{n-1} \times 2^{n-1}$ can be corrected. If a code, on the other hand, has a less strong random-error-correcting capability, say, it can only correct one random error within a codeword of size 64, then only smaller error bursts, i.e., any burst of $2^{n-3} \times 2^{n-3}$ in the interleaved array can be corrected with the successive packing.

The proof of this theorem is contained in Appendix A.

2.3.4 Performance Comparison

In this subsection, the performance of three techniques: UPS technique, the interleaving technique developed in [21], and the proposed SP technique are compared. It is started with an example, then follows some theoretical analysis.

Figure 2.6 shows the interleaved 2-D arrays of 8×8 obtained by applying the three techniques and the de-interleaved array, from which the following observations can be made. *Firstly*, look at an error burst of 2×2 located in the middle of the

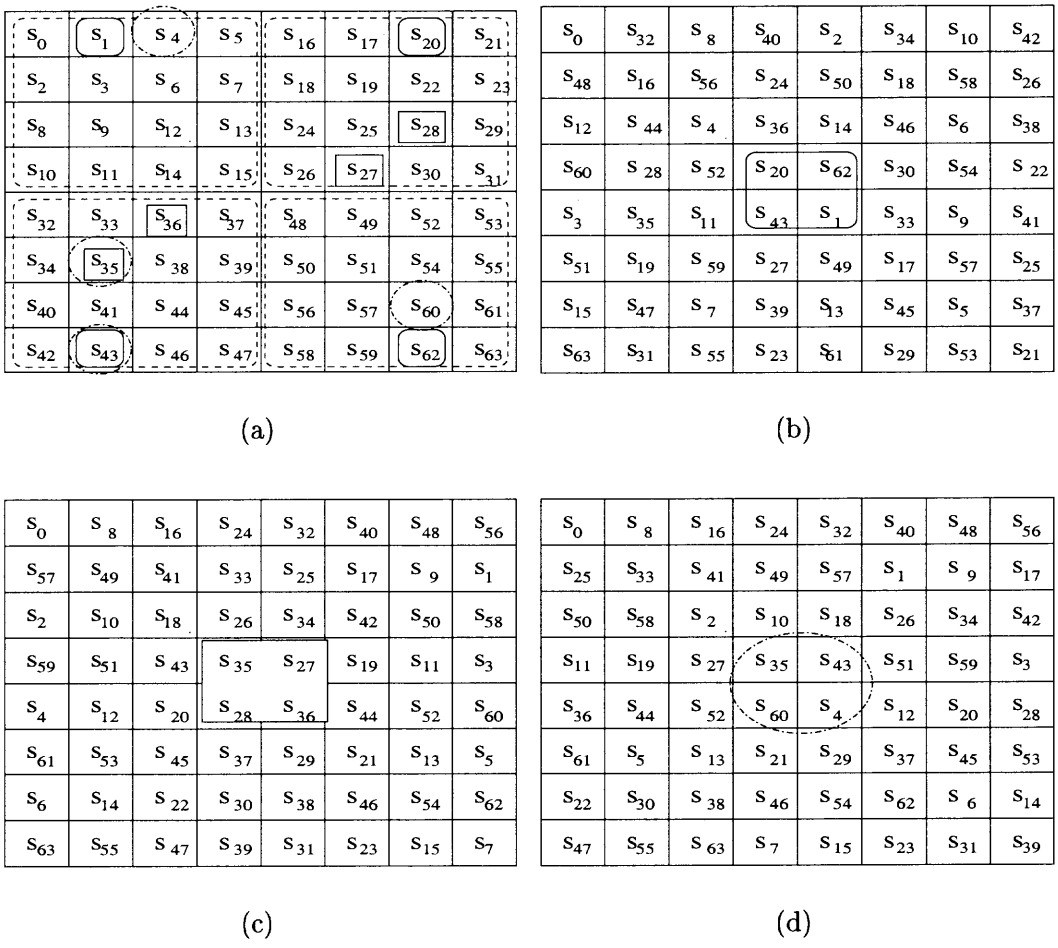


Figure 2.6 Performance comparison of three 2-D interleaving techniques: (a)De-interleaved array, (b) Interleaved array with the SP technique, (c) Interleaved array with the UPS' technique (Boustrophedonic Pattern), (d) Interleaved array with the technique in [6] (Example 2.3).

interleaved arrays. It involves the following four elements: s_{20}, s_{62}, s_{43}, s_1 in the interleaved array with the SP; s_{35}, s_{27}, s_{28}, s_{36}, with the UPS'; and s_{35}, s_{43}, s_{60}, s_4 with the method in [21]. After de-interleaving, the SP spreads the four elements in the error burst into four different quadrants, while either the UPS' method or the method in [21] has two error elements located within one quadrant. This implies that if a code can only correct one random error in each codeword consisting of 16 code symbols, then only the SP can correct the 2×2 error burst. In other words,

only the SP can correct the 2×2 error burst with four codewords, while the other two methods require more than four codewords. *Secondly*, it is noticed that, with respect to the interleaved 2-D array of 8×8, the SP can optimally spread the following types of bursts: any burst of 2×4, or 4×2, or 2×2, according to Theorem 2.3.2; while the method in [21] can spread arbitrarily-shaped error bursts of size 4 with an interleaving degree of 8. Since any error burst of size 4 in an array of 8×8 can be entirely included in either a burst of 2×2, or a burst of 2×4, or a burst of 4×2, it is clear that the SP can also spread and correct any arbitrarily-shaped error burst of size 4 in the interleaved 2-D array of 8×8 with some one-random-error-correction code. As far as the efficiency is concerned, if a burst of size 4 is of 2×2, then as shown above, the SP reaches an interleaving degree of 4, hence, is more efficient. If, more general, a burst of size 4 is included in a burst of 2×4 or 4×2, then the interleaving degree is 8, indicating that in this case the SP achieves the same efficiency as the technique [21] does. *Thirdly*, in addition to the three types of bursts described above, the SP can also optimally spread and correct with some one-random-error-correction code any bursts of 1×2, or 2×1, or 4×4 within the interleaved array of 8×8, according to Theorem 2.3.2. Hence, it is observed that the SP is versatile.

Comment 1: Both the example shown in Section 2 and the example shown above demonstrate that the UPS' approach (using 1-D interleaving plus a special writing procedure to form a 2-D interleaved array) does not work well in combating 2-D error bursts.

Comment 2: While the SP technique works for 2-D arrays of $2^n \times 2^n$, the technique in [21] works for 2-D arrays of $m \times m$, where $m = t^2/2$ for even t and $m = (t^2+1)/2$ for odd t with t being the size of error bursts. In what follows the observation made in the above example for the case of the error burst size $t = 4$ is generalized to the case of any error burst of size 2^p , where p is a positive integer.

Lemma 2.3.3. *Let C be a cluster of size 2^k in 2-D array $2^n \times 2^n$, where $k \leq n$. Then there must exist a rectangular block R_1 and/or a rectangular block R_2, where R_1 is of $2^k \times 2^{k-1}$ and R_2 is of $2^{k-1} \times 2^k$, so that C is entirely contained in either R_1 or R_2, or in both.*

Proof. Assume that there are no such blocks R_1 and R_2 that entirely contain C. Then, C will be outside of R_1 either in X or in Y direction. Since the length of R_1 in Y direction is 2^k, which is equal to the size of C, it is only possible for C to be outside of R_1 in X direction. Hence $C_X > 2^{k-1}$, where the C_X is the dimension of C along X direction. Since C is not entirely contained in R_2 , based on the same reasoning above, it can be obtained that $C_Y > 2^{k-1}$, where C_Y is the dimension of C along Y direction. Hence, the assumption made at the beginning of this proof lead to that the size of cluster C, $Size(C)$, satisfies the following:

$$Size(C) \geq C_X + C_Y - 1 \geq 2 \times (2^{k-1} + 1) - 1 = 2^k + 1$$

This contradicts the given condition that C is of size 2^k. Hence, the lemma is proved. □

Theorem 2.3.3. *Consider a 2-D array, A, of $2^n \times 2^n$, where $n = 2p - 1$ and integer $p \geq 1$, obtained by applying the successive packing technique. Then, any arbitrarily-shaped error burst of size $t = 2^p$ in A can be spread in the de-interleaved array so that each element in the error burst falls into a distinct block of size 2^{2n-p}.*

Meaning of Theorem 2.3.3: This theorem indicates that if a distinct code symbol is assigned to each element in the blocks of size $t = 2^{3p-2}$ (refer to Definition 2.3.6) and all the code symbols associated with a block form a distinct codeword, then the SP technique can correct the arbitrarily-shaped error burst of size $t = 2^p$ with one-random-error-correction code, provided the code is available. That is, the SP technique achieves the same performance as that achieved by the technique in [21].

Proof. From Lemma 2.3.3, it is derived that any error bursts having a size of 2^p in a 2-D array of $2^n \times 2^n$, where $p \leq n$, is entirely contained in either a rectangular block R_1 and/or a rectangular block R_2 , where R_1 is of $2^p \times 2^{p-1}$ and R_2 is of $2^{p-1} \times 2^p$. Applying Theorem 2.3.2 at this point and noticing that the corresponding interleaving degree is 2^{2p-1} lead to the completion of the proof of Theorem 2.3.3. □

Comment 3: Needless to say that there is certain constraint with the SP technique. That is, it is assumed that given 2-D arrays are of $2^n \times 2^n$, and the error bursts considered are of $2^k \times 2^k$ with $0 \leq k \leq n-1$, or $2^k \times 2^{k+1}$ or $2^{k+1} \times 2^k$ with $0 \leq k \leq n-1$. It is noted that, however, up to now most of the 2-D burst error-correction codes (without involving interleaving) [17], [18] are dealing with error bursts of rectangular shapes. Furthermore, the assumption made with the SP is similar to that made in the development of the Fast Fourier Transform (FFT) technique, i.e., both are based on integer powers of 2 (base-2). Therefore, the assumption is practical and reasonable.

In summary, the SP approach does provide an alternative way for 2-D interleaving. For a given 2-D array of $2^n \times 2^n$, it can be applied once, and is optimal for the set of error bursts having different sizes defined in Theorem 2.3.2. In addition, for the case of arbitrarily-shaped error bursts having a size of an integer power of 2 to which both the SP technique and the technique in [21] can be applied, the SP approach can also spread and correct with some one-random-error-correction code arbitrarily-shaped error bursts with the same lower bound obtained by the approach in [21]. It is noticed that there are square 2-D arrays of some dimensions to which the approach in [21] cannot be applied while the SP can be applied. This is also true for the SP technique. That is, there are square arrays of some dimensions to which not both techniques can be applied.

2.4 Summary

In this chapter, the characteristics of M-D interleaving are studied and the optimality is discussed. A new 2-D interleaving method, called successive packing (SP) is then proposed to combat 2-D error bursts. The proposed SP method can spread any error burst of $2^k \times 2^k$ (with $1 \leq k \leq n-1$), $2^k \times 2^{k+1}$ (with $0 \leq k \leq n-1$), and $2^{k+1} \times 2^k$ (with $0 \leq k \leq n-1$) to different code blocks. Thus, the simple error correction code which is optimal for independent channel can be used to correct this kinds of error bursts. It needs to be implemented only once for a given 2-D array and is thereafter optimal for the set of error bursts having different sizes. A performance comparison between the proposed method and some existing techniques is given.

CHAPTER 3

SUCCESSIVE PACKING FOR MULTI-DIMENSIONAL INTERLEAVING

In Chapter 2, a novel method, called successive packing (SP), is proposed for 2-D interleaving [29]. The optimal performance of SP on square arrays of $2^n \times 2^n$ is claimed and proved. However, the analysis and application of SP is restricted on square arrays of $2^n \times 2^n$. In this chapter, the investigation is extended to the Multi-dimensional (M-D) array.

This chapter is organized as follows. In Section 3.1, some prerequisite definition are introduced, the concept of basis interleaving array is then presented for M-D interleaving. The corresponding method is proposed to realize it optimaly in 2-D case. Next, in Section 3.2, the combination of SP technique and basis interleaving array is used to realize M-D interleaving. It can work well in a set of error bursts where the size of 2-D arrays is given and the size of error bursts is not known in advance. Finally, this research is summarized in Section 3.3.

3.1 Basis Interleaved Array

To facilitate the description of the proposed concept, basis interleaving array, some definitions are first provided. The same procedure as Chapter 2 is followed. Namely, in what follows the treatment is presented with respect to one-random-error-correction codes for the sake of brevity of the discussion unless otherwise stated. The results can be straightforwardly extended to r-random-error-correction codes with $r > 1$.

Definition 3.1.1. *In M-D arrays, the distance between any two elements is the shortest length of all possible path, where path consists of a series of neighbour elements that connect this two elements.*

According to the above definition, the distance of two elements (x_1, y_1) and (x_2, y_2) in 2-D array is $|x_1 - x_2| + |y_1 - y_2|$. Interleaving generally means mixing up code symbols so that each element in an error burst can be spread into a different codeword (with respect to one-random-error-correction codes). Therefore, if any two elements within a distinct codeword in the interleaving M-D array are separated away from one another in the de-interleaving array such that the distance obtains the maximum, then a big error burst can be corrected.

Definition 3.1.2. *Let A be a M-D array of $m_0 \times m_1 \times \cdots \times m_{M-1}$. Each element of A, $s_{i_0,i_1,\cdots,i_{M-1}}$, is reindexed as s_k with k being a function of i_0 to i_{M-1} (For instance, for 2-D array, $k = m_1 \times i_0 + i_1$.). Let $N = m_0 \times m_1 \times \cdots \times m_{M-1}$. After having 1-D indexing, consider a division of A into L blocks with $1 \leq L \leq N$. That is, each block thus generated contains $K = N/L$ elements. The first block contains elements having the subscripts from 0 to $K - 1$. In general, for any element having an index k satisfying $K(d-1) \leq k < Kd$, $1 \leq d \leq L$, this element is said to belong to the dth block. The element having index $K(d-1)$ is referred to as the beginning element of the dth block. All the elements belonging to the same block are referred to as the K-equivalent elements.*

According to Definition 3.1.2, s_{2l}, s_{2l+1} are 2-equivalent elements; s_{3l}, s_{3l+1}, s_{3l+2} are 3-equivalent elements for any integer $l \geq 0$. It is obvious that K_1-equivalent elements are also K_2-equivalent elements if K_2/K_1 is integer. Let one block be a codeword with length K, then all elements of a distinct codeword is K-equivalent each other. Hence, the objective of effective interleaving is transfered to the problem of maximizing the minimum distance between any two K-equivalent elements. In Definition 3.1.2, the size of the M-D array is $m_0 \times m_1 \times \cdots \times m_{M-1}$. If m_0 to m_{M-1} are prime number, the corresponding M-D array can only contains m_n $(n < M)$ codewords or the multiple of several of them. Motivated by this observation, a concept of *basis interleaving array* is proposed to realize the effective interleaving.

3.1.1 Square 2-D Basis Array

Definition 3.1.3. *Given an interleaving array B of $m \times m$, where m are prime number. If the minimum distance between any two m-equivalent elements obtain the maximum, then this array is called as basis interleaving array.*

It is obvious from definition 3.1.3 that there are basis array of 2×2, 3×3, 5×5, $\cdots$ for square array. In last chapter, an optimal interleaving technique based on the successive packing of a 2×2 array is presented. In fact, that 2×2 array is a basis interleaving array.

Theorem 3.1.1. *2-D array*

$$\begin{bmatrix} S_0 & S_2 \\ S_3 & S_1 \end{bmatrix} \tag{3.1}$$

is a basis interleaving array according to definition 3.1.3.

Proof. In a 2×2 array, the distance from one corner to its opposite corner is the maximum distance between any two elements. It is obvious that this distance equal to 2. For the two 2-equivalent elements pairs (s_0, s_1) and (s_2, s_3), it is seen the distance between s_0 and s_1, the distance between s_2 and s_3 obtain 2 at the same time. Thus, Theorem 3.1 is proved. □

In order to find the basis interleaving array, it is necessary to know the upper bound of the minimum distance. For a square array $m \times m$, each element has $m-1$ m-equivalent elements. Thus, it needs to constitute a 2-D sphere with size m center around each of the m-equivalent elements. The m spheres should be able to tile to a $m \times m$ array without overlapping. If this sphere exists, then the maximum radius of it is the upper bound of the minimum distance. This problem is first approached in [25] for m is odd. Then this idea is extended to m is even in [21]. It has been proved that if m is even and $m = t^2/2$, then the upper bound of the radius is t; if m is odd and $m = (t^2+1)/2$, then the upper bound is also t [21].

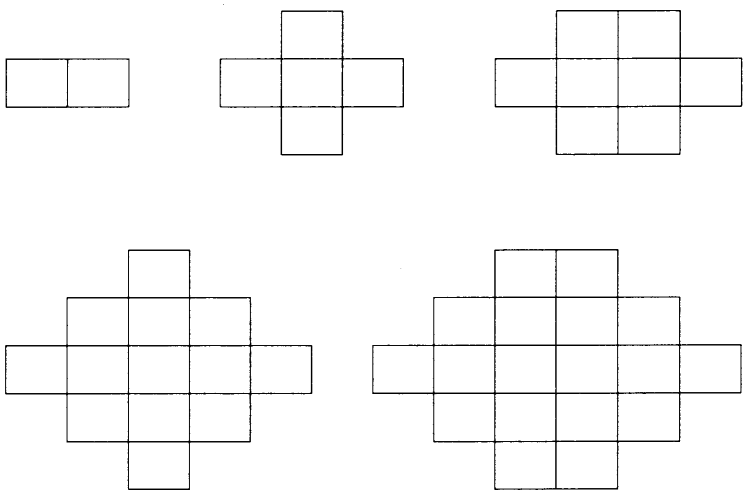


Figure 3.1 Typical 2-D sphere with size 2, 5, 8, 13, 18.

Some typical 2-D spheres are shown in Figure 3.1. Notice that the size of the sphere is not consecutive. For most of the interger m such as $3, 4, 6, 7, 9, 10, 11, \cdots$, there are no size m sphere exist. Thus, the upper bound of the minimum distance is the radius of the largest sphere with size less than m. According to this observation, the upper bound is 2 for $2 \leq m \leq 4$; the upper bound is 3 for $5 \leq m \leq 7$; upper bound is 4 for $8 \leq m \leq 12$ and etc.

In the following, a method to construct the square basis interleaving array with different size is presented. Namely, the constructed array satisfy the requirements of Definition 3.1.3

Procedure 3.1.1. *Let A be a 2-D array of $m \times m$, $m \geq 2$, and let d_r be the upper bound of the distance. The coordinates (i, j) of each element are labeled in the array toroidally on m, namely, coordinate $i = i(mod)m$ if $i \geq m$, coordinate $j = j(mod)m$ if $j \geq m$. The elements of first row are first attributed as $s_0, s_m, s_{2m}, \cdots, s_{m(m-1)}$, which are in the location $(0,0), (0,1), \cdots, (0, m-1)$ respectively. Then, let $X = 1$ and $Y = d_r - 1$. For each element with location (i, j), add 1 to the subscript of this element and put it in the location (i+X, j+Y). For example, s_1 is put in the location (X, Y). Repeat this procedure until all of the positions are occupied.*

Example 3.1.1.1: Consider the case of 2×2 array. According to [21], it can be obtained $d_r = 2$. Thus, $X = 1, Y = 1$. Using the above procedure with $X = 1, Y = 1$, The basis interleaving array is constructed as in Figure 3.2, which is exactly the same as the 2×2 basis array above.

S_0	S_2
S_3	S_1

Figure 3.2 2×2 basis interleaving array.

Example 3.1.1.2: Consider the case of 3×3 array. According to [21], $d_r = 2$. Thus, $X = 1, Y = 1$. Using the above procedure with $X = 1, Y = 1$, the array is constructed as in Figure 3.3.

S_0	S_3	S_6
S_7	S_1	S_4
S_5	S_8	S_2

Figure 3.3 3×3 basis interleaving array.

Example 3.1.1.3: Consider the case of 5×5 array. According to [21], $d_r = 3$. Thus, $X = 1, Y = 2$. Using the above procedure with $X = 1, Y = 2$, the array is constructed as in Figure 3.4.

S_0	S_5	S_{10}	S_{15}	S_{20}
S_{16}	S_{21}	S_1	S_6	S_{11}
S_7	S_{12}	S_{17}	S_{22}	S_2
S_{23}	S_3	S_8	S_{13}	S_{18}
S_{14}	S_{19}	S_{24}	S_4	S_9

Figure 3.4 5×5 basis interleaving array.

Theorem 3.1.2. *For any prime number m, the square array constructed by procedure 3.1.1 is a basis interleaving array.*

Proof. According to procedure 3.1.1, the positions for elements $s_0, s_m, s_{2m}, \cdots, s_{m(m-1)}$ are generated first, then the positions of their m-equivalent elements are generated respectively. It can be seen that the corresponding distance of two co-positional m-equivalent elements in each of the m-equivalent set is same. For eaxmple, the distance between s_0 and s_k is equal to the distance betweens_{2m} and s_{2m+k}, where $k < m$. Therefore, if Theorem 3.1.2 can be proved for the case of m-equivalent set beginning with s_0, then theorem 3.1.2 is proved. Based on the same reasoning, if it can be proved that the distance between s_0 and any its m-equivalent elements is greater or equal to d_r, then it holds for any s_k with $0 < k < m$. Here the case that d_r is odd is proved first, then the proof is extended to the case that d_r is even.

Due to toroidally labeling of the coordinates, the coordinates of s_k is (kX, kY $(mod)m$). The distance between s_0 and s_k is $kX + kY(mod)m$. Thus, the problem is transfered to prove the following un-equation:

$$kX + kY(mod)m \geq d_r \tag{3.2}$$

If $kY < m$, $kY(mod)m = kY$. Thus, $kX+kY(mod)m = k(X+Y)$. Since $X+Y = d_r$, it is obvious that $k(X + Y) > d_r$. Now consider the case that $kY > m$, kY can be described as follows

$$kY = l_1 m + l_2.$$

Using the above equation, it can be obtained that

$$k = \frac{l_1 m + l_2}{Y}.$$

Since $X = 1$, $Y = d_r - 1$, then

$$kX + kY(mod)m = \frac{l_1 m + l_2}{d_r - 1} + l_2.$$

Hence, the unequation 3.2 becomes

$$\frac{l_1 m + l_2}{d_r - 1} + l_2 \geq d_r \tag{3.3}$$

Calculate the above, it is obtained that

$$l_1 m \geq d_r^2 - (l_2 + 1)d_r > d_r^2 - (l_2 + 1)d_r - l_2.$$

According to the result in [21], $m \geq \frac{d_r^2+1}{2}$. So if

$$l_1 \frac{d_r^2 + 1}{2} > d_r^2 - (l_2 + 1)d_r - l_2$$

holds, then un-equation 3.3 holds. For the above un-equation, obviously it holds for $l_1 > 1$.If $l_1 = 1$, then

$$\frac{d_r^2 + 1}{2} \geq d_r^2 - (l_2 + 1)d_r - l_2.$$

Thus, the value of l_2 should satisfy the following condition

$$l_2 \geq \frac{d_r - 3}{2} + \frac{1}{d_r + 1} \geq \frac{d_r - 2}{2}.$$

In order to find the range of l_2 when $l_1 = 1$, decompose m as

$$\begin{aligned} m &= \frac{d_r^2 + 1}{2} \\ &= (d_r - 1)\frac{d_r + 1}{2} + 1 \end{aligned} \tag{3.4}$$

Now let $k = \frac{d_r+1}{2} + 1$, since $Y = d_r - 1$, it is derived that

$$\begin{aligned} kY &= (d_r - 1)\frac{d_r + 1}{2} + 1 + (d_r - 2) \\ &= m + (d_r - 2) \end{aligned} \tag{3.5}$$

Hence, $l_2 = d_r - 2$. According to the above procedure, $d_r - 2$ is the minimum of l_2 when $l_1 = 1$. Thus, Theorem is proved for d_r is odd. For d_r is even, the same reasoning is used to prove. Therefore, Theorem 3.1.2 is proved. □

In Procedure 3.1.1, a technique is proposed to generate the basis interleaving array. Next, this method is generalized to any $m \times m$ array such that the minimum distance between any two m-equivalent elements obtains the maximum.

Procedure 3.1.2. *Let A be a 2-D array with $m \times m$, $m \geq 2$, and let d_r be the upper bound of the minimum distance. The coordinates of the array are labeled toroidally on m, namely, coordinate $i = i(mod)m$ if $i \geq m$, coordinate $j = j(mod)m$ if $j \geq m$, the elements of first row are first attributed as $s_0, s_m, s_{2m}, \cdots, s_{m(m-1)}$, which are in the location $(0,0), (0,1), \cdots, (0, m-1)$. Then let $X = 1$ and Y satisfy the condition of $d_r - 1 \leq Y \leq d_r + 1$, and Y is relative primal to m. For each element with location (i, j), add 1 to the subscript of the element and put it in the location (i+X, i+Y). For example, s_1 is put in the location (X, Y). Repeat this procedure until all of the positions are occupied.*

Theorem 3.1.3. *For any interger $m > 1$, the square array constructed by procedure 3.1.2 obtains the maximum in the sense of minimum distance between any two m-equivalent elements.*

The proof of Theorem 3.1.3 can be completed by using the same procedure of proving Theorem 3.1.2. Notice that *Construction* 2.1 in [21] is a special case of Procedure 3.1.2, where $Y = d_r$ for $m = \frac{d_r^2+1}{2}$ and $Y = d_r + 1$ for $m = \frac{d_r^2}{2}$.

3.1.2 Rectangular Basis Interleaving Array

In Section 3.1.1, a method has been proposed for generating square basis array and proved its validity based on the well defined theoretical results. For the rectangular basis array, the following result exists.

Theorem 3.1.4. *Given a $m \times n$ rectangular basis array. If $m < n$, the upper bound of the minimum distance between any two n-equivalent elements is the same as the minimum distance of any two m-equivalent elements in $m \times m$ basis array. If $m > n$, the upper bound of the minimum distance between any two m-equivalent elements is the same as the minimum distance of any two n-equivalent elements in $n \times n$ basis array.*

Proof. To prove theorem 3.1.4, the upper bound of its minimum distance is first proved cannot greater than the corresponding square basis array. Then a method is presented to show that the equality can be obtained.

If the minimum distance is greater than the corresponding square basis array, for instance, if $n > m$, the minimum distance of any two n-equivalent elements greater than the minimum distance of $m \times m$ basis array. Truncating the $m \times n$ array to $m \times m$ array, according to the above assumption, the new obtained $m \times m$ array will obtain the minimum distance larger than the $m \times m$ basis interleaving array, which contradicts the definition of basis interleaving array. Hence, the minimum distance of the n-equivalent elements in the $m \times n$ array cannot be larger than the corresponding $m \times m$ square basis interleaving array. To obtain the same minimum distance as the square basis interleaving array, the Procedure 3.1.1 can be changed a little bit to generate the rectangular basis interleaving array.

Procedure 3.1.3. *Let A be a 2-D array with $m \times n$, let d_r be the upper bound of the distance for the corresponding square interleaving array. If $m < n$, the row coordinates of the array are labeled toroidally on m, namely, coordinate $i = i(mod)m$ if $i \geq m$, column coordinate $j = j$. The elements of first row are first attributed as $s_0, s_n, s_{2n}, \cdots, s_{(m-1)n}$, which are in the location $(0,0), (0,1), \cdots, (0, m-1)$. Then let $X = d_r - 1$, $Y = 1$. For each element with location (i, j), add 1 to the subscript of the element and put it in the location (i+X, j+Y). For example, s_1 is put in the location (X, Y). Repeat this procedure until all of the positions are occupied. If $m > n$, the column coordinates of the array are labeled toroidally on n, namely, column coordinate $j = j(mod)n$ if $j \geq n$, row coordinate $i = i$. The elements of first column are first attributed as $s_0, s_m, s_{2m}, \cdots, s_{(n-1)m}$, which are in the location $(0,0), (1,0), \cdots, (n-1, 0)$. Then let $X = 1$, $Y = d_r - 1$. For each element with location (i, j), add 1 to the subscript of the element and put it in the location (i+X, j+Y). Repeat this procedure until all of the positions are occupied.*

According to Procedure 3.1.3 and based on the same reasoning of the proof for Theorem 3.1.2, it is easy to see that the same minimum distance as the corresponding square array is obtained. □

3.1.3 3-D Basis Interleaving Array

In this section, instead of doing thoroughly investigation as previous section, it is aimed to introducing how to extent the previous results to higher dimension by showing some examples of extending 2-D results to 3-D results.

Definition 3.1.4. *Given an interleaving array B of $l \times m \times n$, where l, m, n are prime number and $l \leq m \leq n$. If the minimum distance between any two mn-equivalent elements obtain the maximum, then this array is called as basis interleaving array.*

It is noted l, m, n can exchange their positions in the above definition. For a square 3-D array $m \times m \times m$ with minimum distance d_r, it has been proved that m is bounded by

$$m \geq \frac{d_r^3 + 2d_r}{6}, \quad \text{for } d_r \text{ even},$$
$$m \geq \frac{d_r^3 + 5d_r}{6}, \quad \text{for } d_r \text{ odd},$$

where m is the size of 3-D sphere [21]. This result is further extended to M-D array in [21]. However, this bound cannot be always to achieve for $M \geq 3$. The detail analysis refer to [25, 21]. Here, an example of $2 \times 2 \times 2$ 3-D basis interleaving array is presented as in Figure 3.5, which is useful in the investigation in the following sections.

3.2 Successive Packing of Basis Interleaving Array

It is noted that an initial idea of successive packing for interleaving was presented in [28, 29]. In these works, the authors focus on the interleaving of array of $2^n \times 2^n$. Whether it can applied to M-D array with arbitrary size has not been investigated.

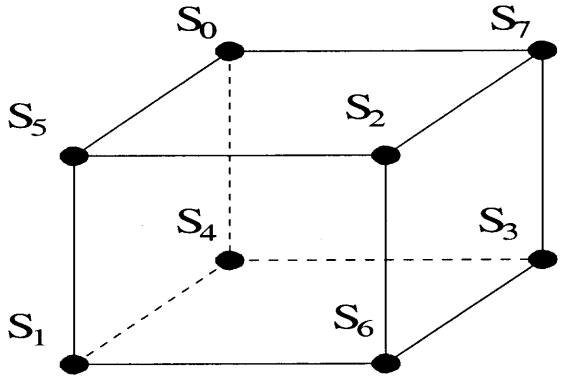


Figure 3.5 3-D $2 \times 2 \times 2$ basis interleaving array.

Also, it is not clear if its optimal performance can also hold for rectangular 2-D array. In this section, the M-D interleaving technique based on the successive packing of basis interleaving array is first presented. Then its performance for spreading error burst is analyzed. Its optimality is discussed and proved.

3.2.1 Successive Packing

Now the proposed SP technique in M-D case is presented in a general and compact way.

Procedure 3.2.1. *The M-D interleaving using the successive packing proceeds as follows. Consider an M-D basis interleaving array of* $m_0 \times m_1 \times \cdots \times m_{M-2} \times m_{M-1}$. *The interleaving array is the original basis interleaving array itself. When* $m_0 = m_1 \cdots = m_{M-1} = 1$, *it is*

$$S_1 = [s_0] \tag{3.6}$$

where s_0 *represents the element in the array, and* S_1 *the array. It is noted that the subscript in the notation* S_1 *representing the total number of elements in the interleaving array. Let* $N = m_0 \times m_1 \times \cdots \times m_{M-2} \times m_{M-1}$, *then the above interleaving array is denoted by* S_N. *Given* S_N, *the interleaving array of* S_{N^2} *can be generated by transfering each element* s_i *in* S_N *to a M-D array according to the operation* $N \times S_N + i$ *(the meaning of this operation can be seen in the follows). This packing*

procedure is carried out successively to generate S_{N^K} by transfering each element s_i in $S_{N^{K-1}}$ to a M-D array according to the operation $N \times S_{N^{K-1}} + i$.

In the above procedure, the operation $N \times S_N + i$ is the key point. Generalizing what presented in last chapter, operation $N \times S_N + i$ generates a M-D array with the same dimensionality as S_N. Furthermore, each element in $N \times S_N + i$ is indexed in such a way that its subscript equals to the N times of that of the corresponding elements in S_N plus i. The corresponding element means the element occupying the same position in the M-D array. Next, some examples are used to present it more clearly.

Example 3.2.1: Given a 1-D basis array $S_3 = \{s_0, s_1, s_2\}$, the interleaving array is $S_9 = \{s_0, s_3, s_6, s_1, s_4, s_7, s_2, s_5, s_8\}$.

Example 3.2.2: Given a 3×3 basis array as in Figure 3.3, the 9×9 interleaving array is generated as in Figure 3.6.

S_0	S_{27}	S_{54}	S_3	S_{30}	S_{57}	S_6	S_{33}	S_{60}
S_{63}	S_9	S_{36}	S_{66}	S_{12}	S_{39}	S_{69}	S_{15}	S_{42}
S_{45}	S_{72}	S_{18}	S_{48}	S_{75}	S_{21}	S_{51}	S_{78}	S_{24}
S_7	S_{34}	S_6	S_1	S_{28}	S_{55}	S_4	S_{31}	S_{58}
S_{70}	S_{16}	S_4	S_{64}	S_{10}	S_{37}	S_{67}	S_{13}	S_{40}
S_{52}	S_{79}	S_{25}	S_{46}	S_{73}	S_{19}	S_{49}	S_{76}	S_{22}
S_5	S_{32}	S_{59}	S_8	S_{35}	S_{62}	S_2	S_{29}	S_{56}
S_{68}	S_{14}	S_{41}	S_{71}	S_{17}	S_{44}	S_{65}	S_{11}	S_{38}
S_{50}	S_{77}	S_{23}	S_{53}	S_{80}	S_{26}	S_{47}	S_{74}	S_{20}

Figure 3.6 Successive Packing generated 9×9 interleaving array.

Example 3.2.3: Given a $2 \times 2 \times 2$ basis array as in Figure 3.5, the $4 \times 4 \times 4$ interleaving array is generated as in Figure 3.7. Where the left is the $2 \times 2 \times 2$ array obtained by $8 \times S_8 + 5$.

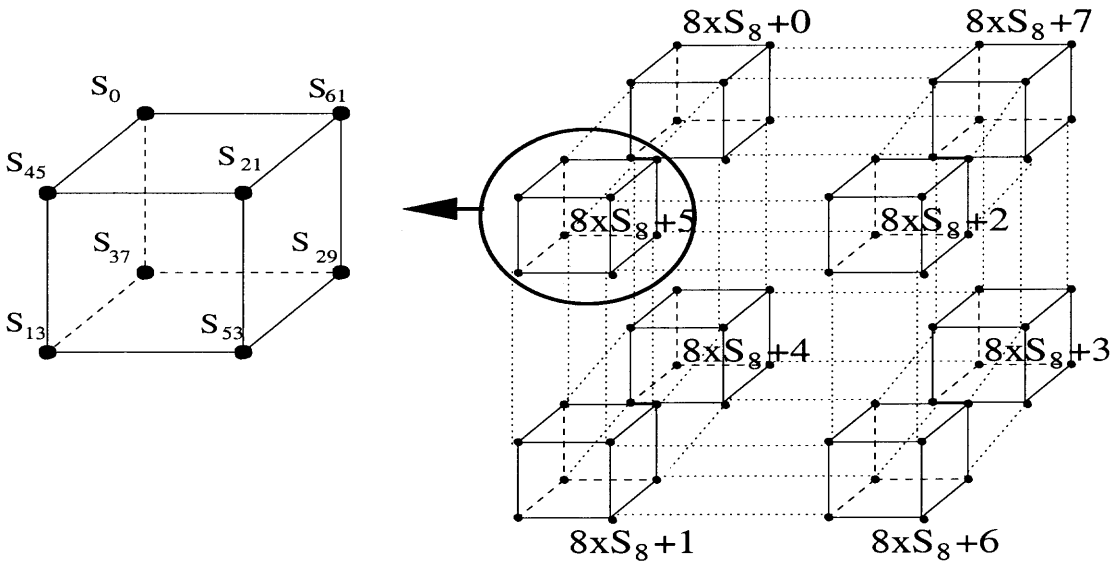


Figure 3.7 Successive Packing generated $4 \times 4 \times 4$ interleaving array.

To generate the interleaving array with arbitrary size, the successive packing is used on the combination of different basis interleaving array. For instance, given basis interleaving array S_N and S_2, the interleaving array S_{2N} can be generated by $\{S_N \times 2 + 0, S_N \times 2 + 1\}$.

Example 3.2.4: Given 2×2 basis array as in Figure 3.2 and 3×3 basis array as in Figure 3.3, the 6×6 interleaving array is generated as in Figure 3.8.

3.2.2 Performance Analysis

Prior to presenting the performance of the SP based M-D interleaving technique, the definition in Chapter 2 is firstly extended as follows.

Definition 3.2.1. *Consider two bursts, B_1 and B_2, in an interleaving M-D array. If these two bursts have the same size and shape, and each element in a burst (e.g.,B_1) is either an element of another burst(e.g.,B_2) or a K-equivalent element of an element of another burst (B_2), and vice versa, then these two bursts, B_1 and B_2, are called the K-equivalent bursts.*

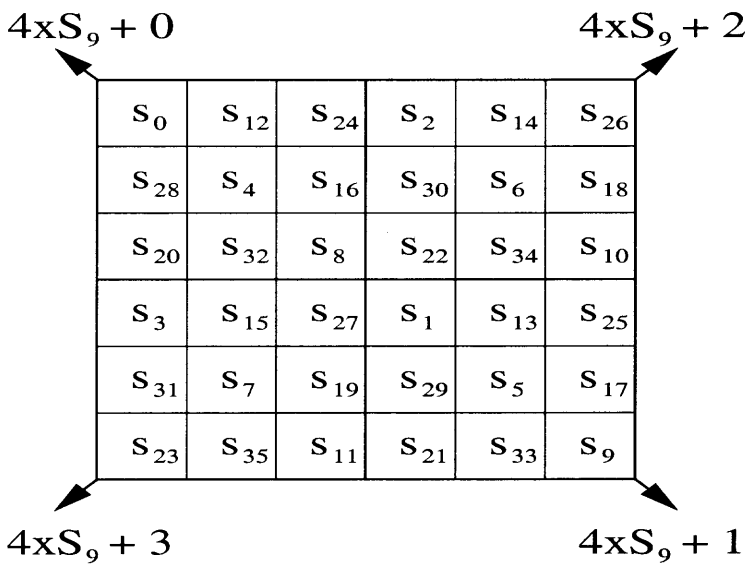


Figure 3.8 Successive Packing generated 6×6 interleaving array.

In the remaining part, when discussing the error burst correction, each set of equivalent elements defined in Definition 3.1.2 may be considered as an M-D codeword. This implies that a codeword consists of a set of consecutive code symbols. The new way is necessary since it need to discriminate code symbols within a codeword in the following discussion of the SP technique for M-D interleaving. An error burst (in the interleaving array) is said to be spread and can be corrected with one-random-error-correction codes if each element in the burst has been spread in the de-interleaving array into a distinct codeword. From this point of view, it is easy to see that given two equivalent bursts, if one has been interleaved, the other must have also been interleaved.

With the concept of *equivalent bursts*, the following results hold:

Lemma 3.2.1. *Let A be an 2-D array of $m_0^n \times m_1^n$ obtained by using the successive packing of basis interleaving array of $m_0 \times m_1$. Then all bursts of $m_0^k \times m_1^k$ in A with $k < n$ are K_1-equivalent to one another, where $K_1 = (m_0 \times m_1)^{n-k}$.*

The proof of this lemma is contained in Appendix B. Now it is the time to present the following theorem.

Theorem 3.2.1. *Consider a 2-D array A of* $m_0^n \times m_1^n$. *Then any burst of* $m_0^m \times m_1^m$ *with* $m \leq n$ *in the interleaving array, A, obtained by using the successive packing is spread in the de-interleaving array so that each element of the burst falls into a distinct block of size* $m_0^{n-m} \times m_1^{n-m}$.

Meaning of Theorem 3.2.1: Theorem 3.2.1 claims that in a 2-D array of $m_0^n \times m_1^n$, A, generated with the successive packing technique, any error burst of $m_0^m \times m_1^m$ with $m \leq n$ can be spread so that each element in the burst falls into a distinct block in the de-interleaving array, where the block size, K, is $m_0^{n-m} \times m_1^{n-m}$. This indicates that, if a distinct code symbol is assigned to each element in a block (refer to Definition 3.1.2) and all the code symbols associated with an individual K-equivalent class form a distinct codeword, then this technique guarantees that the burst error can be corrected with a one-random-error-correction code, provided the code is available. Furthermore, the interleaving degree equals to the size of the burst error, hence, minimizing the number of codewords required in an interleaving scheme. In other words, with the successive packing technique, the interleaving degree obtains the lower bound (the interleaving gain). In this sense, the successive packing interleaving technique is optimal(The meaning of the interleaving degree and interleaving gain refer to Chapter 2). It is noted that what presented in Chapter 2 is a special case of Theorem 3.2.1 with $m_0 = 2, m_1 = 2$.

The proof of this theorem is contained in Appendix B.

Based on Theorem 3.2.1 and its proof, it is conjectured that following results hold.

Corollary 3.2.1. *Consider an M-D array A of* $m_0^n \times m_1^n \times \cdots \times m_{M-1}^n$. *Then any burst of* $m_0^m \times m_1^m \times \cdots \times m_{M-1}^m$ *with* $m \leq n$ *in the interleaving array, A, obtained by using the successive packing is spread in the de-interleaving array so that each element of the burst falls into a distinct block of size* $m_0^{n-m} \times m_1^{n-m} \times \cdots \times m_{M-1}^{n-m}$.

In Section 3.2.1, it is proposed to generate arbitrary size interleaved array by combining different basis interleaving array. Here, it is first shown how to generate the square 2-D array of $2m^2 \times 2m^2$. Then its optimal performance is proved. The procedure can be described as follows:

Procedure 3.2.2.

- *Generate the $m \times m$ interleaving array according to the SP technique.*
- *Generate the $2m \times 2m$ interleaving array by*

$$S_{4m^2} = \begin{bmatrix} 4 \times S_{m^2} + 0 & 4 \times S_{m^2} + 2 \\ 4 \times S_{m^2} + 3 & 4 \times S_{m^2} + 1 \end{bmatrix} \tag{3.7}$$

- *Let $l_{i,j}$ denote the value of the subscript of the corresponding element in the 2-D interleaving $m \times m$ array S_{m^2}. The $2m^2 \times 2m^2$ interleaving array is generated by*

$$S_{4m^4} = \begin{bmatrix} m^2 \times S_{4m^2} + l_{0,0} & \cdots & m^2 \times S_{4m^2} + l_{0,m-1} \\ \cdots & & \\ m^2 \times S_{4m^2} + l_{m-1,0} & \cdots & m^2 \times S_{4m^2} + l_{m-1,m-1} \end{bmatrix} \tag{3.8}$$

Prior to presenting the results, the following lemma is needed.

Lemma 3.2.2. *Let C be a cluster of size $2m$ in 2-D array $m_1 \times m_1$, where $2m < m_1$. Then there must exist a rectangular block R_1 and/or a rectangular block R_2, where R_1 is of $2m \times m$ and R_2 is of $m \times 2m$, so that C is entirely contained in either R_1 or R_2, or in both.*

Proof. Assume that there are no such blocks R_1 and R_2 that entirely contain C. Then, C will be outside of R_1 either in X or in Y direction. Since the length of R_1 in Y direction is $2m$, which is equal to the size of C, it is only possible for C to be outside of R_1 in X direction. Hence, $C_X > m$, where the C_X is the dimension of C along X direction. Since C is not entirely contained in R_2 , based on the same reasoning above, it is obtained that $C_Y > m$, where C_Y is the dimension of C along

Y direction. Hence, the assumption made at the beginning of this proof lead to that the size of cluster C, $Size(C)$, satisfies the following:

$$Size(C) \geq C_X + C_Y - 1 \geq 2 \times (m+1) - 1 = 2m + 1$$

This contradicts the given condition that C is of size $2m$. Hence, the lemma is proved. □

Theorem 3.2.2. *The 2-D interleaving array generated by procedure 3.2.2 is optimal in the sense that it can spread arbitrary burst of size* $2m$ *to distinct codeword of size* $2m^2$.

Proof. According to Lemma 3.2.1 and Theorem 3.2.1, it is easy to see that any $2m \times 2m$ bursts within the generated $2m^2 \times 2m^2$ array are m^2-equivalent bursts each other, any $m \times 2m$ or $2m \times m$ bursts within the generated $2m^2 \times 2m^2$ array are $2m^2$-equivalent bursts each other. Thus, any bursts with size $m \times 2m$ or $2m \times m$ can be spread into distinct blocks with size $2m^2$. According to Lemma 3.2.2, arbitrary burst of size $2m$ is either contained in a $m \times 2m$ or a $2m \times m$ burst. Thus, Theorem 3.2.2 is proved. □

This theorem indicates that if a distinct code symbol is assigned to each element in the blocks of size $2m^2$ (refer to Definition 3.1.2) and all the code symbols associated with a block form a distinct codeword, then the SP technique can correct the arbitrarily-shaped error burst of size $2m$ with one-random-error-correction code, provided the code is available.

It is seen that what has been presented in Chapter 2 can be attributed as a special case of this chapter. Needless to say that the proposed technique in this chapter cannot guarantee the optimal preformance to any rectangle array. For example, the upper bound of the minimum distance cannot guranteed to achieve for the larger 2-D array (multiple basis interleaving array) with size $(2m+1) \times (2m+1)$.

It is noted that, however, the proposed interleaving method is optimal to a large set of bursts. It is versatile in this sense.

3.3 Summary

In this chapter, how to develop effective M-D interleaving technique is investigated. A novel concept, basis interleaving array is first presented. Then the combination of it and the SP technique is proposed to combat M-D error bursts. In doing so, an effective method for M-D interleaving is obtained. For a given 2-D array of $m_0^n \times m_1^n$, it can be applied once, and is optimal for a set of error bursts having different sizes defined in Theorem 3.2.1. In addition, for the case of arbitrarily-shaped error bursts having a size of $2m$ to which both the SP technique and the technique in [21] can be applied, the SP approach can also spread and correct with some one-random-error-correction code arbitrarily-shaped error bursts with the same lower bound obtained by the approach in [21]. For the basis interleaving array, a method which is proved to be optimal in 2-D case is proposed. For $M > 2$ case, the optimality cannot guaranteed [25, 21]. Part of this research work has been published in [30].

CHAPTER 4

SUCCESSIVE PACKING FOR TURBO CODES INTERLEAVER DESIGN

In last two chapters, a novel successive packing (SP) based M-D interleaving method has been presented. Its theoretical meaning and the optimality in the sense of burst error spreading have been discussed. In this chapter, the research is focus on how to utilize this method to various applications. Specifically, how to apply the SP method to turbo codes interleaver design.

Since Berrou *et al.* [31] proposed turbo codes in 1993, great efforts have been made to investigate the methods with similar principle to achieve near Shannon capacity. The basic idea of turbo codes is to break up decoding of a fairly complex and long code into steps while the transfer of probabilities or *soft* information between the decoding steps guarantees alomost no loss of information [32]. In doing so, the big improvement is obtained. To explain the superior performance of turbo codes, McEliece *et al.* connect it with Pearl's belief propagation algorithm [33], and Moher *et al.* [34] generalize it to iterative techniques for minimizing cross-entropy. Figure 4.1 shows a typical structure of the turbo code encoder. It consists of two systematic recursive convolutional encoders connected in parallel, with an interleaver preceding the second one (lower one). Among the components of various turbo codes, interleaver played an important role. Its design influences the capability of the corresponding turbo codec greatly. To acheive the superior performance of turbo codes, some criterions are used in the design of interleaver. Some optimization techniques which combine several criterions have also been proposed [35], [36]. Among the proposed criterions, improving the distance spectrum properties of the code has been adopted by most of the prior *state-of-art* methods. Besides it, iterative decoding suitability (IDS) [37] criterion was proposed to measure the effectiveness of the

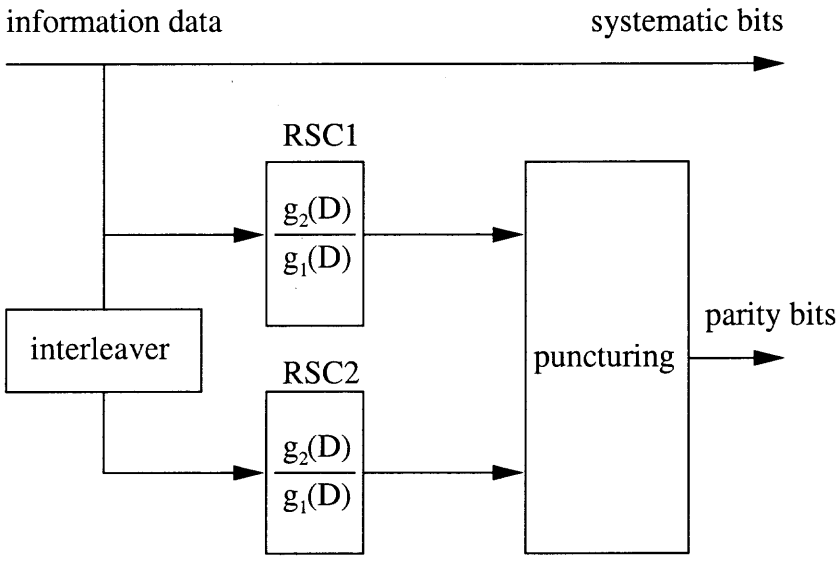


Figure 4.1 Typical structure of turbo encoder.

iterative decoding algorithm. It has been shown that a large pseudo-random interleaver can obatin the optimal results. This is because a low-weight sequence in one branch is tranformed to a high-weight sequence with very high probability by using it, the obviously *bad* interleavers like the replicating interleaver are avoided [38]. However, there are some drawbacks in random generated interleavers. One drawback is the computational cost owing to the exhaustive search based on cost functions. In addition, once the search is completed, the obtained interleaving result must be stored somewhere. Moreover, the exhaustive search has to be re-implemented in order to generate the interleaved sequence with different block length. In reality, it is hoped that the interleaver design can support different block sizes from several hundreds of bits to tens of thousands of bits according to variable input data requirements. Thus, it is important to have a turbo interleaver design which can adapt to various block size. Another drawback is their worse performance with short block length. In many applications such as multimedia transmission, the large delay cannot be tolerated. Hence, one needs to find an effective short interleaver design scheme.

To overcome the above drawbacks of pseudo-random algorithms, various deterministic interleaver design algorithms have been proposed [39, 40, 41, 42] etc. Among them, Takeshita *et al.* [42] have shown that their proposed deterministic interleavers have nearly the same statistical distribution as a random interleaver and perform as well as or better than the average of a set of random interleavers. In order to achieve the interleaver which can adapt to different block length, Eroz *et al* [41]. propose a prunable interleaver family and has been adopted for cdma2000 [43]. In their method, an interleaver is first generated corresponding to big block length. Then it can be pruned according to the actual requirements. In this chapter, the research focuses on the deterministic interleaver design and aimed to obtain an effective prunable interleaver.

This chapter is organized as follows. In Section 4.1, some characteristics of deterministic interleaver design are discussed. An interleaver design scheme that obtain the upper bound based on the principle of S-random interleaver design is proposed in Section 4.2, its pereformance is analyzed. It is further shown that the proposed scheme can be optimized according to the specific requirements. Experimental results demonstrate the effectiveness of the proposed technique in Section 4.3. The conclusions of the work are presented in Section 4.4.

4.1 Deterministic Interleaver Design

In the traditional turbo code, pseudo-random interleaver design is adopted. The principle underlying the pseudo-random design is to remove some aparent order such that low-weight parity output sequences would not occcur on both upper and lower constitunent encoders at the same time. Among them, S-random interleaver is considered to be one of the best turbo interleavers available in the literature so far. The idea behind the S-random interleaver is to spread the position of the elements within a window as far as possible, such that any two elements within a window of size

S will not locate in a window of size S at the same time. The objective of this design is to make S as large as possible. To realize the S-random interleaver, exhaustive search is often be used. The computational expense is high. However, even using exhaustive search, such pseudo random design cannot gurantee the realization of this objective due to the difficuly of effective analysis. On the other hand, if the deterministic interleaver design is used, the requirements of S-random interleaver can be obtained in very low computational cost. That is because it is easy to find some input patterns that result low-weight parity output for the fixed turbo code architecture. The interleaver can then be optimized to prevent the happening of them. In the following, some important results of deterministic interleaver design will be presented and proved.

Theorem 4.1.1. *For any interleaving scheme that meet the following requirement: if* $|i-j| < \alpha$, *then* $|\pi(i) - \pi(j)| \geq \alpha$. *The maximum value of* α *is less and equal to* $\sqrt{N}$, *where* i, j *denote the positions of two information elements;* $\pi(i)$, $\pi(j)$ *denote their positions after interleaving;* N *is the length of the interleaved sequence.*

The proof of this theorem and some of its application can be found in [36]. For completeness the proof is repeated here.

Proof. Assume there exists an interleaving scheme with $\alpha > \sqrt{N}$. Let's select arbitrary α consecutive elements. Then the elements $\pi(k), \cdots, \pi(k+\alpha)$ have pairwise distance greater than $\sqrt{N}$, where $1 \leq k \leq N - \alpha$. If each element is considered as the "center" of a ball, then the balls with radius $\alpha/2$ cover the numbers $\{1, \cdots, \alpha^2\}$ completely. Since $\alpha^2 > N$, which is not possible. Thus, Theorem 4.1.1 is proved. □

Corollary 4.1.1. *For any interleaving scheme, if* $|i-j| < \alpha$, *then the maximum value of* $|\pi(i) - \pi(j)|$ *is less and equal to* $\lfloor \frac{N}{\alpha} \rfloor$, *where* N *is the length of the interleaved sequence.*

Proof. The proof of this corrolary can use the similar reasoning as above to complete. □

Theorem 4.1.1 provides the upper bound for the window size. Among the interleaver design so far in literature, block interleaver can be used to obtain the sub-optimal result. Block interleavers are usually defined by rectangular matrices of size $N = m \times n$ that are written row-wise and read column-wise [42]. It can be described by the form

$$\pi(i) = ni + \lfloor \frac{i}{m} \rfloor (mod N), \;\; 0 \leq i < N \,. \tag{4.1}$$

In order to analyze it, Takeshita *et al.* [42] linearize the floor function $\lfloor \rfloor$ in Eq. 4.1 to obtain a linear interleaver as

$$\pi(i) = ki + v(mod N), \;\; 0 \leq i < N \tag{4.2}$$

where k is a fixed integer relatively prime to N and v is a fixed integer. A similar deterministic interleaver design is also presented in [36].

It is noted that the upper bound cannot be obatined by the block interleaver or the linearization of it. Also, in block interleaver, n should be divided by N; in linear interleaver, k should be relatively prime to N. Thus, some interleaving choices are restricted. In the following, a slightly different interleaver design method will be presented. It can not only obtain the upper bound but also has no above restrictions.

Theorem 4.1.2. *Consider a sequence with length N. For any* $\alpha \leq \lfloor \frac{N}{2} \rfloor$ *and* N/α *is integer, there exists an interleaving scheme such that for any* $|i - j| < \frac{N}{\alpha}$, *and* $i \neq j$, *it can obtain that* $|\pi(i) - \pi(j)| \geq \alpha$.

Proof. Consider the following interleaving scheme.

1. Let $\pi(i) = \alpha i$, where $0 \leq i \leq N - 1$.

2. Divide the sequence to α segment, such that in any segment m with $1 \leq m \leq \alpha$, $(m-1)N \leq \pi(i) < mN$ holds.

3. Let $\pi(i) = \pi(i) - (m-1) \times N$ in any segment m.

4. Reverse the order of each segment and concatenate them. For example, the inverse of $0, 3, 6$ will be $6, 3, 0$.

The following will show that the above interleaving scheme can obtain the result claimed in Theorem 4.1.2.

For any i and j, if they locate in the same segment, then

$$|\pi(i) - \pi(j)| = \alpha|i - j|$$

It is noted that step 2 and 3 will not influence the above equation. If i, j locate in a same segment, $\alpha|i-j| \leq N-1$. Since $|i-j| \geq 1$, it can be obtained that $|\pi(i) - \pi(j)| \geq \alpha$. For i and j locate in different segments, Figure 4.2, (a) shows two neighbour segments before the reverse operation; (b) shows two neighbour segments after the reverse operation. Since the minus calculation in step 3 does not influence the order, it is ignored. If the length of the segment is denoted as M, then $M = \frac{N}{\alpha}$. In each segment, $i_{k+1} = i_k + 1$ and $j_{k+1} = j_k + 1$, with $1 \leq k \leq M$. It can be seen that

$$\pi(i_k) = \alpha i_k - (m-1) \times N \quad 1 \leq k \leq M,$$

$$\pi(j_k) = \alpha j_k - m \times N \quad 1 \leq k \leq M$$

before the reverse operation. After the reverse operation, the followings are obtained

$$\pi(i_k) = \alpha i_{M+1-k} - (m-1) \times N \quad 1 \leq k \leq M,$$

$$\pi(j_k) = \alpha j_{M+1-k} - m \times N \quad 1 \leq k \leq M$$

Since $j_k = i_{k+M}$, $\pi(j_k)$ can be described as

$$\pi(j_k) = \alpha i_{2M+1-k} - m \times N \quad 1 \leq k \leq M$$

Without losing generality, let i_a locate in segment $m-1$ and j_b in segment m. Then

$$\pi(i_a) = \alpha i_{M+1-a} - (m-1) \times N \quad 1 \leq a \leq M,$$

$$\pi(j_b) = \alpha i_{2M+1-b} - m \times N \quad 1 \leq b \leq M.$$

Calculate the absolute difference of above, it is

$$\begin{aligned} |\pi(i_a) - \pi(j_b)| &= |(\alpha i_{M+1-a} - (m-1) \times N) - (\alpha i_{2M+1-b} - m \times N)| \\ &= |N - \alpha(i_{2M+1-b} - i_{M+1-a})| \\ &= |\alpha(M - b + a) - N| \\ &\geq |\alpha(M+1) - N| \\ &> \alpha \end{aligned} \tag{4.3}$$

In above calculation, the fact of $a - b \geq 1$ is utilized. This can be obtained directly from $j_b - i_a < \alpha$. Thus, Theorem 4.1.2 is proved. □

It is conjectured that the similar result as theorem 4.1.2 exists for $N/\alpha \neq$ *integer*. According to Theorem 4.1.2, if $\sqrt{N}$ is integer, then let $\alpha = \sqrt{N}$. The upper bound in Theorem 4.1.1 can be obtained. Although upper bound for any window size less than $\sqrt{N}$ can be obtained by the above method, it is not optimal in general. This is because the interleaved sequence appears some apparent systematic order. Also, when $|i_1 - j_1| \leq |i_2 - j_2|$, it is hoped that $|\pi(i_1) - \pi(j_1)| \geq |\pi(i_2) - \pi(j_2)|$ holds. Therefore, some more effective methods are needed.

4.2 Successive Packing (SP) Based Interleaver Design

In this section, a novel interleaver design, successive packing based interleaver is presented. It aims to maximize the distance of $\pi(i)$ and $\pi(j)$ of any two elements i and j after interleaving at the same time.

4.2.1 SP Algorithm

The concept of successive packing is first proposed in [28] for 2-D interleaving, the optimality for different size error bursts is claimed. In [28], two dimensional basis array is proposed and used as a basic unit for successive packing. To construct a 2-D

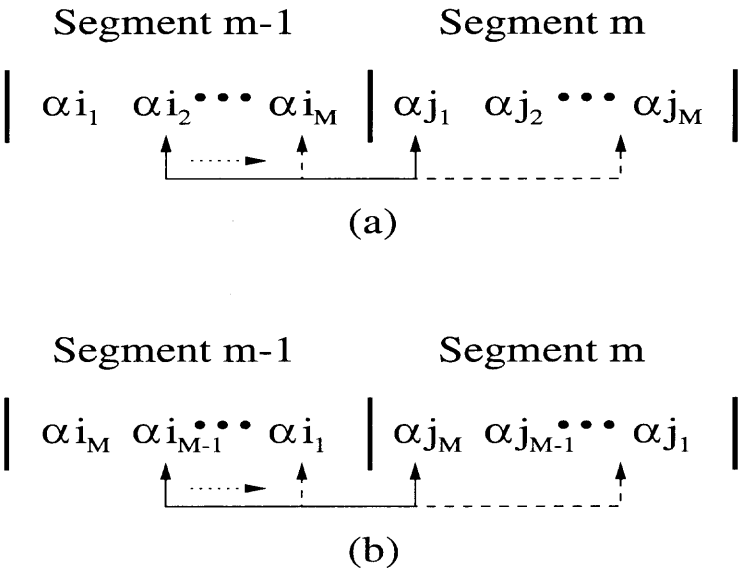


Figure 4.2 Illustration of the element locations before and after reverse.

basis array, the maximization of the distance among the neighbour elements in both X and Y direction must be satisfied. Though there are two branches in the topology of turbo code, the interleaver is still basically one dimensional. In the following, the algorithm is first presented by generating interleaver step by step. Then some theoretical results of it are proved.

Consider a length-2 sequence $S_1 = \{s_0, s_1\}$. Since S_1 only has two elements, even if the order of the elements is changed, the distance between the two element will still be 1. Thus, S_1 is chosen as the basis interleaver. The successive packing can be realized by the following procedure:

Step 1 Generate basis interleaver as

$$S_1 = \{s_0, s_1\}. \tag{4.4}$$

In the following, this is called as basis-2 initialization.

Step 2 Generate length 4 interleaver as

$$S_2 = \{S_1 \times 2 + 0, S_1 \times 2 + 1\} = \{s_0, s_2, s_1, s_3\}. \tag{4.5}$$

Step 3 Continue the above step to generate length 2^{M+1} interleaver as

$$S_M = \{S_{M-1} \times 2 + 0, S_{M-1} \times 2 + 1\}. \tag{4.6}$$

In the above algorithms, the multiplication "+" and plus calculation "-" are applied to the subscript of each element, which denotes the position of each element in the sequence. Thus, the interleaved sequence is generated with the rearrangement of the position of each element. For example, the length 8 interleaver generated sequence is $S_3 = \{s_0, s_4, s_2, s_6, s_1, s_5, s_3, s_7\}$. Where s_4 denotes the second element in the original sequence, and fifth element in the interleaved sequence. For the above SP algorithm, the following results exist.

Lemma 4.2.1. *Let A be a information frame of size $M = 2^n$, and interleaved by using SP and the above mentioned basis-2 interleaver S_1. If A is uniformly divided into 2^{n-k} segments with $0 < k < n$, then for any i, j belong to one segment and $i \neq j$, $|\pi(i) - \pi(j)| \geq 2^{n-k}$. Where $\pi()$ denotes the position after interleaving.*

Proof. Consider the basis-2 interleaver S_1. The distance between s_0 and s_1 is 1. According to the procedure above, $S_2 = \{s_0, s_2, s_1, s_3\}$, where $n = 2, k = 1$. Uniformly dividing S_2 into two segments $\{s_0, s_2\}$, $\{s_1, s_3\}$, it can be obtained that $\pi(0) = 0$, $\pi(1) = 2$ belong to the first segment, and $\pi(2) = 1$, $\pi(3) = 3$ belong to the second segment. In the first segment, $|\pi(i) - \pi(j)| = |0 - 2| = 2$; in the second segment, $|\pi(i) - \pi(j)| = |1 - 3| = 2$. Thus, Lemma 4.2.1 is proved in case $n = 2$. Now assume the result holds in case $n = N$, if it can be proved in case $n = N + 1$, Lemma 4.2.1 will be proved.

Let the interleaved frame be generated as $S_{N+1} = \{S_N \times 2 + 0, S_N \times 2 + 1\}$ in case $n = N + 1$. Based on the assumption, Lemma 4.2.1 is true for S_N, that means if S_N is uniformly divided into 2^{N-k} segments with $0 < k < N$, then for any i, j belong to one segment and $i \neq j$, $|\pi(i) - \pi(j)| \geq 2^{N-k}$. According to the procedure of successive packing, after the calculation $S_N \times 2$, $\pi(i)$ in S_N change to

$2\pi(i)$ in the first half of S_{N+1} and $2\pi(i)+1$ in the second half of S_{N+1}. If i, j belong to one segment in S_N, then $2\pi(i), 2\pi(j)$ belong to the same segment in S_{N+1}, and $2\pi(i)+1, 2\pi(j)+1$ belong to another segment in S_{N+1} with same length. Therefore,

$$|2\pi(i) - 2\pi(j)| = 2|\pi(i) - \pi(j|) \geq 2 \times 2^{N-k} = 2^{N+1-k},$$

and

$$|(2\pi(i)+1) - (2\pi(j)+1)| = 2|\pi(i) - \pi(j)| \geq 2 \times 2^{N-k} = 2^{N+1-k}.$$

Hence, Lemma 4.2.1 is proved. □

It is noted that the length of one segment will be 2^k if A is divided uniformly into 2^{n-k} segments. Thus, if $|i-j| \leq 2^k$, $|\pi(i) - \pi(j)| \geq 2^{n-k}$ holds provided that i and j locate within a same segment.

Theorem 4.2.1. *Let A be a information frame of size $M = 2^n$, and interleaved by using SP and the above mentioned basis interleaver S_1. Then (a) if $|i-j| = 1$, $|\pi(i) - \pi(j)| \geq 2^{n-2}$ as $n \geq 2$; (b) if $|i-j| = 2$, $|\pi(i) - \pi(j)| \geq 2^{n-3}$ as $n \geq 3$; (c) if $|i-j| < 2^k$ as $k \leq n-2$, $|\pi(i) - \pi(j)| \geq 3$.*

Proof. **(a)**: According to Lemma 4.2.1, as $|i-j| \leq 2^k$ and they belong to a same segment, $|\pi(i) - \pi(j)| \geq 2^{n-k}$. Let $k = 1$, then $|\pi(i) - \pi(j)| \geq 2^{n-1} > 2^{n-2}$. Thus, Theorem 4.2.1 is proved under this condition and it only need to prove the case that i, j locate in neighbour segments respectively. Figure 4.3 shows the numbers corresponding to the positions of the symbols after interleaving. When $n = 2$, the only case that $\pi(i), \pi(j)$ locate in neighbour segments in S_2 are $\pi(1) = 2$ and $\pi(2) = 1$. So $|\pi(i) - \pi(j)| \geq 2^{n-2} = 1$. When $n = 3$, In the first half, there is $S_2 \times 2$; in the second half, there is $S_2 \times 2 + 1$. Thus, $\pi(1) = 4$, $\pi(2) = 2$, $\pi(5) = 5$, $\pi(6) = 3$. Correspondingly, $|4-2| = 2$ and $|5-3| = 2$. Now consider the new generated neighbour-2 symbols in the middle of the interleaved sequence. According to the SP procedure, they are decided by the first and the last symbol of S_2 as shown in Figure

4.3 by the solide arrow line. The subscript of the left symbol equals to double of the last symbol of S_2, and the position of the right symbol equals to double of the first symbol of S_2 plus 1. In S_3, there is $|2 \times 3 - (2 \times 0 + 1)| > 2$. Hence, Theorem 4.2.1 is proved for the case $n = 3$. Now assume Theorem 4.2.1 holds in case $n = N$, if it can be proved to hold in case $n = N + 1$, it will be proved. Based on the same reasoning as the proof of Lemma 4.2.1, it can be shown that Theorem 4.2.1 holds in the left half and right half of S_{N+1}. Now, the new generated neighbour-2 symbols in the middle of the interleaved sequence S_{N+1}, are determined by the first and last symbols of S_N through the SP algorithm. In general, there is $|2 \times (2^N - 1) - (2 \times 0 + 1)| > 2^{N-2}$. Thus, the result holds for all of the neighbour-2 symbols, and Theorem 4.2.1 (a) is proved.

(b): This part can be proved based on the same reasoning as part (a). In part (a), it is shown that the new generated neighbour-2 symbols in the middle of the interleaved sequence play key roles in the proof for $|i - j| = 1$. For $|i - j| = 2$, the new generated neighbour-3 symbols in the middle of the interleaved sequence play key roles in the proof. Besides the neighbour-2 symbols shown in (a), the other two symbols are decided by the symbols pairs which are shown in Figure 4.3 connected by the dashed arrow line. Thus, the two neighbour-3 symbols groups in the middle of the interleaved sequence S_{N+1} will be $\{2^N - 2, 2^{N+1} - 2, 1\}$ and $\{2^{N+1} - 2, 1, 2^N + 1\}$. It is apparently that distance between any two elements in the same neighbour-3 symbols group is greater than 2^{N-2}. Theorem 4.2.1 (b) is proved.

(c): According to Lemma 4.2.1, when $|i - j| \leq 2^k$ and they belong to a same segment, $|\pi(i) - \pi(j)| \geq 2^{n-k}$. If $2 \leq k \leq n - 2$, $|\pi(i) - \pi(j)| \geq 2^{n-k} \geq 2^{n-(n-2)} = 4 > 3$ holds. Hence, one only needs to prove the case that i, j locate in neighbour-4 segments respectively. Since the case of neighbour-2 and neighbour-3 symbols in the middle of the interleaved sequence have been proved, only the other neighbour-4 symbols, which are decided by the symbols pairs shown in Figure 4.3 connected by

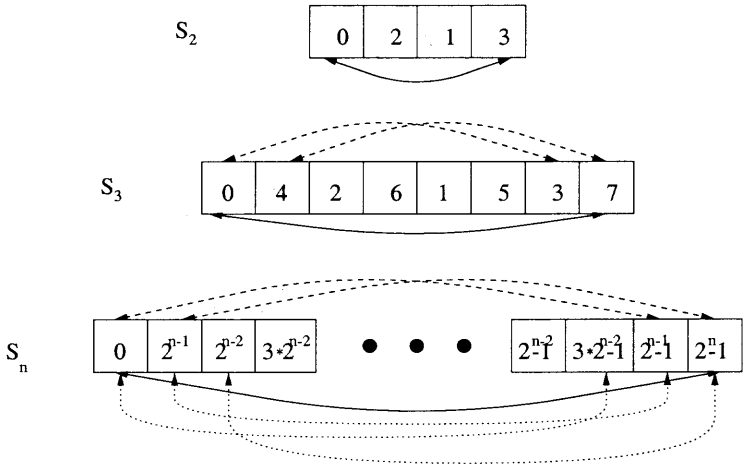


Figure 4.3 Key elements that influence the distance after packing.

the dotted arrow line, are considered. Thus, the three neighbour-4 symbol groups in the middle of the interleaved sequence S_{N+1} will be $\{3\times 2^{N-1}-2, 2^N-2, 2^{N+1}-2, 1\}$, $\{2^N-2, 2^{N+1}-2, 1, 2^N+1\}$ and $\{2^{N+1}-2, 1, 2^N+1, 2^{N-1}+1\}$. Among the three groups, only $|(2^N-2)-(2^N+1)| = 3$. For any two other elements in the same neighbour-4 symbol group, the distance is greater than 2^{N-2}. Since $N \geq 4$, Theorem 4.2.1 (c) is proved. □

It is noted the distance of 3 is only a lower bound when $|i-j| < 2^k$ and $2 \leq k \leq n-2$. In fact, it only happens in the middle of the interleaved sequence $N/2$. If the window is moved to either left or right, the shortest distance will increase step by step to $6, 12, 24, \cdots$. That means the large distance can be obtained in most of the case.

In last section, the block interleaver and the extented linear interleaver is introduced. In fact, they can also be obtained by the principle underlying the proposed SP technique. Consider the following procedure:

Step 1 Initialize the SP procedure by

$$S^* = \{s_0, s_1, \cdots, s_{m-1}\}. \tag{4.7}$$

Step 2 Generate length $N = m \times n$ interleaver as

$$S(N) = \{S^* \times n + 0, S^* \times n + 1, \cdots, S^* \times n + m - 1\} \tag{4.8}$$

The above SP procedure generates the same sequence as Eq. 4.1. For example, let $N = 3{\times}4$. Based on Eq. 4.1, the interleaved sequence is $\{0, 4, 8, 1, 5, 9, 2, 6, 10, 3, 7, 11\}$. Using the proposed approach, the basis interleaver is first generated as $\{0, 1, 2\}$, then the SP procedure is used to obtain $\{0, 4, 8, 1, 5, 9, 2, 6, 10, 3, 7, 11\}$. To generate the interleaver that proposed in last section, it only need to let

$$S^* = \{s_{m-1}, s_{m-2}, \cdots, s_0\},$$

and then use the above SP procedure.

4.2.2 Optimality Discussion

The main objective of the interleaver is to permute the input sequence with low Hamming weight such that both the upper and lower concolutional encoder do not generate low Hamming weight parity output at the same time. To optimize the interleaver, the effort is mainly focused on the weight-2 input sequence. Let L be the cycle length (period) of the feedback primitive polynomial, it has been shown that any weight-2 sequence of the form $(1 + D^L)D^s$ applied to an encoder with cycle length L, produces a low weight parity sequence [38]. To avoid the occurrence of it, if the two 1 in the input sequence are iL apart, they should be put into the positions that are not iL apart. Using this principle, if one encoder generates the low Hamming weight parity sequence, the other encoder will generate higher Hamming weight parity sequence. It has been shown that block interleaver cannot overcome some combinations of two weight-2 error (weight-4 error). Although the good S-random interleaver has been proved to avoid the weight-2 sequence generated low weight parity output effectively, however, as a pseudo-random generated interleaver, it is computational expensive and cannot guarantee the performance after changing

the block length. Among the approaches to overcome this problem, the prunable interleaver proposed by Eroz *et al.* [41] gets the satisfied result and has been adopted in the standard of cdma2000 [43]. Considering our proposed SP method, it can be seen that most of the occurrence of weight-2 error can be avoided. Moreover, SP interleaver is prunable owing to its generation algorithm. In the cdma2000 prunable interleaver, the interleaving and deinterleaving depend on an optimized table. In the SP algorithm, the interleaving and deinterleaving are identical, thus, SP has the implementation advantage.

Besides the weight-2 criterion, IDS criterion has also been proposed recently [37]. This criterion is based on the measure of correlation between two data sequences. If these two data sequences are less correlated, then the performance of the iterative decoding algorithm is improved. Since the calculation of correlation is not easy, a graphical representation is used to show the scattering of the points. An interleaver with mapping function $i \rightarrow \pi(i)$ can be represented by N points in the (i, j)-plane. Intuitively, if large distances exist among arbitrary points, a small correlation can be obtained. Figure 4.4 shows the SP interleaver with basis interleaver $\{s_0, s_1\}$ and $N = 130$, which is generated by pruning SP interleaver with length 256. It is seen that the points $\pi(i)$ spread over all of the (i, j) plane. If it is compared with the random interleaver, their distributions are similar. Thus, it is reasonable to conjecture that SP interleaver has the advantage of pseudo-random interleaver. Inspite of several points, there is a large distance between the points. Therefore, the IDS criterion is satisfied. Figure 4.5 shows the prunable interleaver adpted by cdma2000. It is obtained by pruning mother interleaver with length 512. It is seen its distribution is not random.

Block interleavers appear some apparent order and are not optimal in the sense that $|i_1 - j_1| \leq |i_2 - j_2|$ does not lead to $|\pi(i_1) - \pi(j_1)| \geq |\pi(i_2) - \pi(j_2)|$. But the minimum distance between any two elements is maximized. On the other hand,

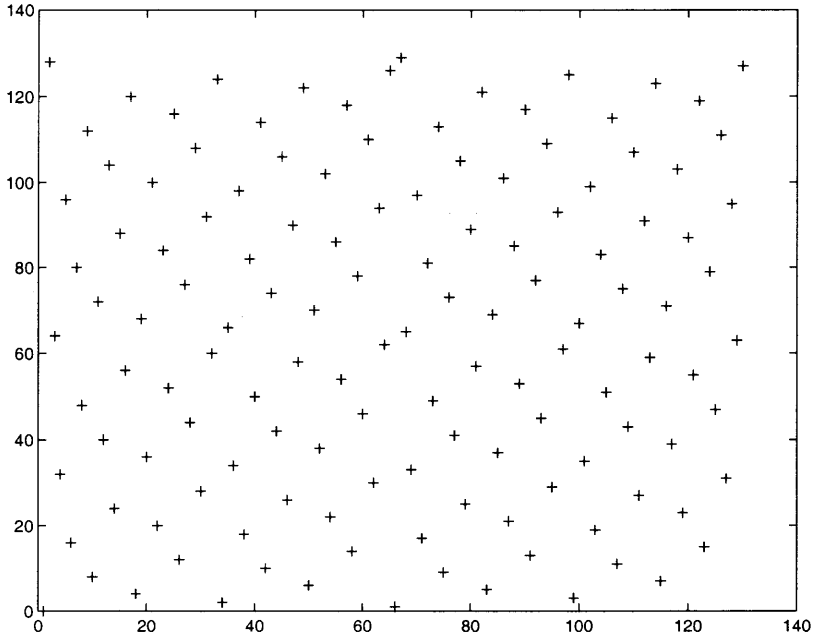


Figure 4.4 Graphical representation of the elements distribution of the SP interleaver with $N = 130$.

the proposed basis-2 SP interleaver has such optimal characteristic except for few elements. Thus, if the advantages of both can be combined, the resultant interleaver will be better. As what have shown before, block interleavers can also be implemented by the SP principle. To combine their advantages, the following procedure can be used.

Step 1 Generate S^* with length N_1 by basis-2 initialization.

Step 2 Generate length $N = N_1 \times n$ interleaver as

$$S(N) = \{S^* \times n + 0, S^* \times n + 1, \cdots, S^* \times n + N_1 - 1\} \tag{4.9}$$

For example, let $N_1 = 4$ and $N = 16$. Then $S^* = \{0, 2, 1, 3\}$, and $S(N) = \{0, 8, 4, 12, 1, 9, 5, 13, 2, 10, 6, 14, 3, 11, 7, 15\}$.

Although weight-2 criterion has been adopted by most of the researchers, Wang *et al.*[44] show that the short length interleavers obtained in this manner provide little performance advantage for the resultant turbo codes compared to turbo codes using

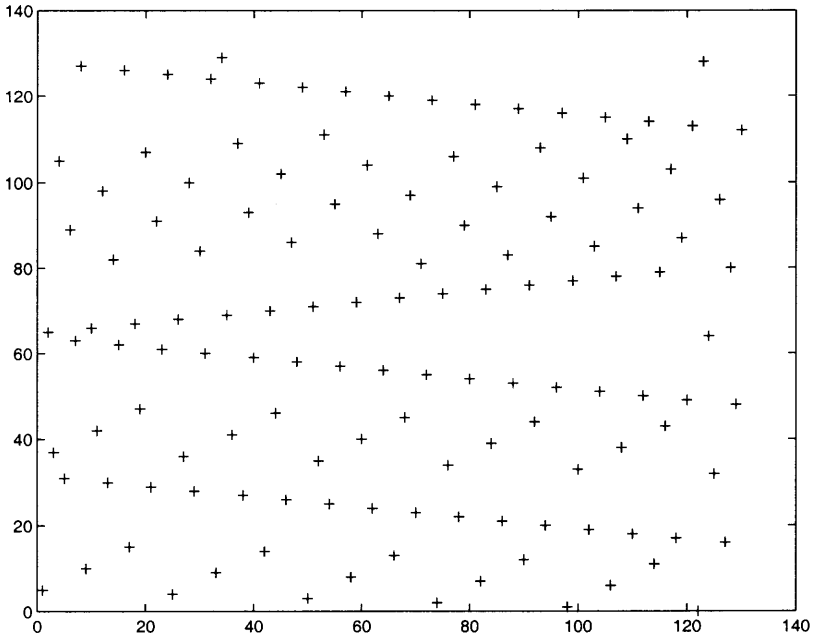


Figure 4.5 Graphical representation of the elements distribution of the CDMA prunable interleaver with $N = 130$.

block interleavers. They further attribute this to the neglect of weight-3 and weight-4 sequences. Consider an encoder with cycle length 7 and the feedback polynomial $1+D+D^3$. It can be shown that any input sequence with $(1+D^{1+7j}+D^{3+7k})D^i$ lead to low weight parity output, and for each of these weight-3 input, there is a weight-4 input lead to low weight parity output too, where i, j, k are arbitrary integers. To solve this problem, they propose to use an length-7 sequence $\{4, 1, 5, 6, 0, 2, 3\}$ to avoid the occurrence of the above mensitioned weight-3 and weight-4 sequences. This sequence is then used to construct the longer interleaver. They claim that the elements with similar value should be separated far apart. To optimize the proposed SP interleaver, the length-7 sequence $\{4, 1, 5, 6, 0, 2, 3\}$ can also be used in the procedure of successive packing. For simplicity, assume $N = 7 \times 2^n$, for other length, the interleaver can be obtained by pruning. The procedure can be described as follows.

- Using the SP algorithm (basis-2) to generate the interleaver S^* with length 2^n.

- Generate the optimize interleaver as

$$S(N) = \{S^* \times 7+4, S^* \times 7+1, S^* \times 7+5, S^* \times 7+6, S^* \times 7, S^* \times 7+2, S^* \times 7+3\}.$$

Using the above procedure, the advantage of basis-2 SP is kept and the weight-3 and weight-4 input resultant low weight parity output is avoided.

4.3 Experimental Results

In this section, the proposed SP algorithm and other well known deterministic interleaver design techniques are applied to the turbo encoder and decoder as shown Figure 4.1. The simulation results of Bit error rate (BER) and Frame error rate (FER) respecting to interleavers with different length are demonstrated. In all of the results, the number of simulation iterations is eight.

Figure 4.6 shows the comparison result of the BER performance of three interleaver design: cdma2000 interleaver, deterministic interleaver in [36] and the SP interleaver with block length of 378. Among the three interleavers, cdma2000 interleaver is obtained by pruning the "mother" interleaver with block length of 512; the deterministic interleaver in [36] is obtained by $\alpha = 19, \beta = 9$; the SP interleaver is obtained by pruning the "mother" interleaver with block length of 512, which is generated by basis-2 initialization. Figure 4.7 shows the comparison result of the FER performance of the above three interleaver design with the same design parameters. Since the BER and FER are very small with the increase of signal to noise ratio (Eb/No), In both figures, some points are missing.

The performance of the three interleavers with block length of 570 are also compared. For the design of the three interleavers, cdma2000 interleaver is obtained by pruning the "mother" interleaver with block length of 1024; the deterministic interleaver in [36] is obtained by $\alpha = 23, \beta = 11$; the SP interleaver is obtained by pruning the "mother" interleaver with block length of 1024, which is generated by

basis-2 initialization. Figure 4.8 and Figure 4.9 shows the BER and FER performance of the above three interleaver design respectively. Similiar as above, some points are missing in both figures due to the very small value.

To optimize the proposed SP interleaver, the above simulations are repeated with the following interleaver design: To generate the interleaver with block length of 378, the basis-2 initialization is first used to generate a sequence S^* with length 32, then the length-512 interleaver is obtained by

$$S(512) = \{S^* \times 16 + 0, S^* \times 16 + 1, \cdots, S^* \times 16 + 15\}.$$

Length-378 interleavers is finally obtained by pruning the length-512 interleaver $S(512)$. To generate the interleaver with block length of 570, the basis-2 initialization is first used to generate a sequence S^* with length 64, then the length-1024 interleaver is obtained by

$$S(1024) = \{S^* \times 16 + 0, S^* \times 16 + 1, \cdots, S^* \times 16 + 15\}.$$

Length-570 interleavers is finally obtained by pruning the length-1024 interleaver $S(1024)$.

The performance of the optimized SP interleaver is compared with the above three interleaver, the results are also shown in Figures 4.6 to 4.9. It is seen the performance of the proposed SP interleaver is better than cdma2000 prunable interleaver and the deterministic interleaver in [36] for the structure Figure 4.1 without any optimization. If the advantages of both block interleaver and the basis-2 SP interleaver are combined, the performance is further improved.

4.4 Summary

In this chapter, a novel deterministic turbo codes interleaver design method, successive packing (SP), is presented to improve the turbo codes performance. The SP method is based on the packing of the basis initialization. It can support

different block sizes from several hundreds of bits to tens of thousands of bits according to variable input data requirements with simple pruning. Thus, the re-implementation cost is avoided. Moreover, various criterions can be easily incorporated in the SP algorithm to optimize its performance. It is shown that the SP resultant interleaved sequence has the random-like distribution. According to its characteristics, the value of $|i-j|$ is inversely proportional to $|\pi(i)-\pi(j)|$ in most of cases. Hence, the corresponding correlation is minimized. Part of this research work has been published in [45].

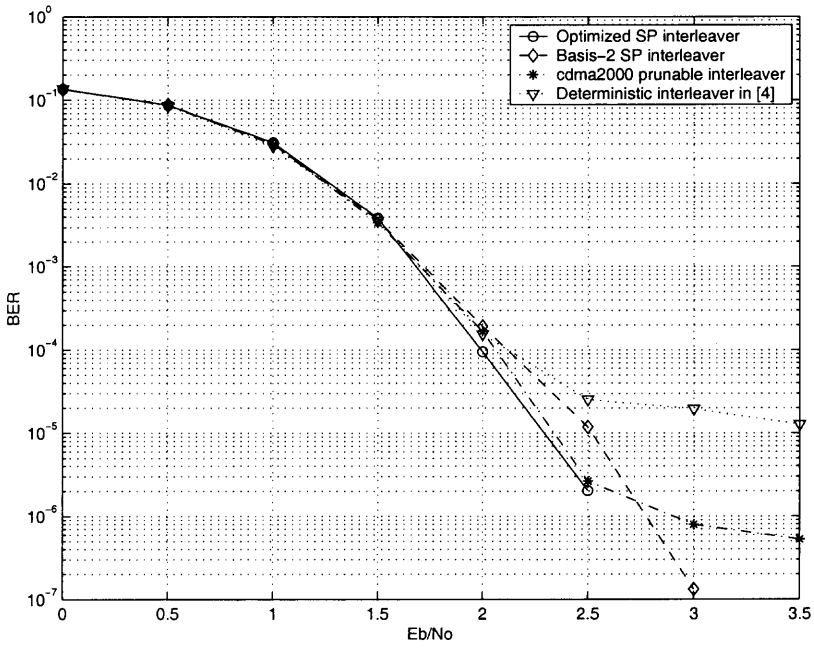


Figure 4.6 BER performance of cdma2000 interleaver and SP interleaver with $N = 378$.

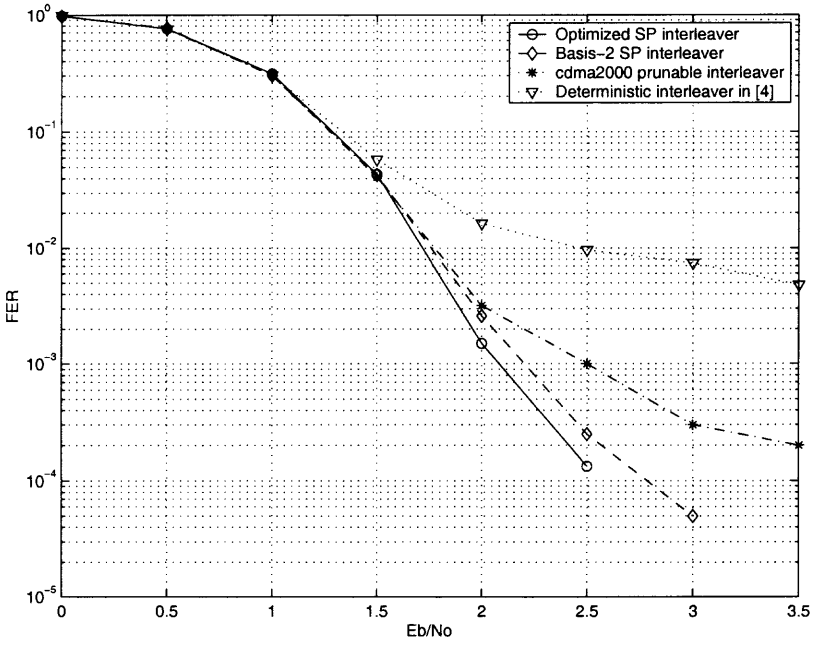


Figure 4.7 FER performance of cdma2000 interleaver and SP interleaver with $N = 378$.

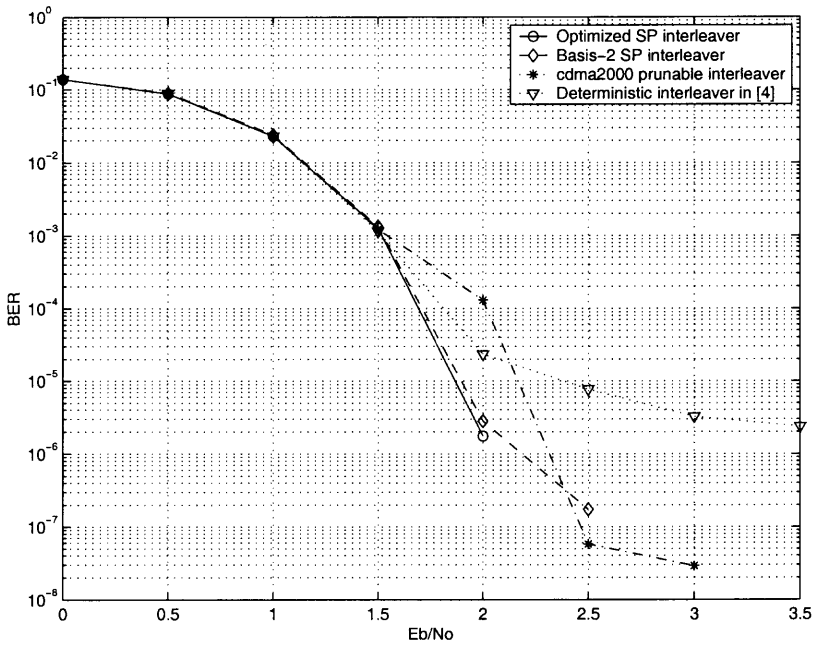


Figure 4.8 BER performance of cdma2000 interleaver and SP interleaver with $N = 570$.

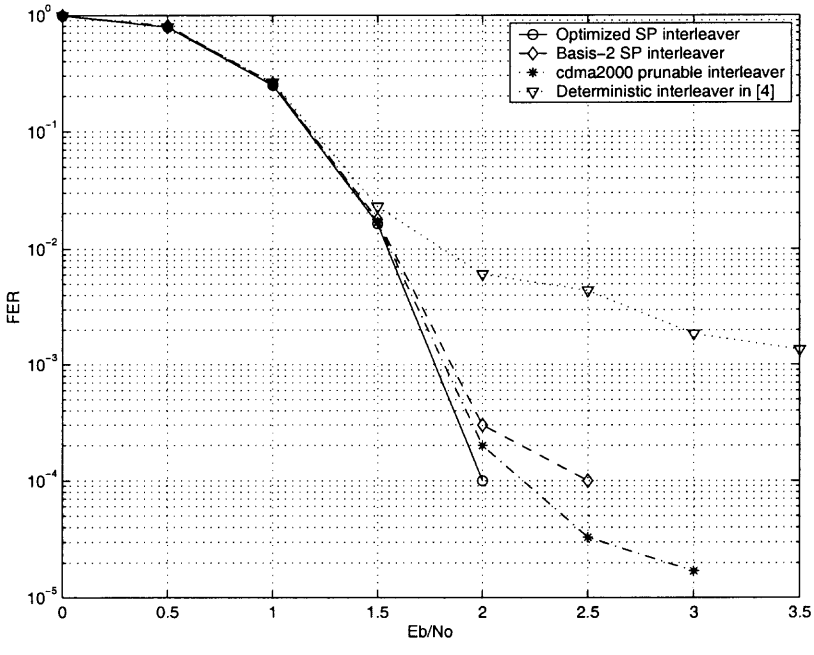


Figure 4.9 FER performance of cdma2000 interleaver and SP interleaver with $N = 570$.

CHAPTER 5

CONSTANT QUALITY CONSTRAINED RATE ALLOCATION FOR FGS CODED VIDEO

With advances in networking technology, the Internet has become a primary medium for information transmission. The technique that can adapt the outgoing traffic to meet constraints imposed by users and networks is desirable. In CBR coding scheme, to account for minor fluctuations in the bits produced at each frame, the output bits of an encoder are sent to a buffer. Subsequently, it is the buffer that releases bits at a constant bit-rate to the channel. There are many advantages with CBR coded video, however, it does have certain drawbacks. One drawback is that picture quality fluctuates. In the case of video recorded on a DVD, picture quality should be constant and there is no need to impose CBR restrictions. Another drawback of CBR is that it does not provide an efficient means of transmitting video over time-varying heterogeneous networks. Such a network is characterized by varying bandwidth and/or sessions that are established based on available bit-rate (ABR) among many users. Thus, variable bit-rate (VBR) is often used to slove the above problems.

Optimal bit allocation in the rate-distortion (R-D) sense was first addressed by Huang and Schultheiss [7] for transform coded data. This chapter is focused on the optimal allocation of bits among different quantizers assuming a Gaussian random source. In the context of video coding, rate control methods that adjust the quantization step size have been proposed [8, 9], where optimal video coding is obtained when all macroblocks have the same R-D characteristic. The above works consider optimal allocation among blocks for single-layer coding schemes, i.e., scalable transmission has not been considered. Recently, a scalable quadratic R-D model was proposed and became the basis for MPEG-4 rate control [46]. The rate control scheme corresponding to multi-objects was discussed in [47].

To obtain more reliable transmission under a given bandwidth constraint, joint source-channel coding has been studied. In [48], a joint power and rate allocation framework was proposed for subband image and video transmission. The authors first perform a subband decomposition of the source, then the source encoder and channel modulator are devised to provide unequal error protection for different subbands. The bit allocation is implemented by minimizing the distortion in the mean square sense. In [49], Gallant and Kossentini introduced a rate-distortion optimized mode-selection algorithm within a prioritized layered framework. Their algorithm is based on a joint source-channel coding approach and trades off source-coding efficiency for increased error-resilience to optimize the video-coding mode selection within and across layers.

For scalable coding schemes, Li and Lei [50] consider the R-D optimization of an embedded image coder that is based on wavelet decomposition and zero-tree coding. In their work, the performance of the encoder is improved by coding the coefficients with decreasing R-D slope. However, due to the computational cost, it is difficult to extend this idea to video coding.

The previous optimal rate allocation techniques methods heavily depends on the model used. However, the existing model cannot approximate real conditions very well. To solve this problem, Lin and Ortega [51] proposed to model R-D characteristics based on measurements of the actual rate-quantizer and distortion-quantizer data. In yet other works, dynamic bandwidth allocation has been studied to smooth the burstiness of compressed video stream [52], [53], however, smoothing the variation in quality on a frame-by-frame basis has not been considered in this work.

Users require playback with minimal variation in quality, but dynamic network conditions often make this difficult to achieve with single-layer coding schemes. Advances in scalable video coding have made it easier to scale pre-encoded bitstreams to meet such network conditions [54], [55]. In a heterogeneous network environment,

such as the Internet, variable channel conditions can damage the integrity of the reconstructed video. However, if the server can transmit only the important data at a reduced rate, congestion is prevented and the overall video quality is improved considerably. On the other hand, particular clients may expect high quality video provided that they have enough bandwidth and a capable decoder that can handle the higher data rates. If scalable coding is used, one copy of high quality video bit stream may be stored in a multimedia server and only part of the bit stream is delivered depending on the client demand and channel condition.

Fine-Granular Scalablilty (FGS) coding [11] is different from traditional scalable coding schemes. With traditional scalable coding techniques, the content would be coded into a base layer and possibly several enhancement layers, where the granularity is only as fine as the number of enhancement layers that are formed. As a result, the resulting rate-distortion curve resembles a step-like function. In contrast, FGS provides an enhancement layer that is continually scalable. This is accomplished through a novel bit-plane coding method of DCT coefficients in the enhancement layer, which allows the enhancement layer bitstream to be truncated at any point. In this way, the quality of the reconstructed frames is proportional to the number of enhancement bits received. The principle of FGS coding is illustrated in Figure 5.1. Since the prediction is always based on the base-layer, the bit stream of each frame can be truncated at any point without affecting subsequent frames. However, the coding efficiency is sacrificed to some extent compared with single-layer coding schemes. In order to obtain a good balance between granularity and coding efficiency, Progressive Fine Granular Scalability (PFGS) coding has been proposed [56]. In contrast to FGS coding, this framework supports prediction from an improved reference frame, which is essentially the base layer with a portion of the enhancement layer added.

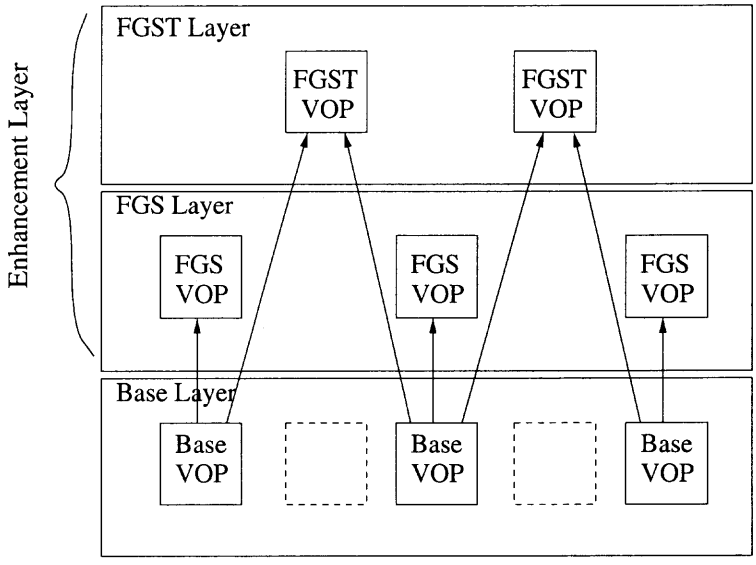


Figure 5.1 Illustration of basic structure used for FGS and FGST coding.

An important point to emphasize is that the standard itself does not specify how any form of rate allocation should be done. In the FGS/FGST framework, there are several unique types of rate allocation that one may consider, e.g., the truncation of enhancement layer bits, the optimal allocation of rate between the base and the enhancement layers, as well as the temporal-SNR trade-off proposed in [12]. This chapter deals only with the truncation of enhancement layer bits to achieve constant quality and is independent of the original bit allocation between base and enhancement layers. The rate and quality of the base layer is assumed to be a lower bound. Enhancement layer bits cover the range of bit-rates from this lower bound to near lossless quality. Also, once the enhancement layer bitstream has been generated, it is stored at the server and re-used many times. According to e.g., network characteristics, an appropriate number of bits will be allocated to a frame and transmitted. Wang, et al. [57] have studied the problem of rate allocation in the enhancement layer for the PFGS coding scheme. In their paper, an exponential model is used to realize optimal rate allocation. Average PSNR improvements in the range of 0.3 to 0.5db have been reported.

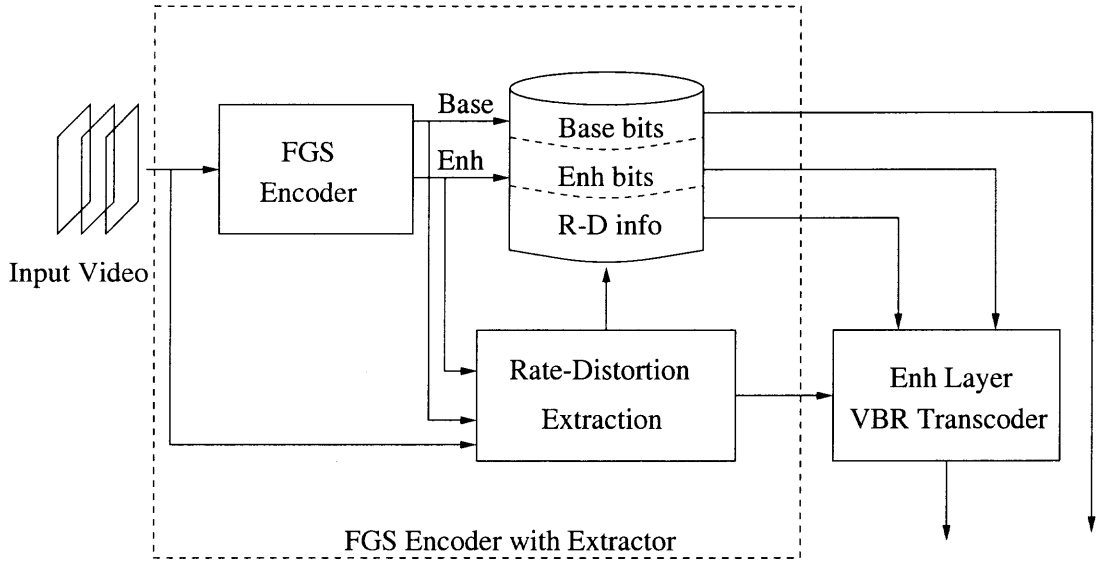


Figure 5.2 Overview of proposed rate allocation framework for achieving constant quality.

In this chapter, an optimal rate allocation strategy is considered for FGS and FGST coded bitstreams that achieves constant quality reconstruction of frames under a dynamic rate budget constraint. In doing so, it is also aimed to minimize the overall distortion at the same time. The basic concept of the proposed scheme is illustrated in Figure 5.2. The scheme includes an FGS encoder, a rate-distortion extractor and enhancement-layer VBR transcoder. The FGS encoder produces a base layer and an FGS enhancement layer bitstream. The rate-distortion extractor is based on a novel R-D labeling scheme to characterize the R-D relationship of the source coding process. Specifically, a set of actual R-D points are sampled in the encoding process and linear interpolation is used to estimate the real R-D curve of the enhancement layer signal. The enhancement layer VBR transcoder is based on a rate allocation method that is based on a sliding window approach. This approach makes use of the R-D information that is generated by the R-D extractor. The transcoder determines the bits that should be allocated per frame so that constant quality among frames is maintained.

This rest of this chapter is organized as follows. In Section 5.1, the optimal rate allocation problem is formulated, and R-D characteristics of the video sequence are discussed. A rate-distortion labeling scheme is then proposed to extract the R-D relationship of source coding in Section 5.2. Specifically, a set of actual R-D points are sampled in the encoding process and linear interpolation is used to estimate the actual R-D curve. A sliding window based rate allocation approach is then proposed in Section 5.3. Experimental results demonstrate that the proposed method can be used to effectively minimize the variation in quality of the reconstructed frames in Section 5.4. The summary of this work are presented in Section 5.5.

5.1 Background of Optimal Rate Allocation

It is known that differential sensitivity has a significant impact on the human's visual perception [58]. Therefore, the focus is on minimizing the variation in quality. In this section, the problem of optimal rate allocation is first studied from the viewpoint of minimizing average distortion. Then, the relationship between minimizing the average distortion and minimizing the variation in quality is investigated. Some important points are clarified for completeness.

In the FGS coding scheme, the enhancement layer of each frame is coded independently of other enhancement layer frames. Thus, the total distortion, D_{FGS}, and total rate, R_{FGS}, of the enhancement layer can be calculated as the sum of distortion and rate over all frames.

$$D_{FGS} = \sum_{i=0}^{N-1} D_i(R_i), \quad and \quad R_{FGS} = \sum_{i=0}^{N-1} R_i \tag{5.1}$$

where N is the total number of frames, and D_i and R_i are the distortion and rate at each frame i, respectively.

Let D_i and R_i be the distortion and rate at each frame i, respectively. To minimize the average distortion, it is equivalently to minimize the cost function,

$J(\lambda)$,

$$J(\lambda) = \sum_{i=0}^{N-1} D_i(R_i) + \lambda \sum_{i=0}^{N-1} R_i, \quad \text{subject to} \quad \frac{F_s}{N} \sum_{i=0}^{N-1} R_i \leq R_{budget} \tag{5.2}$$

where N is the total number of frames, λ is a Lagrangian multiplier, F_s is the source frame rate, and R_{budget} denotes the available bandwidth. Given the Gaussian model, $D(R) = a\sigma^2 2^{-2R}$, where R denotes the average bits per pixel, σ^2 is the signal variance, and a is a constant that is dependent on the pdf of the signal and quantizer characteristics, the solution to the above problem is given by,

$$R_i = -\frac{1}{2} log_2 \frac{\lambda}{2a\sigma_i^2} \tag{5.3}$$

If the given constraint is satisfied with equality, then the optimal rate allocation is,

$$R_i = \frac{R_{budget}}{F_s} + \frac{1}{2} log_2 \frac{\sigma_i^2}{(\prod_{j=0}^{N-1} \sigma_j^2)^{\frac{1}{N}}} \tag{5.4}$$

In the above result, one constraint is ignored. That is $R_i \geq 0$. For a CIF resolution video sequence, one frame contains 352 × 288 pixels. Since the unit of rate is bits per pixel (bpp) in the selected model, R_{budget} will be very small at low bit-rates. According to Equation (5.4), it is possible to get $R_i < 0$. For example, let the rate budget be 10 kbps and F_s = 30Hz. Then, R_{budget}/F_s = 0.00329 bpp. Without loss of generality, let $N = 10$, $\sigma_1^2 = 20$ and $\sigma_j^2 \geq 25$ for all $j > 1$. For this set of input parameters, R_1=-0.1449 bpp, which equates to -14.355 kb/frame. For $j > 1$, R_j=1.9667 kb/frame.

$$\begin{aligned} R_1 &= \frac{R_{FGS}}{10} + \frac{1}{2} log_2 \frac{\sigma_i^2}{(\prod_{j=0}^{N-1} \sigma_j^2)^{\frac{1}{N}}} \\ &< 0.00986 + \frac{1}{2} log_2 \frac{20}{(25^9 \times 20)^{\frac{1}{10}}} \\ &= -0.1448 \end{aligned} \tag{5.5}$$

Since a negative amount of bits have been allocated, this result cannot be used for bit allocation. By adding the constraint that $R_i \geq 0$, the solution to the problem

can be re-written as,

$$R_i = (-\frac{1}{2} log_2 \frac{\lambda}{2a\sigma_i^2})^+ \tag{5.6}$$

where $(x)^+$ denotes the positive part of x. In this way, the rate constraint given in Equation (5.2) would be satisfied with equality. This result is referred to as *water filling* in information theory. It is noted that the solution given by Equation (5.6) will give different results from solution given by Equation (5.4). Considering the same input parameters as before, the solution given by Equation (5.6) is R_1=0 and R_j=0.37 kb/frame for $j > 1$.

In the above, it can be assumed that the R-D function, $D(R_i)$, of any frame is monotonically decreasing. Then, the solution to this problem is also the solution to the same problem with the constraint $R_{FGS} = R_{budget}$. This observation is clarified in the following example.

Consider the solution to Equation (5.2) as a set of rates $\{R_0, R_1, \cdots, R_l, \cdots, R_{N-1}\}$, where R_l corresponds to frame l which $D_l(R)$ is monotonically decease, and $\sum_{i=0}^{N-1} R_i < R_{budget}$. Then $R_{budget} - \sum_{i=0}^{N-1} R_i > 0$. Let $R_{new} = R_l + R_{budget} - \sum_{i=0}^{N-1} R_i$ and replace R_l with. Since $R_{new} > R_l$, according to the monotonic condition, it is obvious that $D(R_{new}) < D(R_l)$. Consequently,

$$D(R_1) + \cdots + D(R_l) + \cdots + D(R_{N-1}) < D(R_1) + \cdots + D(R_{new}) + \cdots + D(R_{N-1})$$

which contradicts that the set of rates $\{R_0, R_1, \cdots, ZR_l, \cdots, R_{N-1}\}$, is the solution of minimizing the average distortion. Thus, the above point is clarified.

To minimize the variation in quality, it is equivalent to minimize the following function,

$$\xi = \sum_{i=0}^{N-2} |D_i(R_i) - D_{i+1}(R_{i+1})|, \quad \text{subject to} \quad \frac{F_s}{N} \sum_{i=0}^{N-1} R_i \leq R_{budget} \tag{5.7}$$

Since the above should be solved while minimizing the average distortion, a similar result as above exists. Namely, if the R-D functions, $D(R_0), \cdots, D(R_{N-1})$,

of every frame are monotonically decreasing, the solution to the problem of Equation (5.7) is also the solution to the same problem with the constraint $R_{FGS} = R_{budget}$. Here, a similar example as given above can be used to clarify this observation.

Given the solution of Function 5.2 as a set of rates $\{R_0, R_1, \cdots, R_{N-1}\}$, and $\sum_{i=0}^{N-1} R_i < R_{budget}$. Then $R_{budget} - \sum_{i=0}^{N-1} R_i > 0$. Let $R_{new} = R_{budget} - \sum_{i=0}^{N-1} R_i$ and divided it into N parts as $\{R_0^{new}, R_1^{new}, \cdots, R_{N-1}^{new}\}$, where $R_{new} = \sum_{i=0}^{N-1} R_i^{new}$. Due to the monotonic decrease property, there must exist one distribution method such that $D(R_i) - D(R_i + R_i^{new}) = C$ for any $0 \leq i \leq N-1$, where C is a positive constant. It is obvious that this will not increase the variation. At the same time, the overall distortion $\sum_{i=0}^{N-1} D(R_i + R_i^{new}) = \sum_{i=0}^{N-1} D(R_i) - NC < \sum_{i=0}^{N-1} D(R_i)$. This contradicts that the set of rates $\{R_0, R_1, \cdots, R_l, \cdots, R_{N-1}\}$, is the solution for minimizing of the variation in quality without increasing the overall distortion. Thus, the above observation is clarified.

Ideally, there is no variation in quality among neighboring frames, i.e., $\xi = 0$. It can be shown that the solution to Equation (5.4) for minimizing the overall distortion given the model $D(R) = a\sigma^2 2^{-2R}$ leads to equal quality among frames, i.e., $D_0(R_0) = D_1(R_1) = \cdots = D_{N-1}(R_{N-1})$ [57]. More generally, it can be shown that this result holds for any exponential model, b^{-kx}, in which each frame has the same constant k in the exponent.

The converse can also be considered if it is true. That is, whether constant quality also leads to minimum overall distortion. It can be shown that for a monotonically decreasing R-D function, $D(R_i)$, of each frame, the solution that satisfies the problem given by Equation (5.7) is unique, given the additional constraint that $D_i(R_i) = D_j(R_j)$, $i \neq j$, $0 \leq j \leq N-1$.

Considering all of the discussion above, the following result can be obtained: *If the R-D relationship $D(R_i)$ of each frame is exponentially decreasing with the same constant in the exponent, the solution to the problem given by Equation (5.7)*

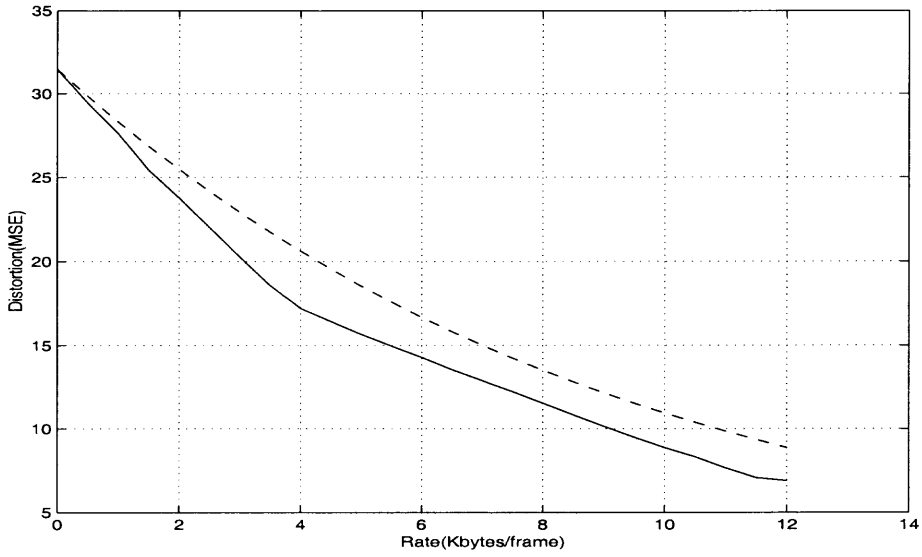


Figure 5.3 Actual R-D curve (solid) vs exponential model estimated R-D curve (dashed): First B-frame in FGST layer of Foreman at CIF resolution.

for minimizing the variation in quality is also the solution to the problem given by Equation (5.2) for minimizing the overall distortion.

According to the above result, it is possible for one rate allocation scheme to provide constant quality under an overall rate constraint, while also minimizing the overall distortion in an optimal way. This analysis provides a new perspective with regard to the optimal rate allocation problem for video coding schemes.

5.2 R-D Extraction of FGS Coded Video

In this section, a scheme is proposed to extract the R-D characteristics of FGS coded video. An R-D labeling scheme is used to provide the necessary R-D information about the source coding. This method is mainly used to overcome problems caused by the inaccuracy of closed form models at low bit-rates.

A comparison of the actual R-D curve for FGS with the exponential R-D model is shown in Figure 5.3 and Figure 5.4. From these plots it can be seen that the exponential model is not sufficient to characterize the FGS coded video. The reasons

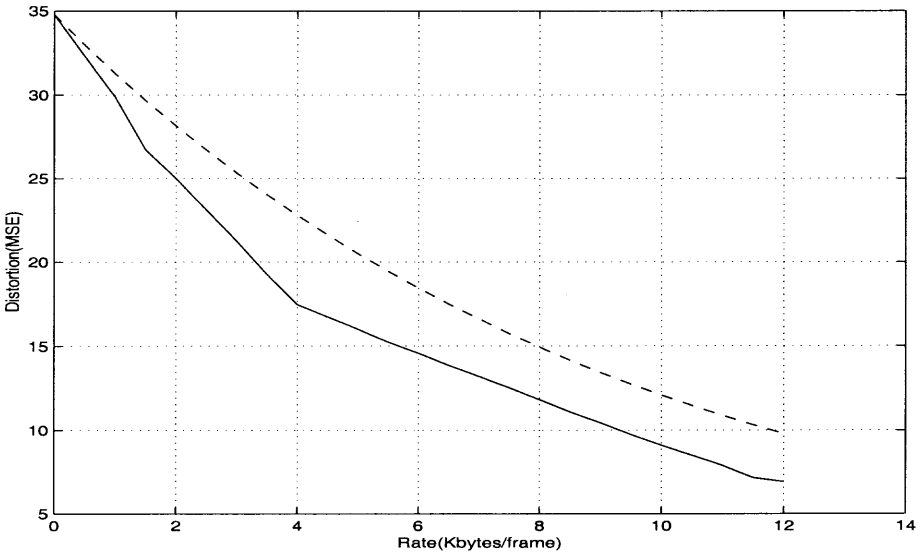


Figure 5.4 Actual R-D curve (solid) vs exponential model estimated R-D curve (dashed): Second B-frame in FGST layer of Foreman at CIF resolution.

are as follows. In FGS coding, the lower bit rate is obtained by truncating the enhancement layer. Cutting the bits within a bit-plane is equivalent to reducing the quality in only part of the frame. Therefore, after decoding, the quality is not uniform across parts of the frame. Hence, the exponential model is not accurate to describe the relationship between the rate (obtained by truncation) and the variance of the FGS residual component.

To overcome the problems described above, a set of R-D label parameters are proposed to approximate the complete R-D relationship. Piecewise interpolation between data points is used to approximate the curve. It should be noted that the R-D points corresponding to the base layer frames are first extracted. These points provide the starting points from which the R-D curves are formed. The validity of this approach is based on the following two assumptions: (a) the required R-D label parameters are achievable and (b) the R-D labels used have a low overhead. Given this, an efficient R-D label extraction scheme is presented for FGS and FGST coding.

In this scheme, the R-D information for the enhancement layer can be obtained either during the encoding process for real-time operation, or from stored bitstreams

after the entire video has been encoded [12]. Since the variance of the enhancement layer data is invariant of the DCT, the specified distortion can be obtained either in the DCT domain or in the spatial domain.

Figure 5.5(a) is a block diagram of a R-D extractor that determines rate and distortion sample points in the spatial domain. The enhancement layer bitstream is first passed through a bitstream controller. The function of this controller is to first determine the sample rate points. The sample rate points may be linearly spaced or determined according to a pre-specified function. The sample rate points are recorded as a first part of each R-D pair. Based on each rate point, the specified number of bits are used to reconstruct an FGS residual signal. The reconstruction is performed using a bit-plane VLD, bit-plane shift and IDCT. The reconstructed FGS residual is subtracted from the original FGS residual to yield an error signal. The distortion is then calculated based on the spatial domain error to yield a distortion sample point, which is the second part of each R-D pair. This process is repeated for multiple rate sample points to yield a set of R-D pairs.

Figure 5.5(b) is a block diagram of an alternate R-D extractor that determines rate and distortion sample points in the DCT domain. The process is similar to the process shown in Figure 5.5(a), except that no IDCT is taken to yield a reconstructed FGS residual in the DCT domain. This reconstructed FGS signal is subtracted from the FGS residual in the DCT domain and the DCT error is used to determine the corresponding distortion sample points.

Although a model that can accurately represents the R-D relationship of video coding has not been found yet, the exponential decreasing property of R-D relationship has been adopted. To capture this property, piecewise exponential interpolation can be used to estimate the R-D relationship. In the following, the performance of piecewise exponential interpolation and piecewise linear interpolation provided is compared. It will be shown that piecewise linear interpolation is

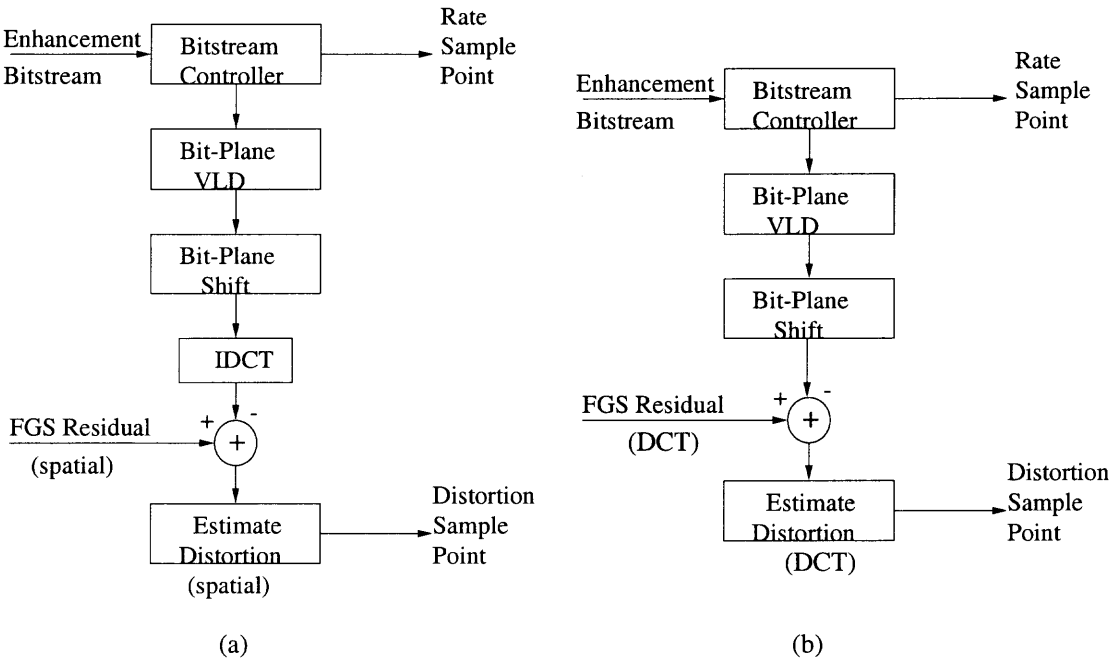


Figure 5.5 R-D extraction for FGS enhancement layer bitstreams: (a) spatial domain approach, (b) DCT-domain approach.

better than piecewise exponential interpolation with the probable selected sampling position and sampling interval. Only a small amount of side information is sufficient to accurately approximate the actual R-D curve.

Consider any two neighbor sampling point $D(R_m)$ and $D(R_n)$ along the R-D curve, where $R_m < R_n$. Let $\Delta R = R_n - R_m$ denote the difference between the two rates. The piecewise linear model is given by,

$$D_L(R) = D(R_m) - \frac{D(R_m) - D(R_n)}{\Delta R}(R - R_m) \tag{5.8}$$

where $D_L(R)$ is the distortion at point R, and $R_m \leq R \leq R_n$. Alternatively, using the piecewise exponential model to approximate the actual characteristics, the distortion is given by,

$$\begin{aligned} D_E(R) &= \sigma^2 e^{-\eta(R_m + R - R_m)} \\ &= D(R_m) e^{-\eta(R - R_m)} \end{aligned} \tag{5.9}$$

Table 5.1 Comparison of Model Accuracy Using Linear and Exponential Interpolation

Foreman						
Rates	R=5 kbpf	R=10 kbpf	R=20 kbpf	R=40 kbpf	R=80 kbpf	R=120 kbpf
MD_{lin}	2.93%	1.72%	2.81%	2.3%	2.2%	1.1%
MD_{exp}	3.73%	4.61%	8.01%	5.9%	16.2%	12.2%
AD_{lin}	0.72%	0.37%	1.23%	0.37%	0.94%	0.28%
AD_{exp}	1.38%	2.90%	5.88%	4.2%	9.4%	10.3%
Carphone						
Rates	R=5 kbpf	R=10 kbpf	R=20 kbpf	R=40 kbpf	R=80 kbpf	R=120 kbpf
MD_{lin}	2.02%	2.97%	2.72%	2.06%	1.33%	2.15%
MD_{exp}	3.71%	7.56%	9.22%	10.52%	8.43%	15.03%
AD_{lin}	0.76%	1.81%	0.37%	1.04%	0.74%	0.69%
AD_{exp}	1.86%	5.72%	3.94%	9.03%	6.06%	11.85%

(Note: MD denotes maximum difference, AD denotes average difference, and kbpf denotes Kbits per frame.)

where η is obtained from two neighboring sampling points. Given that $D(R_n) = D(R_m)e^{-\eta\Delta R}$, η can be calculated by

$$\eta = \frac{1}{\Delta R} log \frac{D(R_m)}{D(R_n)} \tag{5.10}$$

In order to compare the performance of exponential and linear interpolation, the Foreman and Carphone sequence is encoded at CIF resolution using FGS coding. The encoding frame rate is fixed at 30 fps, the interval ΔR=16 kb/frame and uniformly sample along the R-D curve. It is found that linear interpolation is better in some intervals, while exponential interpolation is better in others. The R-D curve is then sampled at the end of each bit-plane. Six different points (bit-rates) are selected at each frame to calculate the maximum and average differences between the approximated and actual values. The results for the first 100 frames of each sequence are summarized in Table 5.1.

The results in Table 5.1 demonstrate that the R-D relationship within each bit-plane is more linear than exponential. These results are consistent with both

sequences tested and over a wide range of bit-rates. It is believed that the reason for this phenomenon is due to the nature of the bit-plane coding method. In bit-plane coding, each quantized DCT coefficient is considered as a binary number of several bits instead of a decimal integer of a certain value [11]. For each 8×8 DCT block, the 64 absolute values are zigzag ordered into an array. A bit-plane of the block is defined as an array of 64 bits, where each element in the array corresponds to a particular bit position of the DCT coefficients. Thus, every bit in a bit-plane has the same impact on the distortion. With this type of coding scheme, it is reasonable to think the relationship between the number of bits and the distortion is linear within each bit-plane.

To summarize, it is found that linear interpolation is better than exponential interpolation when two neighboring points are located within same bit-plane. On the other hand, exponential interpolation is better than linear interpolation when two neighboring points are located in different bit-planes. Since the actual R-D lines have different gradients for different bit-planes, using linear interpolation to cross two bit-planes leads to a larger deviation than if exponential interpolation is used. In typical FGS coded frames, seven bit-planes are used. Therefore, eight sampling points are enough to approximate the R-D curve of each frame, which is a minor overhead.

It is noted that the sampling of R-D data followed by interpolation is seemingly similar to the technique of Lin and Ortega [51] to determine optimal quantization in video encoding. The key difference between their method and the proposed method is that the proposed method operates on bit-plane coded data that is produced by a FGS enhancement layer encoder and the method of Lin and Ortega operate on base layer data. To produce corresponding R-D points for base-layer data involves coding the input video with various quantizers and is very computationally demanding and not suitable for real-time. In contrast, the enhancement layer is coded using a bit-

plane coding method that produces an embedded bitstream from which R-D points can directly be extracted.

5.3 Constant-Quality Rate Allocation

The proposed R-D labeling scheme described in the previous section provides sufficient R-D information that can be used for rate allocation. Given this data, a similar cost function such as that given by Equation (5.2) may then be used by placing constraints to minimize the variation in quality among frames. In the literature, an exhaustive search is typically used to find the optimal solution.

As shown in Section 5.1, the solution that leads to constant quality is also the solution for minimizing the overall distortion provided the R-D relationship $D(R_i)$ of each frame is exponentially decreasing with the same constant in the exponent. Motivated by this observation, a practical rate allocation method is proposed. In the proposed method, instead of placing constraints to minimize the variation, constant quality is applied as the constraint. Based on this constraint and a suitable initial estimation of the constant distortion, the optimal solution can be obtained in one pass. In the following, the constant quality constrained rate allocation approach is first introduced. A sliding window technique is then used to adapt to the channel variation in time.

According to the piecewise linear interpolation scheme as described by Equation (5.8), the rate allocation can be calculated by,

$$\begin{cases} \sum_{i=0}^{N-1} R_i = N \times R_{budget}/F_s \\ D_{m_i} - (R_i - R_{m_i})\frac{\Delta D_i}{\Delta R_i} = D_{m_{i+1}} - (R_{i+1} - R_{m_{i+1}})\frac{\Delta D_{i+1}}{\Delta R_{i+1}}, \quad 0 \le i \le N-2 \end{cases} \tag{5.11}$$

where R_{budget} is the available bandwidth, N is the total number of frames, F_s denotes the source frame-rate, R_i is the optimal rate that should be allocated to frame i to achieve the constant distortion D. Let $\{R_{m_i}, D_{m_i}\}$ and $\{R_{n_i}, D_{n_i}\}$ be two adjacent R-D points such that $D_{m_i} \ge D \ge D_{n_i}$ and $R_{m_i} \le R_i \le R_{n_i}$. In the above equations, $\Delta R_i = R_{n_i} - R_{m_i}$ and $\Delta D_i = D_{m_i} - D_{n_i}$ represent the difference in rate and distortion

at adjacent R-D points, respectively. The above yields a set of N equations with N unknowns and can be solved by using known methods.

To solve the above equations, the correct interval should be determined first, namely, two correct adjacent R-D points. The most straightforward method is exhaustive search. However, it is time consuming. In the proposed method, the initial value of the constant distortion, D, is estimated first, then the two adjacent R-D points $\{R_{m_i}, D_{m_i}\}$ and $\{R_{n_i}, D_{n_i}\}$ are determined such that $D_{m_i} \geq D \geq D_{n_i}$. This initial estimation of D is calculated by using the extracted side information, i.e., $D = \frac{\sum_{i=0}^{N-1} D_i}{N}$, where D_i is the distortion associated with uniform bit allocation. This initial estimation is empirically found to be effective. Using the rate associated with uniform bit allocation provides a simple way to approximate the neighborhood within which an optimal rate is located.

5.3.1 Sliding Window Approach for Single FGS Source

Since the channel condition is changing with time, the available bandwidth for each frame is varying. Under this condition, a sliding-window resource allocation scheme is developed. Let the rate budget for a window of M frames beginning with frame a be denoted by, W_a , and be given by,

$$W_a = \begin{cases} M \times R_{budget}/F_s; & a = 0 \\ W_{a-1} - R_{a-1} + R_{budget}/F_s; & 1 \leq a \leq M-1 \end{cases} \tag{5.12}$$

where R_{budget} is the available bandwidth at time a and F_s denotes the source frame-rate. For each frame, the rate budget is computed and the rate allocation for the current frame is found based on the set of equations given below.

$$\begin{cases} \sum_{i=a}^{a+M-1} R_i = W_a \\ D_{m_i} - (R_i - R_{m_i})\frac{\Delta D_i}{\Delta R_i} = D_{m_{i+1}} - (R_{i+1} - R_{m_{i+1}})\frac{\Delta D_{i+1}}{\Delta R_{i+1}}, \quad a \leq i \leq a+M-2 \end{cases} \tag{5.13}$$

If the solution to the above equations is negative for frame i, let $R_i = 0$ and recompute the solution. Since the rate allocation to each window is changing on a

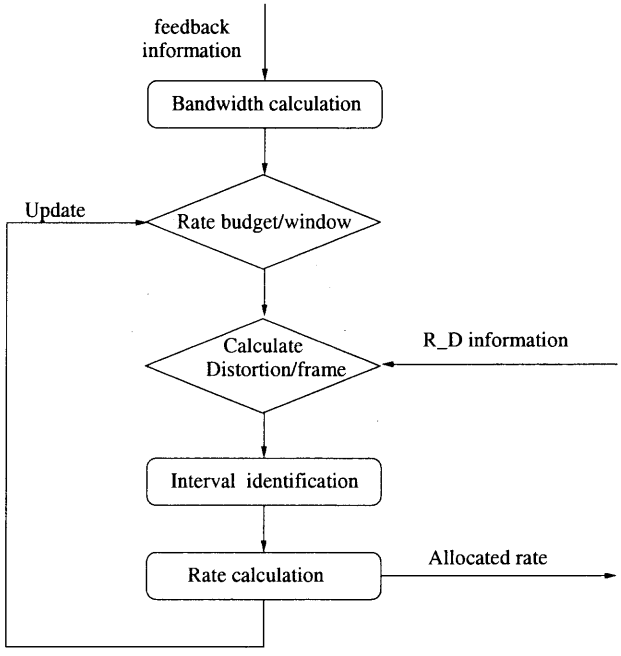


Figure 5.6 Flow diagram for sliding window method.

per frame basis, it is only need to solve the above function for the current frame R_i. In general, the proposed sliding window approach can be applied step by step according to Figure 5.6.

Although the computational complexity of above method is very low, the computational cost can be further reduced by computing the rate for every set of M frames, rather than on a per frame basis. In this way, the sliding window would move by M instead of 1 frame. At each pass, the rate allocated to each frame in the window would be assigned. This would work best for slowly varying channel conditions. Also, the smoothness can be improved by increasing the size of the sliding window. This can be observed from the calculation of the rate budget for the moving window as given by Equation (5.12). Each time the available bandwidth changes, the rate budget for the window is updated accordingly. Then, the influence of this variation is distributed to each of M frames within the current window. Statistically, each frame absorbs $1/M$ of the total variation. Thus, fluctuations between frames are expected to reduce to $1/M$ of the value without window.

With an increase of window size, the R-D information of more frames have to be known before the transmission. If the R-D information has been obtained off-line and stored, a rate control processor has instant access to this data. Since the computation complexity of the proposed method is very low $O(M)$, the computation delay can be ignored. Under the stable channel condition, it is desirable to select a larger window to smooth the fluctuations caused by varying scene complexity. On the other hand, if the channel condition is unstable, the smoothness is paid at the expense of an initial delay. In this case, a buffer is used to temporarily store the current M frames and adjust the bit-rate allocation among them. In a real application scenario, the window size can adaptively be determined based on the maximum variation among frames, the sensitivity to the initial delay and the target smoothness. The optimal solution will be a balance of these factors.

5.3.2 Sliding Window Approach for Multiple FGS Sources

In modern communication system, the server is usually connected to a wide-band network. The downstream channel is typically a CBR channel with high bandwidth. In the transmission of multiple sources over this high bandwidth network, the individual bitstreams are multiplexed and must satisfy a constant aggregate bit-rate. This problem is commonly referred to as the StatMux problem and has been studied in [59] for multiple encoding of MPEG-2 sources and in [60] for the transcoding of multiple MPEG-2 bitstreams.

In these works, the main objective was to utilize the high bandwidth link and maintain constant quality among the multiple sources, where each source is VBR coded and the sum of multiple VBR sources yields a constant aggregate bit-rate. The method that described is an extension of this previous work to scalable FGS coded bitstreams. The objective is still the same, but rather than considering rate

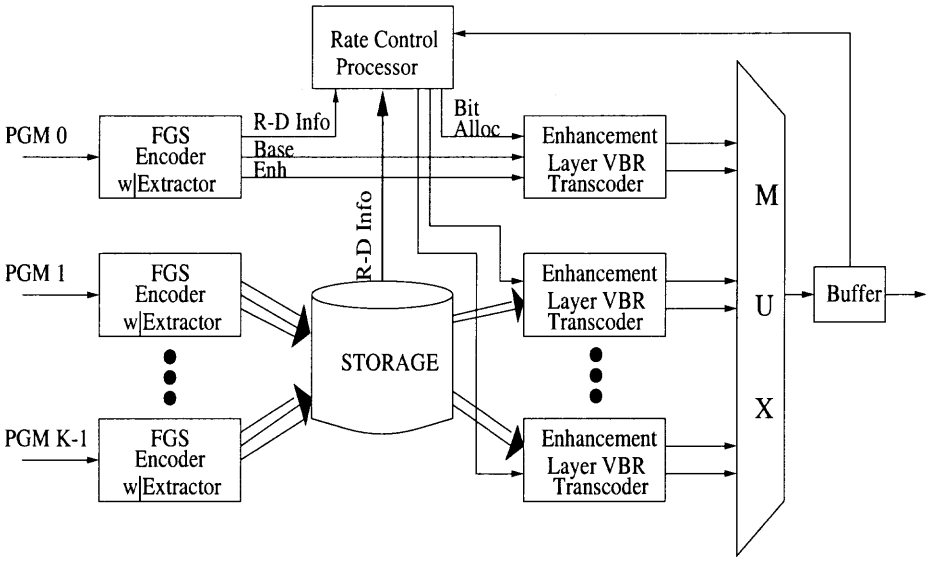


Figure 5.7 Block diagram of statistical multiplexer for multiple FGS coded sources.

allocation of a multiple base-layers, the allocation of bits in the enhancement layer is considered.

Figure 5.7 shows a block diagram of the proposed enhancement-layer statistical multiplexer. Each video program is subject the FGS encoder with extractor discussed earlier. This will output a base-layer bitstream, an enhancement layer bitstream and corresponding R-D information. For non-real-time applications, all the output is stored in a storage device. R-D information is sent to a rate control processor that provides rate allocation to each enhancement layer VBR transcoder. For real-time operation, the storage would be bypassed and base and enhancement layer bitstreams would be passed directly to the enhancement layer VBR transcoder. The reduced rate bitstreams are multiplexed, buffered and transmitted over a CBR channel. The buffer provides feedback to the rate control processor on the buffer fullness.

In the following, the operation of the rate control processor is introduced by extending the formulation described above for a single source to multiple sources. Similar assumptions are made. Namely, the minimum quality variance across the multiple sources lead to the minimum overall distortion. Let the rate budget for a

2D window of M frames and K sources beginning with frame b be denoted by, W_b, and be given by,

$$W_b = \begin{cases} M \times R_{budget}/F_s; & b = 0 \\ W_{b-1} - \sum_{j=0}^{K-1} R_{j,b-1} + R_{budget}/F_s; & 1 \leq b \leq M-1 \end{cases} \tag{5.14}$$

where R_{budget} is now the total bit budget for K sources and $R_{j,i}$ denote the bits used for source j at frame i. For each frame, the rate budget is computed and the rate allocation for the current frame is found using the set of equations given below,

$$\begin{cases} \sum_{j=0}^{K-1} \sum_{i=b}^{b+M-1} R_{j,i} = W_b; \\ D_{m_{j,i}} - (R_{j,i} - R_{m_{j,i}}) \frac{\Delta D_{j,i}}{\Delta R_{j,i}} = D_{m_{j+1,i}} - (R_{j+1,i} - R_{m_{j+1,i}}) \frac{\Delta D_{j+1,i}}{\Delta R_{j+1,i}}; \\ \qquad i = b, \quad 1 \leq j \leq K-2 \\ D_{m_{j,i}} - (R_{j,i} - R_{m_{j,i}}) \frac{\Delta D_{j,i}}{\Delta R_{j,i}} = D_{m_{j,i+1}} - (R_{j,i+1} - R_{m_{j,i+1}}) \frac{\Delta D_{j,i+1}}{\Delta R_{j,i+1}}, \\ \qquad b \leq i \leq b+M-2, \quad 0 \leq j \leq K-1 \end{cases} \tag{5.15}$$

where $\Delta R_{j,i} = R_{n_{j,i}} - R_{m_{j,i}}$ and $\Delta D_{j,i} = D_{m_{j,i}} - D_{n_{j,i}}$ represent the difference in rate and distortion at adjacent R-D points of source j, respectively. The above yields a M×K equations with M×K unknowns and can be solved for in a straightforward manner as the single source.

5.4 Experimental Results

To validate the effectiveness of the methods that have been described, the Foreman sequence (CIF resolution) is encoded using FGS and FGST coding. The encoding frame rate for the base layer is fixed at 10 fps for both FGS and FGST coding. Three rate allocation methods are tested: uniform bit allocation, Gaussian model based optimal bit allocation and the proposed method. For both the Gaussian model based method and the proposed method, N=3 is selected. Figure 5.8 shows the distortion for each frame corresponding to different rates, where each group of the first three consecutive frames in the sequence are compared and each bar denotes the distortion of the corresponding frame. Among the three bars, the gray bar denotes

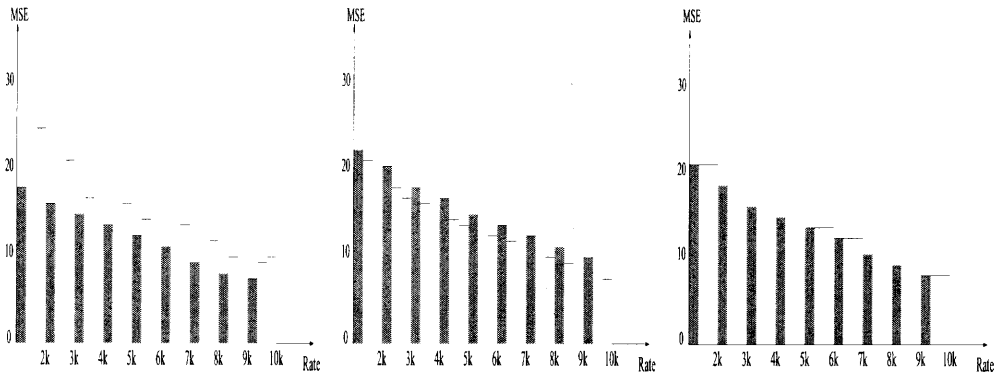


Figure 5.8 Simulation results comparing fluctuation among frames across bit-rates with three methods. Each group of three frames are compared and the gray bar denotes the first frame. (a) Uniform bit allocation, (b) Gaussian model based bit allocation, (c) Piecewise interpolation based bit allocation.

the first frame (I frame). For example, the first three bars in each figure demonstrate the distortions of the three frames correspond to rate 480kbps, which allows 2kBytes per fgs-vop and fgst-vop. It is evident from these plots that the proposed method can achieve constant quality across frames for a wide range of bit-rates.

Figure 5.9 and Figure 5.10 compare the proposed sliding window approach to uniform bit allocation. The base layer is encoded with two sets of quantization parameters and the enhancement layer is allocated a rate of 600 kbps, which allows 2.5 kBytes per fgs-vop and fgst-vop with uniform bit allocation. The distortion for each method is plotted over 100 consecutive frames and the results indicate that the quality becomes constant after only a few frames with the proposed approach, while the quality obtained by the uniform bit allocation method contains significant variation. Moreover, the average MSE distortion is decreased from 35.14 to 34.91 in (a), and decreased from 46.31 to 45.50 in (b).

To test the performance of the proposed sliding window approach corresponding to different window size under changing channel condition, the Coastguard sequence is encoded using FGS and FGST coding. All the parameters are same as above.

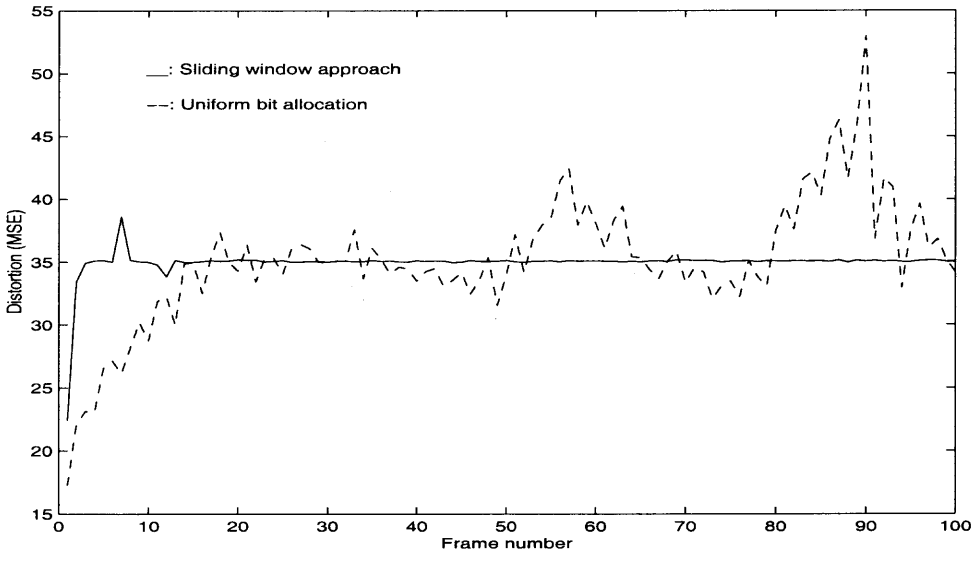


Figure 5.9 Simulation results comparing the distortion per frame using uniform bit allocation and the sliding window based method: Quality variation with quantization, I:10, P:22, B:15.

In this experiment, the same sequence is concatenated three times to generate a longer sequence that is 10s long. Figure 5.11 shows the channel condition used in the simulation. The bandwidth begins at 1920 kbps for the first 3.33s, drops to 1440 kbps for next 3.33s and then recovers back to 1920 kbps in for the final 3.33s. It should be noted that these values and changes in the bandwidth may be estimated from other channel conditions, such as a packet loss ratio or the occurrence of burst errors [61]. Figure 5.12 shows the result of the proposed sliding window approach for a window size equal to 0 frames (uniform bit allocation), 20 frames (0.67s), 60 frames (2s) and 160 frames (5.33s). The results indicate that the quality becomes smoother with the increase of the window size.

To test the performance of the proposed sliding window approach on multiple source rate allocation, the Foreman sequence, Coastguard sequence, Carphone sequence and Mobile sequence (all CIF resolution) are encoded using FGS and FGST coding. The encoding frame rate for the base layer is fixed at 10 fps for both FGS and FGST coding. Figures 5.13(a) and (b) compare the sliding window

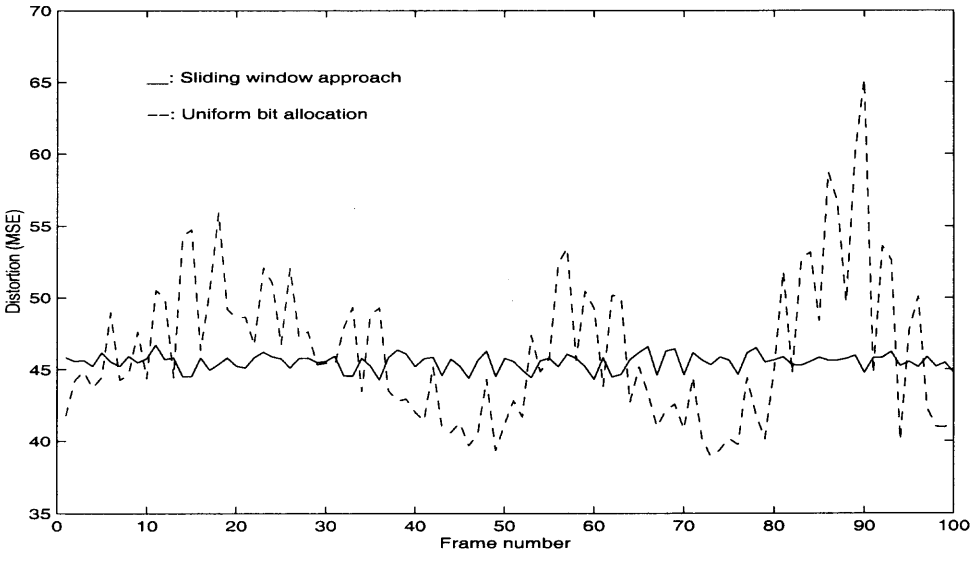


Figure 5.10 Simulation results comparing the distortion per frame using uniform bit allocation and the sliding window based method: Quality variation with quantization, I:30, P:30, B:30.

approach to uniform bit allocation. In the experiment, the sequences are encoded with quantization parameters (I:30,P:30,B:30) for the base-layer and FGS layer coding, and (I:28,P:28,B:28) for FGST layer coding. The enhancement layer is allocated a rate of 1320 kbps, which allows 5.5 kBytes per fgs-vop and fgst-vop with uniform bit allocation. The distortion for each method is plotted over 100 consecutive frames. The results using uniform bit allocation indicate that the sequences have significant quality difference. This is due to the different complexity associated with each sequences. Besides the inter-difference (among the sequences), the intra-fluctuations among the frames within the same sequence cannot be avoided

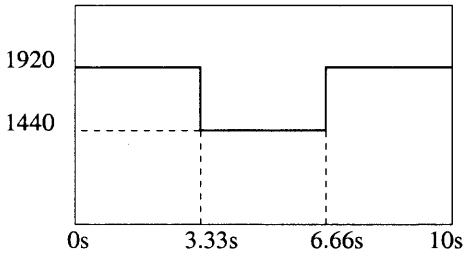


Figure 5.11 Dynamic channel condition illustrating available bandwidth (kbps).

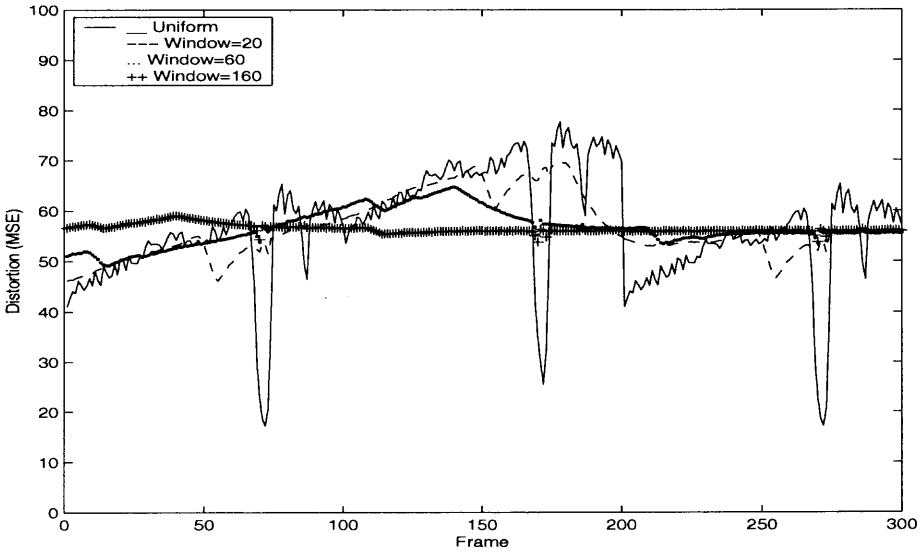


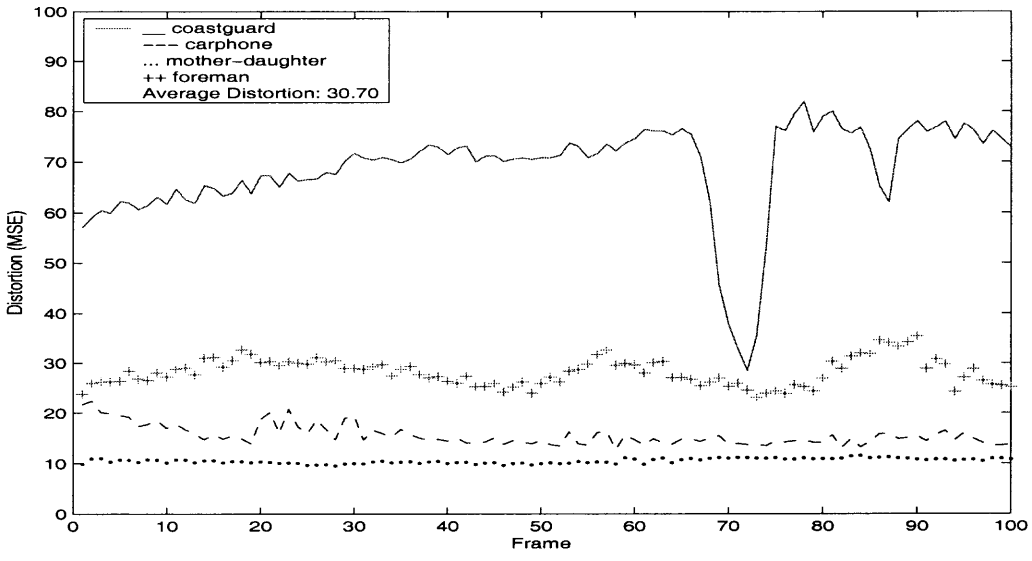
Figure 5.12 Simulation results to illustrate impact of window size. Larger window size results in less quality variation.

by uniform rate allocation. On the other hand, almost constant quality is obtained with the proposed approach. The average distortion is decreased from 30.70 to 27.66.

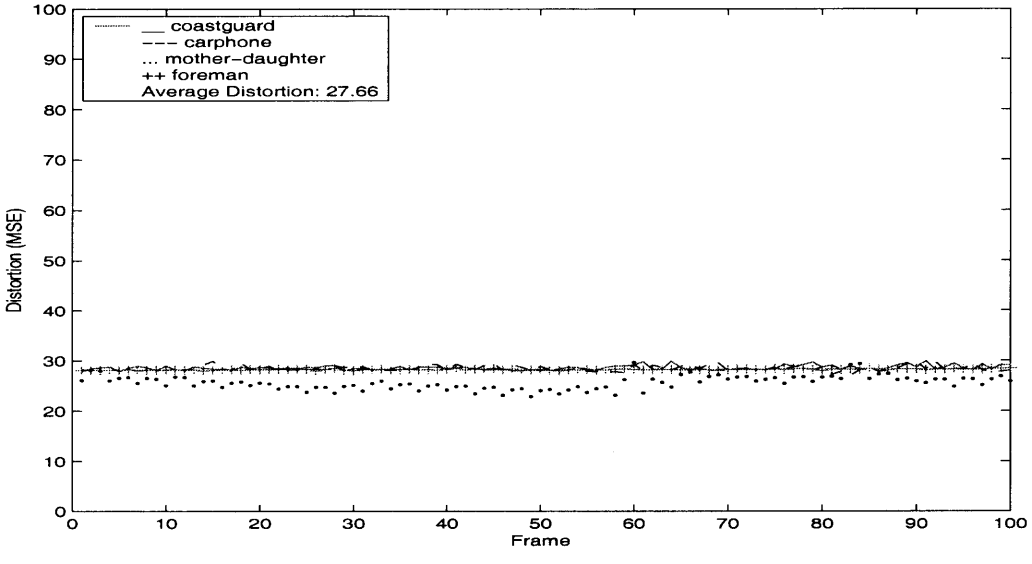
The performance of the proposed approach on multiple source is also tested under changing channel condition. The Foreman sequence, Coastguard sequence and Carphone sequence (all CIF resolution) are encoded by FGS and FGST coding. All the parameters and the channel are same as above. The 10s sequences are agained obtained by concatenating the same sequences three times. Figures 5.14(a) and (b) compare the sliding window approach to uniform bit allocation. The distortion for each method is plotted over 300 consecutive frames with the window size of 10s. The results indicate that the constant quality is obtained with the proposed approach, while the quality obtained by the uniform bit allocation method contains significant variation.

5.5 Summary

In this chapter, the problem of rate allocation for FGS coded video has been studied. Specifically, the techniques that are able to minimize the variation in quality have been proposed. An R-D labeling scheme was first proposed to characterize the R-D relationship of the source coding process. A set of actual R-D points are extracted during the encoding process and piecewise interpolation is used to estimate the actual R-D curve of the enhancement layer signal. The main contribution of this chapter is the sliding window based rate allocation method. With an initial estimate, the optimal rate allocation can be obtained in one pass with very low computation. The impact of the window size on the variation in the quality has been discussed in detail and supporting experimental results have been provided. Overall, the proposed framework is able to achieve constant quality reconstruction with both static and dynamic channel conditions. Simulation results show that smooth transitions in quality result under dynamic channel conditions can be obtained. Also, the effectiveness of this framework have been demonstrated for both single and multiple sources. Part of this research work has been published in [62].

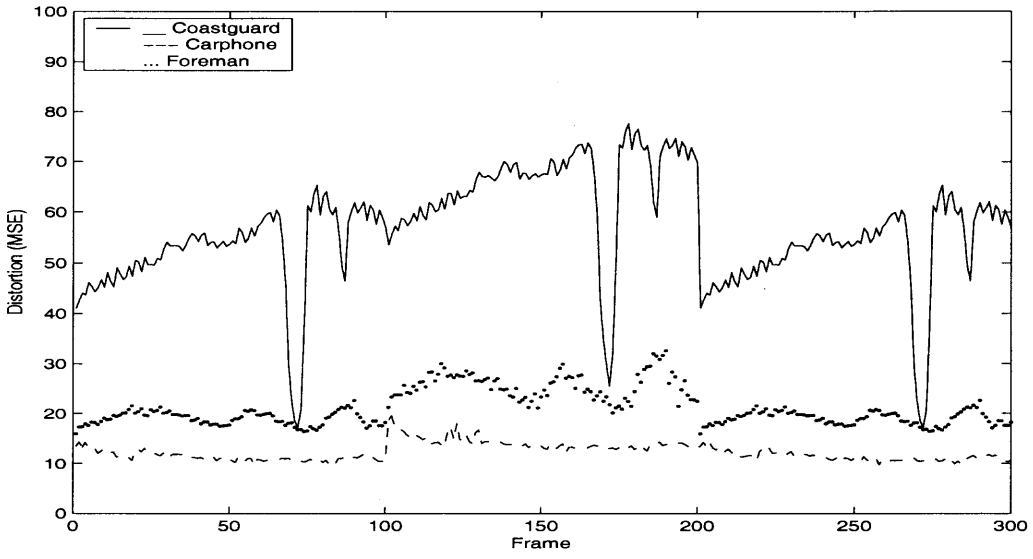


(a)

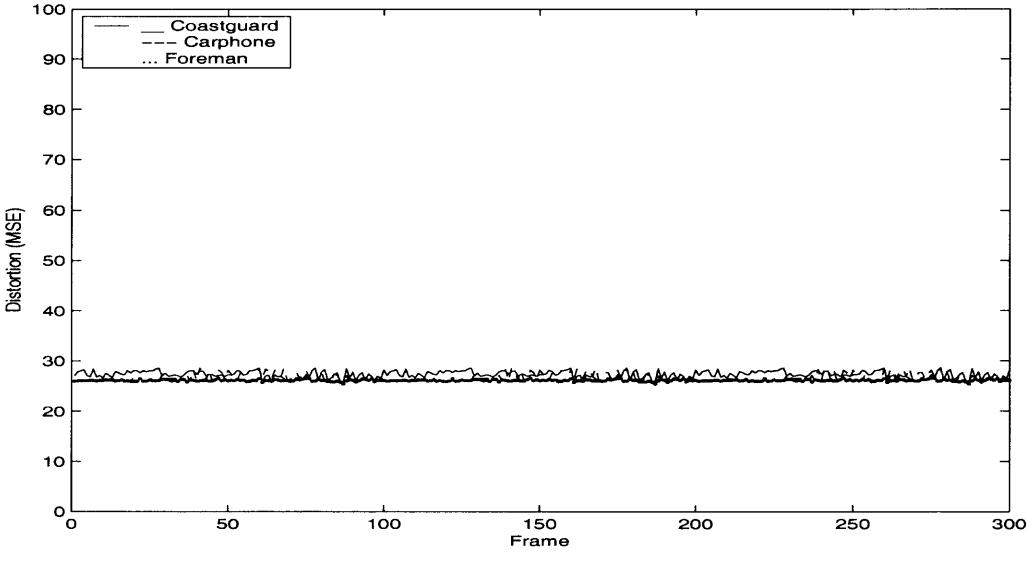


(b)

Figure 5.13 Comparison of quality variation with multiple source transmission and static channel bandwidth. (a) uniform bit allocation, (b) sliding window approach.



(a)



(b)

Figure 5.14 Comparison of quality variation with multiple source transmission and dynamic channel bandwidth. (a) uniform bit allocation, (b) sliding window approach.

CHAPTER 6

ERROR CONCEALMENT AND FEC FOR ROBUST MPEG-4 VIDEO TRANSMISSION

Robust video transmission is a challenging task mainly due to two factors: (1) limited channel bandwidth; (2) delay sensitivity of the video signal. In order to reduce the bit rate to satisfy the constraint of channel bandwidth, compression is necessary for efficient transmission of video. However, communication channels are not error free due to traffic congestion and interference. Also, since predictive coding techniques are typically used, an isolated bit error in one macroblock can propagate to other macroblocks. In order to solve this problem, many error resilient methods have been proposed. They can be categorized into three groups [63]: (i) error resilient encoding, in which the source and/or channel encoder insert bits to make the bit stream more resilient to potential errors [64, 65], (ii) error concealment at the decoder, which recovers or conceals lost information [66, 67], and (iii) interactive error control, which assumes a back-channel and allows the decoder to communicate losses to the encoder [68, 69].

In this chapter, the error resilient methods belonging to the first two groups are approached. The error control technique of forward error correction (FEC) for error resilient encoding is first studied. Specifically, a contribution vector based unequal error protection (UEP) method is proposed to protect the scalable coded video signal in Section 6.1. Then, error concealment methods for MPEG-4 coded video are investigated. An effective scheme is proposed to conceal the effect of error propagation in Section 6.2. This work is finally summarized in Section 6.3.

6.1 Unequal Error Protection of FGS Coded Video Using Contribution Vectors

To obtain optimal video/image transmission over the Internet, the sequence should be encoded according to the available channel bandwidth. However, the channel condition is usually unknown to the encoder, especially for stored video. In order to adapt to the dynamic channel conditions, transcoding can be applied to the encoded video streams by the multimedia server. The traditional decode-encode based transcoding is computationally expensive, so scalable coding techniques have been proposed to solve this problem. For example, with scalable coding, the video bitstream is divided into base-layer and enhancement layer. The base-layer can provide the basic video quality with a low bit rate. A video server may store one copy of the high quality video bitstream with both base-layer and enhancement layer and deliver only the base-layer and part of the enhancement layer depending on the client demand and channel conditions.

Even though the source rate can be adjusted according to the channel bandwidth variation by using scalable coding technique, the packet loss cannot be avoided due to network congestion, buffer overflow and random error. Two kinds of approaches can be used to solve this problem. One is re-transmission based close-loop method, such as automatic repeat request (ARQ). The other is feed-forward error correction (FEC) based open-loop method. The ARQ approach can guarantee reliable transmission at the cost of re-transmission delay. However, the video transmission is very sensitive to delay. Thus, the re-transmission based method is not suitable for streaming video applications.

In communication theory, random bit errors can be corrected by block or convolutional codes, which add redundancy to the data. FEC techniques are based on the same idea, namely, adding redundancy during transmission. Consider a video stream with equal importance for each part. In this case, equal error protection is the best way to combat random packet loss. For a scalable coded video stream, however, the

base layer is more important than the enhancement layer. If some data is dropped or damaged during transmission, it is preferable that the damage occurs within the enhancement layer rather than the base layer. Also, within the enhancement layer, damage to the least significant bits will not have as much of an impact as the most significant bits. In the worse channel condition, it assume that at least the base layer can be received correctly to guarantee minimum quality and no propagation of errors. To achieve this, unequal error protection (UEP) has been proposed [70]. Since FEC increases the source bit-rate, an optimal redundancy allocation between the base and enhancement layers is desirable. Applying different levels of FEC to the progressive compressed image has been investigated thoroughly [71], [72]. In [71], the research is focused on the graceful degradation of image quality with increasing packet loss. Their method use Reed-Solomon codes for erasure channel [73]. In [72], a hybrid coder that combines the source coding and channel coding is presented. Applying FEC-based UEP to video has also been studied in [74], where several fixed-level redundancies were used in their approach.

In this section, the investigation focus on the FEC-based UEP to FGS coded video streams. In Section 6.1.1, the FGS coding and FEC codes for erasure channel are introduced in general. Then, a novel concept referred to as a contribution vector is proposed to estimate the contribution of each redundant packet by its corresponding location. Based on this, an efficient equal contribution based algorithm is proposed to find the optimal UEP in Section 6.1.2. Simulation results are shown in Section 6.1.3.

6.1.1 Background

Recently, Fine-Granular Scalablilty (FGS) coding [11] has been proposed and adopted as amendments to the MPEG-4 standard [13]. FGS provides an enhancement layer that is continually scalable. This is accomplished through a novel bit-plane

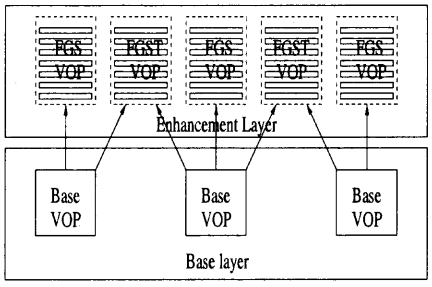


Figure 6.1 Principle of FGS coding.

coding method of DCT coefficients in the enhancement layer, which allows the enhancement layer bitstream to be truncated at any point. In this way, the quality of the reconstructed frames is proportional to the number of enhancement bits received. The principle of FGS coding is illustrated in Figure 6.1. Usually, a maximum of seven bit-planes are used for the enhancement layer. With FGS coded video, the base layer is more important than the enhancement layer, and within the enhancement layer, higher bit-planes are more important than lower bit-planes. So, it is natural to consider applying UEP to improve the performance.

In the IP network, each packet is assigned a unique sequence number. If a packet is lost, its position can easily be identified. For this kind of erasure channel, block code based FEC is very effective. The family of Reed-Solomon (RS) erasure-correction codes [73] have been widely used to recover the lost packets. Recently, a simple XOR-based FEC method has also been proposed [70]. Due to its maximum distance separable characteristics, RS erasure codes are selected to provide FEC. RS codes can be denoted as an $(n,\ k)$ code. Where n is the length of the block, and k is the length of the information data length within the block. So, one codeword contains length n data and length $n-k$ redundancy. By using this $(n,\ k)$ code, any data lost less than or equal to $n-k$ can be corrected.

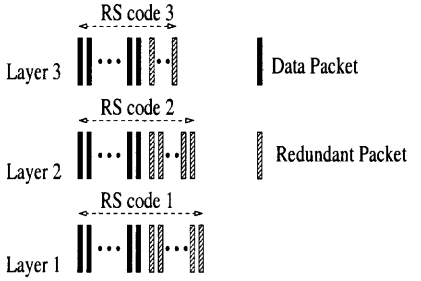


Figure 6.2 UEP framework for FGS coded video stream.

6.1.2 The Unequal FEC for FGS Coded Video

In this section, a UEP framework is first described for FGS coded video. Based on this framework, an equal contribution based redundancy allocation algorithm is then proposed, which is very effective and has low computational cost. Some issues are discussed towards the end.

According to Figure 6.1, the FGS coded video bitstream consists of one base layer stream and several bit-planes in an enhancement layer stream. To apply UEP to the enhancement layer stream, each bit-plane is thought of as a new layer with different importance. Equal length packetization is adopted. The FEC is realized by adding the redundant packets to each layer. In each layer, the data packets and redundant packets form a block. Assume there are M redundant packets in this block, then any packets loss less than M will not influence the correct data decoding. Figure 6.2 shows the framework for a stream with three layers. UEP is achieved by adding a different number of redundant packets according to the importance of each layer. It is noted Figure 6.2 only illustrates the ratio between data packets and redundant packets. The actual number of packets in each layer can be different. For this framework, UEP is claimed to be optimal if the expected quality is the best. If MSE is used as the criterion, minimization of MSE is the objective. If PSNR is used as the criterion, maximization of PSNR is the objective. It is noted that the above optimization is constrained by the overall target bit-rate.

In order to calculate the expected quality, a transmission policy corresponding to redundancy has to be determined. Consider a redundancy assignment with policy $R = \{r_1, r_2, \cdots, r_L\}$, where r_i, $1 \leq i \leq L$ is the length of FEC assigned to layer i. In previous work, the expected quality is first calculated based on this policy. Then, the policy is changed and the corresponding quality is calculated again and compared with the previous result. This procedure is repeated until the best result is obtained. For such a problem, the complexity based on the above brute force methods is $O(2^N)$. Mohr and et. al.[71] have proposed a local search method to reduce the complexity. However, their method is based on a greedy search algorithm that is sub-optimal. To further reduce the complexity and obtain the optimal result, an equal importance based rate allocation method is proposed.

6.1.2.1 Equal Contribution Based Redundancy Allocation. In the proposed approach, this problem is touched from a different view point. It will be shown that the computational cost is reduced considerably by using the proposed approach.

Definition 6.1.1. *In scalable coded video, the distortion will be reduced when more layers are received. The distortion reduction corresponding to each layer is called importance of this layer. If the layer is divided into a number of packets, then the expected distortion reduction corresponding to each packet is called contribution of this packet.*

Consider layer i with data length d_i, importance m_i and redundancy length r_i. If one more redundant packet is added to this layer, the contribution of this packet can be calculated by

$$\beta^{r_i+1} = m_i P(\frac{r_i+1}{d_i+r_i+1}) \tag{6.1}$$

where $P(\frac{r_i+1}{d_i+r_i+1})$ is the probability that r_i+1 packets among d_i+r_i+1 packets are lost. Changing the number of redundant packets from 1 to K, a contribution vector

for layer i can be obtained as

$$\beta_{\mathbf{i}} = \{\beta_i^1, \beta_i^2, \cdots, \beta_i^K\} \tag{6.2}$$

where β_i^k, $1 \leq k \leq K$ is the contribution of the kth redundant packet to layer i. Applying the above procedure to each layer, a set of contribution vectors corresponding to each layer is obtained. With the set of contribution vectors, the following result for UEP schemes can be obtained.

Definition 6.1.2. *Consider an UEP policy $R = \{r_1, r_2, \cdots, r_L\}$. If the overall expected distortion is minimum under a specified channel condition P, then this policy is called optimal under P.*

Theorem 6.1.1. *Consider a scalable coded stream. If the UEP policy $R = \{r_1, r_2, \cdots, r_L\}$ is optimal, $Min[\beta_1^{r_1}, \beta_2^{r_2}, \cdots, \beta_L^{r_L}] \geq Max[\beta_1^{r_1+1}, \beta_2^{r_2+1}, \cdots, \beta_L^{r_L+1}]$, where $\beta_i^{r_i}$ is the contribution of r_ith redundant packet to layer i.*

Proof. To prove theorem 6.1.1, first let $\phi_1 = Min[\beta_1^{r_1}, \beta_2^{r_2}, \cdots, \beta_L^{r_L}]$ and $\phi_2 = Max[\beta_1^{r_1+1}, \beta_2^{r_2+1}, \cdots, \beta_L^{r_L+1}]$ and assume $\phi_1 = \beta_{k_1}^{r_{k_1}}$ and $\phi_2 = \beta_{k_2}^{r_{k_2}+1}$. If $\phi_1 < \phi_2$, remove one redundant packet from layer k_1 and add one redundant packet to layer k_2. This operation does not change the overall bit rate. Now the net quality variation is the difference between the increased contribution $\beta_{k_2}^{r_{k_2}+1}$ and the decreased contribution $\beta_{k_1}^{r_{k_1}}$. Since $\beta_{k_2}^{r_{k_2}+1} - \beta_{k_1}^{r_{k_1}} > 0$, that means the operation improves the quality, which contradicts the claim that policy $R = \{r_1, r_2, \cdots, r_L\}$ is optimal. Hence,

$$Min[\beta_1^{r_1}, \beta_2^{r_2}, \cdots, \beta_L^{r_L}] \geq Max[\beta_1^{r_1+1}, \beta_2^{r_2+1}, \cdots, \beta_L^{r_L+1}]$$

Theorem 6.1.1 is proved. □

According to Theorem 6.1.1, the optimal UEP (minimization of the expected distortion) leads to the equal contribution of packets at the end of each layer, provided that the packets are small enough. Motivated by this result, instead of

trying to minimize the expected distortion, the equal contribution is used as the anchor to find the best result. The proposed method can be described by following procedure:

1. Allocate the redundant packets to each frame such that each frame has equal expected distortion.

2. For each frame, the corresponding redundant packets are first allocated to the base layer and each bit-plane based on the equal error protection. For example, Consider a frame which has N packets, among them, N_B belong to the base layer. If there are K redundant packets, then the number of packets allocated to the base layer will be $K\frac{N_B}{N}$.

3. Calculate the contribution of the last redundant packet of each layer and find the minimum of them, for example, $\beta_{k_1}^{r_{k_1}}$.

4. Calculate the contribution of adding one more redundant packet to each layer and find the maximum of them, for example, $\beta_{k_2}^{r_{k_2}+1}$.

5. Compare the minimum $\beta_{k_1}^{r_{k_1}}$ with maximum $\beta_{k_2}^{r_{k_2}+1}$. If $\beta_{k_1}^{r_{k_1}} < \beta_{k_2}^{r_{k_2}+1}$, remove one redundent packet from layer k_1 and add one redundant packet to layer k_2 such that the redundancy keeps constant. Then go back to step 3.

6. Repeat the above procedure until the condition in step 5 cannot be satisfied.

Using the above procedure, the optimal result can be obtained with complexity O(N).

6.1.2.2 Discussion. In the proposed equal contribution based methods, *importance* is used to calculate the contribution of each redundant packet. It denotes the MSE decreasement or PSNR increasement of the decoded bitstreams. For example, consider one frame with source variance E. The FGS coding is applied

to the source to generate an encoded stream. If the decoding distortion is D_B with only base layer, then the importance of the base layer is $E - D_B$. If the decoding distortion is D_1 with base layer and bit-plane 1, then the importance of bit-plane 1 is $D_B - D_1$. The importance of other bit-planes can be calculated accordingly. In [75], a R-D extraction scheme is proposed to obtain these values during encoding process. In FGS coded video, motion compensation is used to remove the temporal redundancy. It is obvious that I-frames are more important than P-frames, and P-frames are more important than B-frames. To estimate the importance of I- and P-frames, the following equation, $\delta_t^2 \frac{1-t/T}{1+\gamma t}$, can be used [76], where T is the width of the group of pictures, t is the distance to the I-frame, δ_t^2 corresponds to the source variance, and γ is a stream specific constant. The importance of B-frames can be estimated by the average of enhancement layer importance of the frames which it is predicted from.

Although this research is focused on the optimal UEP to FGS coded video streams, the equal contribution based redundancy allocation algorithm can be applied to any scalable coded streams, for example, progressive compressed images. Compared with the previous method, the complexity is reduced considerably.

6.1.3 Experimental Results

In this section, the performance of the proposed framework is evaluated by applying 30% FEC to the FGS coded video streams. The experiments are conducted under the condition that the video stream is transmitted through the channel where each packet loss is identical and independent. If each packet is lost with a probability of p, then the probability m of N transmitted packets are lost is

$$\binom{N}{m} p^m (1-p)^{N-m} \tag{6.3}$$

In this experiment, the proposed UEP algorithm is applied to the FGS coded Foreman stream with packet loss probability $p = 0.2$ and $p = 0.4$, respectively.

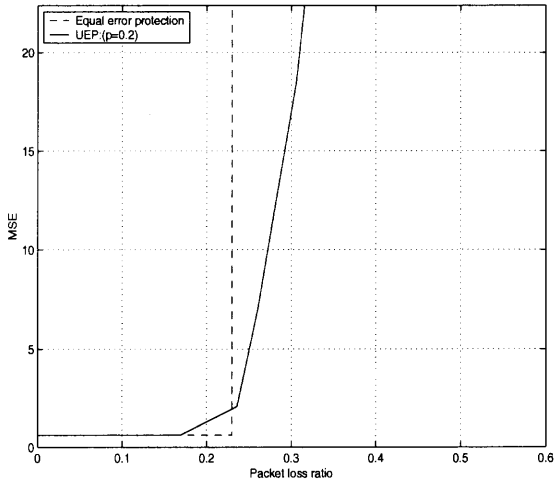


Figure 6.3 Performance comparison.

Since the lower layers only improve the quality slightly and need a large amounts of bits, only four bit-planes are used in this experiment. For $p = 0.2$, the FEC change to $\{0.463, 0.440, 0.353, 0.309, 0.204\}$; for $p = 0.4$, the FEC change to $\{1.02, 0.94, 0.82, 0.0, 0.0\}$, where each value is the ratio of $\frac{FEC}{Data}$. Figure 6.3 and Figure 6.4 show the decoding MSE of the first frame corresponding to different packet loss probabilities. It is seen that a graceful quality degradation is obtained if the UEP is optimized corresponding to a high packet loss probability. In spite of the low packet loss probability conditions, the UEP surpass the EEP in the sense of decoding quality.

6.2 Error Concealment Strategy for MPEG-4 Coded Video

It is well known that images of natural scenes have predominantly low-frequency components. Additionally, the human visual system is not sensitive to the high frequency components. Based on these observations, concealment methods have been investigated to reduce the effects of transmission error [77, 78, 79, 66, 67]

In all of the above methods, the spatial error propagation is assumed along one direction and can be localized by inserting a resynchronization block. However,

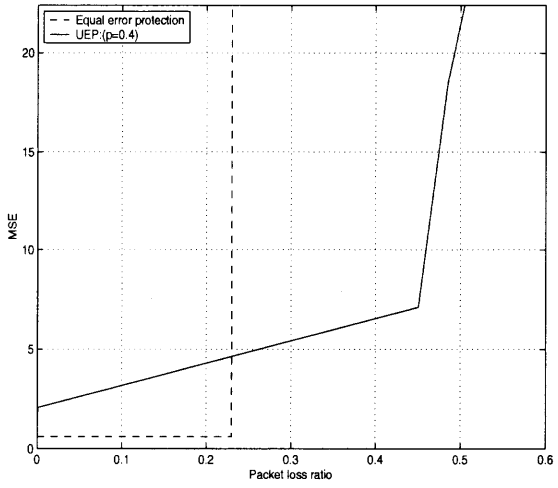


Figure 6.4 Performance comparison.

in MPEG-4, Intra DC/AC prediction has been adopted to increase the coding efficiency. This method allows the DC/AC coefficients of the corresponding block to be predicted from neighboring blocks, either to the left or above the current block. Thus, the error propagation cannot be avoided with resynchronization blocks.

In this section, a robust error-concealment method for reducing the error propagation is proposed. In this method, the smoothness of DC coefficient values and their gradients between the adjacent blocks are combined to estimate the DC value of the damaged blocks. Then, a weighted prediction is used to estimate the following affected blocks. A very important point to note is that small errors in the estimated DC values could lead to serious problems in deciding the direction that DC values are predicted from. To overcome this problem, the natural edges in the image should be distinguished from *error edges* that are due to error propagation. Overall, the proposed concealment method has very low computation complexity and can be realized on-line.

The rest of this section is organized as follows. The background on DC/AC prediction as well as a description of the problem are first reviewed in Section 6.2.1.

Then a robust decoding method is presented in Section 6.2.4. In Section 6.2.5, simulation results are demonstrated.

6.2.1 Problem Statement

Compared with MPEG-1/2, the DC prediction is allowed to adaptively select either the DC value of the left or above block in MPEG-4. This adaptive selection of the DC prediction direction is based on comparison of the horizontal and vertical DC gradients around the block to be decoded. The prediction of AC coefficients uses the same direction.

Figure 6.5 shows the three neighboring blocks A, B and C, surrounding the block to be decoded, X. Let $F_{m,n}[0]$ be the DC value of the current block X, then $F_{m-1,n-1}[0]$, $F_{m-1,n}[0]$ and $F_{m,n-1}[0]$ correspond to the reconstructed DC values of B, A and C, respectively. The gradients are then given by,

$$g_x = |F_{m,n-1}[0] - F_{m-1,n-1}[0]| \tag{6.4}$$
$$g_y = |F_{m-1,n}[0] - F_{m-1,n-1}[0]|$$

Then,

$$DC_{pred} = \begin{cases} F_{m,n-1}[0], & if \quad g_x > g_y \\ F_{m-1,n}[0], & if \quad g_x < g_y \end{cases} \tag{6.5}$$

Based on the above decisions, the DC value of the decoded block can be obtained,

$$F_{m,n}[0] = DC_{pred} + R_{m,n}[0] \tag{6.6}$$

where $R_{m,n}[0]$ is the inverse quantized residual DC value.

If any of the blocks A, B or C are outside of the VOP boundary, or the video packet boundary, or they do not belong to an intra coded macroblock, their $F[0]$ values are assumed to take a value of $2^{bits/per-pixel+2}$ and are used to compute the prediction values.

For the AC prediction, either coefficients from the first row or the first column of a previous coded block are used to predict the co-sited coefficients of the current

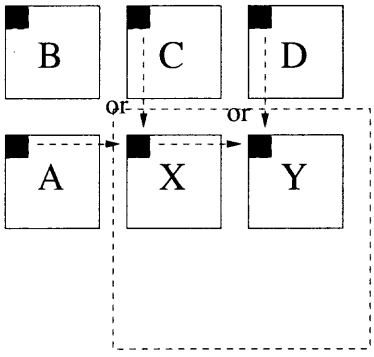


Figure 6.5 Illustration of DC prediction.

block. Again, the same direction that was derived for DC prediction is used. Note, within a macroblock it becomes possible to predict each block independently from either the horizontally adjacent block or the vertically adjacent block.

6.2.2 Analysis of Error Propagation

It should be clear from the previous section that it is possible for errors to not only propagate in the horizontal direction, but also the vertical direction. To analyze the error propagation, it is assumed the prediction directions are equally probable for both horizontal and vertical directions.

It is known that one macroblock is composed of four luminance blocks and two chrominance blocks. Let Y1 to Y4 denote the four luminance blocks according to raster order and U, V denote the chrominance blocks. Suppose one prediction block (left or above) of the current block is damaged. Since both directions are equally probable, the probability that a chrominance block is predicted by the error block is $P_U = P_V = 0.5$. For the luminance block, if the damaged macroblock is to the left, the error probability is as follows:

$$P_{Y1} = 0.5; \quad P_{Y2} = 0.25; \quad P_{Y3} = 0.75; \quad P_{Y4} = 0.5$$

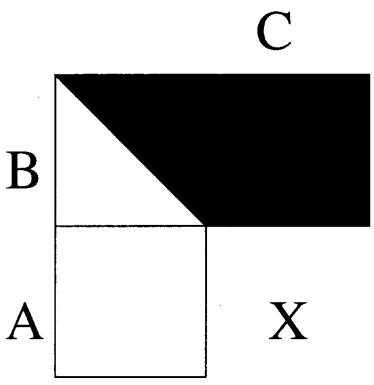


Figure 6.6 The condition lead to possible error edge.

To simplify the analysis, they are considered as a single block Y with error probability 0.5. Then, the right macroblock is affected by incorrect prediction with probability,

$$P = 1 - (1 - P_Y) \times (1 - P_U) \times (1 - P_V)$$

Thus, the expected error propagation length along the horizontal direction will be

$$E\{L\} = \sum_{n=0}^{\infty} nP^n \approx 56 \tag{6.7}$$

For CIF resolution, all the following macroblocks would be influenced. Moreover, for natural video, the transition between neighboring blocks is usually very smooth. However, according to eqn. (6.5), subsequent blocks may be predicted from an error block if the estimated gradients are in error. Hence, the error will propagate along one direction infinitely. The ultimate consequence is that all the blocks in the lower-right quadrant (those to the left and below) of the damaged block will be affected.

Intuitively, if the deviation is small, the influence will gradually disappear. However, due to the prediction, even a small deviation can lead to totally different result. For example, in Figure 6.6, an ideal condition that can lead to an error edge is shown. In this figure, the shaded areas have a gray-level value of 0 and the non-shaded areas have a gray-level value of 255. In this case, $g_x = g_y$ and one small deviation lead to serious problems since the prediction can come from an extremely

different DC value. In general, the prediction error will be equal to the difference, $F_{m-1,n}[0] - F_{m,n-1}[0]$, which can be very large.

6.2.3 Coding Efficiency

From the analysis above it can be seen that if an error in the DC coefficient occurs, it is very likely that error propagation will severely deteriorate the quality of the entire video. In order to avoid the influence of the error propagation, the DC/AC coefficient of the adjacent correct block can be used as the prediction. The drawback of this approach is that the prediction may be very poor. Another way is to restrict the direction of the prediction in the encoder to one direction only - the horizontal direction. In this way, the propagation will be the same as with previous standards and one may use resynchronization to localize the error. However, this will reduce the coding efficiency.

To understand this impact better, several video sequences have been encoded in the normal way, i.e., allowing prediction from both directions, and for comparison, in the restricted way, i.e., allowing horizontal prediction only. The sequences were encoded with fixed quantization parameters (QPs) equal to 3, 10 and 21 and N=30, M=3. Each sequence contains 300 frames, therefore the sequence being considered includes 10 I-frames. The results are summarized in Table 1. It is observed that the coding efficiency is moderately decrease for the case in which the prediction direction is restricted.

6.2.4 Robust MPEG-4 Decoder

In the proposed method, the focus is on the error concealment once the damaged macroblock has been identified. According to the location and characteristic of the blocks, they can be divided into four categories:

- correct block: all neighboring blocks have been reconstructed normatively without error.

Table 6.1 Comparison of Intra Coding Efficiency

Sequences	Q=3	Q=10	Q=21
Akiyo (restricted)	98037	40982	23542
Akiyo (normal)	91512	36853	21048
Carphone (restricted)	134470	50163	27063
Carphone (normal)	131262	47665	24944
Foreman (restricted)	177719	58909	29300
Foreman (normal)	176486	57698	28032

- damaged blocks: blocks that have corrupted residual values in the DC/AC coefficients.
- gradient error (type-I): either the block to the left or above has been obtained by estimation, hence, one gradient has error.
- gradient error (type-II): both blocks to the left and above of this have been obtained by estimation, hence, two gradient errors. This may also occur if only the diagonally neighboring block has been estimated and horizontal and vertical neighbors are correct.

In order to know the position and category of the corresponding blocks, a tracking function is used to record them. Once one block is reconstructed, its position and category are recorded for later reference.

According to Equation (6.6), the DC coefficient of the current block is determined by the residual value, R, and the prediction, DC_{pred}. Let's consider the damaged block first. Since it is damaged, no information for the residual value can be used. Its DC coefficient value can only be estimated from the block around it. There are lots of methods for the DC estimation. Among these methods, either they depend on complicated computation, or they use simple average values of the neighboring blocks. Considering the vertical and horizontal DC gradients, the

following metric is proposed to estimate the corrupted DC coefficient,

$$\tilde{F}_{m,n}[0] = \frac{F_{m,n-1}[0] \cdot g_x + F_{m-1,n}[0] \cdot g_y}{g_x + g_y} \tag{6.8}$$

In the above, more weighting is given to the DC coefficient that has a larger gradient with respect to $F_{m-1,n-1}[0]$. In the case that $g_x = g_y$, the above is simply an average of the two coefficients.

For a block with type-I gradient error, let $\tilde{F}_{m-1,n}[0]$ be the estimated value and $F_{m-1,n-1}[0]$ and $F_{m,n-1}[0]$ be the correct blocks. Since $\tilde{F}_{m-1,n}[0]$ is an estimated value, it is very possible that deviation exists and will influence the decision in Equation (6.5). However, even if the decision doesn't change, it is still possible for $F_{m,n-1}[0]$ to be the better prediction. Let $\tilde{F}_{m-1,n}[0] = F_{m-1,n}[0] + \Delta$, where Δ is the error. If the correct prediction direction is horizontal and the estimated prediction direction is also horizontal, then the error in the current block will be $-\Delta$. On the other hand, if the estimated prediction is vertical, then the error will be $F_{m-1,n}[0] - F_{m,n-1}[0]$. Therefore, if $|F_{m-1,n}[0] - F_{m,n-1}[0]| < |\Delta|$, $F_{m,n-1}[0]$ is always the better prediction, even though it is different from the direction derived in the encoder.

To solve above problem, more confidence is given to the correct block, $F_{m,n-1}[0]$. Then, the current DC coefficient values can be calculated as follows.

$$DC_{pred} = \begin{cases} F_{m,n-1}[0], & if \quad g_x + w > g_y \\ F_{m-1,n}[0], & if \quad g_x + w < g_y \end{cases} \tag{6.9}$$

where w is a positive confidence parameter. Similarly, if $\tilde{F}_{m,n-1}[0]$ is an estimated value and other blocks are correct, the confidence parameter w in Equation (6.9) is added to g_x, rather than g_y.

To choose w, the distance from the damaged block should be considered. For blocks in close proximity to the damaged block, we expect larger values of w. But as the distance to the damaged block gets further, the probability of error propagation gets less. Therefore, w becomes smaller. For 8-bit gray-level values, it is empirically found that the values of w between 20 and 2 are suitable.

For a block with type-II gradient error, the above method for type-I errors cannot be extended since it does not make sense to derive confidence values in this case. Thus, the normal procedure as in Equation (6.5) is used to reconstruct the DC coefficient for type-II blocks.

Due to the complicated relationship of AC coefficients among the neighboring blocks, it is difficult to estimate them. If the wrong direction is selected, both first-row and first-column will be affected. So the first-row and first-column of the immediately right and below macroblocks are kept unchanged and do not do any prediction, which can guarantee at least half correction. Since human eyes are not sensitive to the AC coefficient, it is reasonable to conjecture that their influence will disappear after several macroblocks. Thus, just the normal AC prediction is used for the other blocks.

According to the analysis in last section, the incorrect prediction direction can cause severe error. To overcome the possibility of predicting from the wrong direction, a smooth-oriented procedure is used for each block. In this procedure, the assumption of smoothness is used for the natural image. If $|g_x - g_y|$ is large, it is likely that the current block is in the vicinity of a natural edge and the deviation can not change the prediction direction; if $|R_{m,n}[0]|$ is very small, it is likely that the current block is in a smooth area; if both g_x and g_y are very small, the wrong direction will only have a minor impact. In any of these conditions, the current block does not need further processing. If none of these conditions are met, the following smooth-oriented procedure that switches the DC prediction direction is considered. If $DC_{pred} = F_{m-1,n}[0]$ and

$$|F_{m,n}[0] - F_{m,n-1}[0]| > |F_{m-1,n}[0] - F_{m,n-1}[0]|$$

then switch DC_{pred} to $F_{m,n-1}[0]$. This effectively switches the prediction from vertical to horizontal. Similar conditions also hold for switching from horizontal to vertical.

After this processing, it can be obtained

$$F_{m,n}[0] > \min\{F_{m-1,n}[0], F_{m,n-1}[0]\}$$
$$F_{m,n}[0] < \max\{F_{m-1,n}[0], F_{m,n-1}[0]\}$$

which accomplishes the smoothing effect.

6.2.5 Experimental Results

To test the effectiveness of the proposed method, the simulation is conducted under the assumption that bit errors have damaged all of the coefficient of one macroblock. The original and damaged I-frame of Akiyo is shown in Figure 6.8(a) and (b), respectively. It is seen that one damaged macroblock severely corrupts all the blocks in the lower-right quadrant. Figure 6.8(c) and (d) show the results of concealment for a damaged block using the proposed method without the smooth-oriented procedure and with smooth-oriented procedure. If the concealment method is only implemented without using smooth-oriented post processing, the error resulting from wrong prediction direction can cause the error edge. Combining every step of the proposed methods together, the quality is improved considerably. Although some distortion still exist, the improvement is obvious.

6.3 Summary

The problem of unequal error protection for FGS coded video stream is firstly investigated. A novel concept, contribution vector, is proposed to estimate the importance of each redundant packet by its corresponding location. By using the contribution vector, maximizing the decoding quality is associated with finding the redundant packets with suitable contributions. Based on this result, an equal contribution based algorithm is proposed to allocate the redundant packets. By using this method, the optimal allocation can be completed with computation complexity O(N). This algorithm can be applied to any scalable coded streams. Then a robust decoding

method is proposed to reduce the propagation of errors when DC coefficients have been damaged. In the proposed method, the damaged DC value is recovered by using neighboring DC coefficients. A novel emthod to estimate the better direction for DC/AC prediction has been presented. The simulation results demonstrate that propagation of errors is reduced considerably. Part of this research work has been published in [80] and [81].

(a) Original frame.

(b) Damaged frame.

Figure 6.7 Simulation results of Akiyo without concealment.

(a) Concealment without smooth-oriented procedure.

(b) Concealment with smooth-oriented procedure.

Figure 6.8 Simulation results of Akiyo with the proposed concealment method.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

In this dissertation, several issues about how to realize the robust and efficient video/image transmission have been studied. To combat the multi-dimensional (M-D) burst error during the transmission, an effective M-D interleaving method is proposed. Its performance is proved theoretically. Its application on turbo code interleaver design is presented. In order to utilize the bandwidth efficiently, a novel R-D labeling scheme is proposed to approximate the R-D relationship of FGS coded bitstreams. Based on the extracted R-D information, a sliding window approach is presented to realize constant quality constrained rate allocation. How to use forward error correction (FEC) code and error concealment strategy to realize error resilient MPEG-4 video transmission is also studied. To summarize, the main contributions made in this dissertation are:

- The problem of M-D interleaving in general has been studied. The problem has been formulated. The optimality issue of the M-D has been discussed.

- An effective two dimensional (2-D) interleaving method has been proposed for correcting 2-D error bursts. It has been shown that the proposed SP method obtains the upper bound for the square burst as $2^k \times 2^k$ and rectangle burst as $2^k \times 2^{k+1}$in the sense of interleaving degree. Moreover, this optimality holds for any $k < n$. So it avoids multiple rounds of computation in redetermining the interleaved array under different error bursts condition. In this sense, it is better than the existing state of art methods which need to be re-implemented for different burst size to maintain the optimality (Chapter 2).

- A novel concept of basis array has been proposed for M-D interleaving. The characteristics of basis array are analyzed. In the basis array, the minimum

distance among neighbouring elements are maximized. The basis array is then processed by the SP method to construct the arbitrarily size interleaved array according to the requirement. Thus, effective M-D interleaving is realized. It has been demonstrated and proven the superior performance of the proposed method in the 2-D case and generalized this result to the M-D case (Chapter 3).

- A successful turbo codes interleaver design method has been proposed based on the principal of successive packing. This method is optimal in the sense of pseudo-random distribution, prunability, and adaptivity to various criterion. Simulation results demonstrate the superior performance of the interleaver generated by the SP method compared to other deterministic interleavers that are generated by the other design techniques (Chapter 4).

- The R-D relationship modeling of encoded video bitstream has been studied. Specifically, a novel R-D labeling scheme is presented to approximate the real R-D curve of FGS coded bitstreams. In this scheme, the real R-D pair is sampled and used. Based on the author's study of the characteristics of bit-plane coding, it has been shown only few sampling pairs are enough to accurately estimate the entire R-D curve. The resulting overhead is minor (Chapter 5).

- An optimal rate allocation scheme has been proposed for FGS coded bitstreams. Instead of minimizing overall distortion, constant quality is utilized as a constraint. This utilization simplified the compution considerably. It has been shown that constant quality and the reduction of overall distortion can be obtained at the same time by the proposed methods. A sliding window approach is then presented to adapt to the variation of the channel or user condition dynamically (Chapter 5).

- An effective contribution vector based unequal error protection method has been proposed to protect the scalable coded video. The concept of contribution vector is presented. The optimility of the method is discussed. Compared with the existing state of the art methods, the computational cost of the proposed method is very low, while keeping the optimal performance (Chapter 6).

- A robust MPEG-4 video decoding scheme has been presented. Specifically, a novel error concealment strategy is proposed to combat the error propagation caused by the intra DC/AC prediction technique. In this strategy, the smoothness of DC coefficient values and their gradients between the adjacent blocks are combined to estimate the DC value of the damaged blocks. Then, a weighted prediction is used to estimate the following affected blocks. Good performance has been demonstrated by the experiments (Chapter 6).

7.2 Future Work

In this section, some possible directions are recommended for future research.

- *Application of SP based M-D interleaving method.*

  The author has proposed the SP based M-D interleaving method and successfully applied it to turbo code interleaver design. It is believed that the proposed method can be further used for various digital information protection applications such as M-D information hiding, image/video error concealement and etc. These issues will be investigated in the future.

- *Streaming video through network.*

  Video streaming is an important component of many Internet multimedia applications. However, the best-effort nature of the current Internet and delay sensitive nature of video pose many challenges. To address these challenges, it would be interesting to conduct research on application-layer QoS control. Congestion control and error control could be studied jointly. As the first step, all of the proposed techniques in this dissertation will be implemented within one framework, their perfromance will be investigated jointly.

- *Optimization of standard based compression techniques.*

  The current standards have not really been optimized due to the computational complexity and compatibility. However, for many storage applications, the video is encoded off-line. Therefore, the encoding process can endure longer time. On the other hand, the limited storage capacity is critical. Moreover, with the development of computer technology, existing algorithms can be executed faster than before. Based on the above observations, various optimization methods such as prefiltering and new rate contro scheme could be investigated. The focus will be also on how to embed them into the overall architecture of the current standards.

APPENDIX A

LEMMA AND THEOREM PROOF FOR CHAPTER 2

Proof of Lemma 2.3.2

Proof. The lemma is first proved for the square bursts first followed by a proof for the rectangular bursts. In either cases, the mathematical induction method is employed. Since the proof is trivial as $n = 0, 1$, and is very simple as $n = 2$, the proof is started as $n = 3$.

I1. When $n = 3$, A is of 8×8, as shown in Figure 7.1 (a). $k < n$ implies that $k = 0, 1, 2$. Since the proof for $k = 0$ is trivial, the lemma is only proved for $k = 1, 2$.

I2. Consider $k = 2$. There are 16 elements (code symbols) in each of this type of square bursts. To correct this type of error bursts with a one-random-error-correction code, the size of the codewords is 4. Look at the burst of 4×4, located at the top-left quadrant of A. In particular, look at the very northwest corner element, s_0. According to Definition 2.3.6, its 4-equivalent elements are: s_1, s_2, s_3, which are marked in Figure 7.1 (a). Similarly, each element in the burst has its own 4-equivalent elements, which follow the same distribution pattern as the 4-equivalent elements of s_0. Therefore, if the burst of 4×4, located at the top-left quadrant of A, is shifted toward the right by one column, it can be observed that the burst and its shifted version have three columns in common, while the first column in the burst (consisting of the elements $s_0, s_{48}, s_{12}, s_{60}$) has been replaced by the first column in the top-right quadrant (consisting of $s_2, s_{50}, s_{14}, s_{62}$). It is easy to see that elements in these two columns are 4-equivalent to each other. Hence, these two bursts are 4-equivalent according to Definition 2.3.7.

I3. This reasoning can be used for three more times (each time the burst of 4×4 is shifted towards the right by one column) and the conclusion can be made that the 4×4 burst located in the top-left quadrant is equivalent to each of its right-shifted versions.

I4. Applying the reasoning contained in I2 and I3 accordingly, it is obvious that the burst of 4×4, located in the top-left quadrant in A, is 4-equivalent to each of its down-shifted versions. After applying the same reasoning to all the down-shifted burst of 4×4 described in the previous sentence in considering its equivalence with its right-shifted versions, it can be concluded that any burst of 4×4 in A are 4-equivalent to each other. Thus, the proof of the lemma for any square bursts when $k = 2$ is completed.

I5. Consider $k = 1$. There are 4 elements in each of this type of square bursts. To correct this type of error bursts with a one-random-error-correcting code, the size of the codewords is 16. Look at the very northwest 2×2 burst in A. In particular, the very northwest corner element s_0. Figure 7.1 (a) shows all its 16-equivalent elements. Applying the reasoning used in the proof for $k = 2$ accordingly leads to the completion of the proof for $k = 1$.

I6. Having finished the proof for $n = 3$, it is assumed now that the lemma holds for any square bursts in the A when $n = N$. (Hence, A may be denoted by $S_{2^{2N}}$.) To complete the proof for the lemma to hold for any square bursts, according to the mathematical induction, it only need to show that the lemma holds for any square bursts in A when $n = N + 1$. (Note that at this time A may be denoted by $S_{2^{2N+2}}$.)

I7. To complete the proof for square bursts as $n = N + 1$, the lemma is proved for $k < N$ first, then for $k = N$.

I8. Consider $k < N$. When $n = N$, each element in the very northwest square burst of $2^k \times 2^k$ has $2^{2N-2k} - 1$ equivalent elements in $S_{2^{2N}}$ due to the assumption made in I6. When $S_{2^{2N+2}}$ is successively packed by using $S_{2^{2N}}$ according to Equation 2.3, the total number of the equivalent elements of each element in the very northwest square burst of $2^k \times 2^k$, located in the top-left quadrant of $S_{2^{2N+2}}$, has increased by four times. Meanwhile, the difference in the subscripts of any two equivalent elements in the top-left quadrant of $s_{2^{2N+2}}$ has increased by four times compared with that in the

$s_{2^{2N}}$. The increased equivalent elements are distributed evenly among the other three quadrants of $s_{2^{2N+2}}$ due to the successive packing (refer to Equation 2.3). Together with Definition 2.3.6, therefore, it is easy to see that the equivalent elements of any specific elements in the very northwest square burst of $2^k \times 2^k$ occupy the same positions in each of the four quadrants of $s_{2^{2N+2}}$. Furthermore, these positions are the same as that in the $s_{2^{2N}}$. Obviously, this observation is true for any elements in the very northwest burst of $2^k \times 2^k$. Therefore, the same reasoning used to prove the case of 8×8 can be applied accordingly to complete the proof for $k < N$.

I9. When $k = N$, the total number of the equivalent elements of each elements in the square burst located in the top-left quadrant is 4. Hence, the situation is similar to the square bursts of 4×4 when $n = 3$. The reasoning, contained in I2, can be used to prove that the lemma holds for $k = N$. Up to this point, the lemma has been proved for any square bursts in A.

In the following, the lemma will be proved for any rectangular bursts defined in Lemma 2.3.2.

II1. Consider $n = 3$. All the six possible types of rectangular bursts described in the lemma in this situation are: 1×2, 2×4, 4×8, 2×1, 4×2, and 8×4.

II2. Consider the rectangular bursts of 4×8 located at the top-half of the 8×8 array. Under the circumstances, the total number of the equivalent elements of any element in the rectangular burst is 2. The equivalent elements of s_0 and s_2 are s_1 and s_3 respectively. If two equivalent elements are linked with a line segments, it is found that these two line segments cross each other. It is noted, in general, that the line segments linking any element in the left-half of the burst of 4×8, located at the top-half of the 8×8 array, with its equivalent element will have the same orientation as that of the line segment linking s_0 and its equivalent element s_1, while the line segment linking any element in the right-half of the burst with its equivalent element has the same orientation with that linking s_2 and its equivalent

element s_3. Obviously, the burst of 4×8 at the top-half in A is equivalent to its one-row-down-shifted version. Repeat this procedure and use the transitive property of the equivalence relation, it can be derived that the burst is equivalent to any of its down-shifted version. This implies that all the bursts of 4×8 are equivalent to each other.

II3. Now prove the lemma for any burst of 2×4. At this time, the dimensionality of the equivalence is 8. Take a look at such a burst, which is located at the very northwest corner of the interleaved array of 8×8. All the 8-equivalent elements of the element s_0, together with s_0, are shown in Figure 7.1 (b) with the solid line segments linking them. It is seen that these eight elements form two zigzag patterns in the array. All the 8-equivalent elements of the element s_8, including the element s_8, are shown in Figure 7.1 (b) with the dashed line segments linking them. Note that the zigzag pattern associated with s_8 has just "opposite" orientation to that with s_0. It is easy to verify that all elements in the left-half of the burst of 2×4 have an equivalent element distribution pattern similar to that of s_0, while all elements in the right-half of the burst have an equivalent element distribution pattern similar to that of s_8.

Once this feature is identified, the reasoning used in the previous proof for the square bursts when $n = 3$ can be used accordingly to finish the proof for any burst of 2×4. For any burst of 1×2, the dimensionality of the equivalent elements for each element in the burst is 32. The similar reasoning to that used above leads to the completion of the proof for any burst of 1×2.

II4. Clearly, the proof for any rectangular burst errors of 2×1, 4×2, and 8×4 can be proved accordingly. Thus, the proof of the lemma is completed for any rectangular bursts when $n = 3$.

II5. Assume that the lemma holds for any rectangular bursts when $n = N$. To show that the lemma holds for any rectangular bursts when $n = N+1$, the same reasoning

as used in I7, I8 and I9 can be applied to the current circumstances accordingly to complete the proof at this step.

II6. Therefore, the whole proof of the lemma is completed.

□

Proof of Theorem 2.3.2

Proof. Due to Lemma 2.3.2, which establishes the equivalent-invariance with respect to burst translation for the types of error bursts under consideration, it only needs to prove the theorem for the types of bursts located at the very north-west corner of the 2-D interleaved array, A. The theorem is proved for the square bursts first followed by a proof for the rectangular bursts. According to the same reasoning in the proof of Lemma 2.3.2, the proof is started for $n = 3$ and $k = 1, 2$.

A1. Consider $k = 2$. Look at the very northwest corner element, s_0. According to Definition 2.3.6, its 4-equivalent elements are: s_1, s_2, s_3, which are marked in Figure 7.1 and are all outside of the square burst. Similarly, each element in the burst has its own 4-equivalent elements, which follow the same distribution pattern as the 4-equivalent elements of s_0. Hence, all elements in the burst will be spread in the de-interleaved 2-D array so that there is only one code symbol, which is in error, within each single codeword of size 4. Consider $k = 1$. There are 4 elements in the northwest square burst. It can be seen from Figure 7.1 that all s_0's 16-equivalent elements are outside of the square burst. Similarly, each element in the burst follow the same distribution pattern as the 16-equivalent elements of s_0. Therefore, it can be derived that all elements in the burst will be spread into each single codeword of size 16 in the de-interleaved 2-D array.

A2. After the proof for $n = 3$, the theorem is assumed to hold for any northwest square bursts in A when $n = N$ (denoted by $S_{2^{2N}}$). That is, as $n = N$, all the northwest square bursts of size $2^k \times 2^k$ with $k < N$ can be spread, where the corre-

sponding codeword size becomes 2^{2N-2k}. To complete the proof of the theorem for all the northwest square bursts, according to the mathematical induction, it only need to show that the theorem holds for all the northwest square bursts in A when $n = N + 1$ (denoted by $S_{2^{2N+2}}$).

A3. The theorem is first proved for $k < N$, then for $k = N$. Consider $k < N$. When $n = N$, s_0 has 2^{2N-2k} equivalent elements (including itself) in $S_{2^{2N}}$. Generating $S_{2^{2N+2}}$ by applying the successive packing to $S_{2^{2N}}$ according to Equation 2.3, s_0 has $2^{2N-2k+2}$ equivalent elements evenly distributed among $s_{2^{2N+2}}$ due to the successive packing (refer to Definition 2.3.6). That is, the equivalent elements within each of the four quadrants of $s_{2^{2N+2}}$ occupy the same positions. Hence, there is not any other $2^{2N-2k+2}$-equivalent element of s_0 within the square burst of $2^k \times 2^k$ in $S_{2^{2N+2}}$.

Obviously, the above observation is also true for any elements in the northwest burst of $2^k \times 2^k$. As mentioned above (in Definition 2.3.6 and meaning of Theorem 2.3.2), all the $2^{2N-2k+2}$-equivalent elements form a codeword. The distribution pattern of all the $2^{2N-2k+2}$-equivalent elements discussed above indicates that there is only one error element in each codeword. Therefore, the proof is finished for $k < N$.

A4. When $k = N$, the total number of the equivalent elements of each element in the northwest square burst is 3. Hence, the situation is similar to the square bursts of 4×4 when $n = 3$. The reasoning used in I2 can be used to prove the theorem for $k = N$.

A5. Up to this point the theorem has been proved for all the northwest square bursts in A. The use of Lemma 2.3.2 at this point leads to the proof for all the square bursts in A.

A6. Now, prove the theorem for the rectangular bursts. Consider $n = 3$. All the six possible types of the rectangular bursts in this situation are: 1×2, 2×4, 4×8, 2×1, 4×2, and 8×4. For the rectangular bursts of 4×8, located at the top-half of the 8×8 array. As shown in the proof of Lemma 2.3.2, the dimensionality of the

equivalence is 2, and the equivalent element of each element in the burst is located in the bottom-half of A. Hence, the burst can be spread. For the northwest burst of 2×4 in A. At this time, the dimensionality of the equivalent class is 8. It is seen that within each of the 8-equivalent class there is only one element is in error (refer to Figure 7.1 (b)). Hence, the burst can be corrected. Clearly, the proof for the northwest rectangular error bursts of 1×2, 2×1, 4×2, and 8×4 can be proved accordingly. Thus, the proof for all the northwest rectangular bursts when $n = 3$ is completed.

A7. Now assume that the theorem holds for any northwest rectangular bursts of $2^k \times 2^{k+1}$ or $2^{k+1} \times 2^k$ when $n = N$, then the theorem is proved for $n = N + 1$. The reasoning used in A3 can be applied to the current circumstances accordingly to complete the proof when $k < N$.

A8. When $k = N$, the proof for the burst of $2^N \times 2^{N+1}$ can be conducted with the same reasoning as that used in proving the burst of 4×8 in the interleaved 2-D array of 8×8 (i.e., $n = 3, k = 2$). The proof for the burst of $2^{N+1} \times 2^N$ can be conducted accordingly.

A9. Therefore, the proof can be concluded by using the mathematical induction that any northwest rectangular bursts of $2^k \times 2^{k+1}$ or $2^{k+1} \times 2^k$ can be corrected. The use of Lemma 2.3.2 leads to that any burst errors of $2^k \times 2^{k+1}$ or $2^{k+1} \times 2^k$ in A can be corrected.

Thus, the proof of the theorem is completed.

□

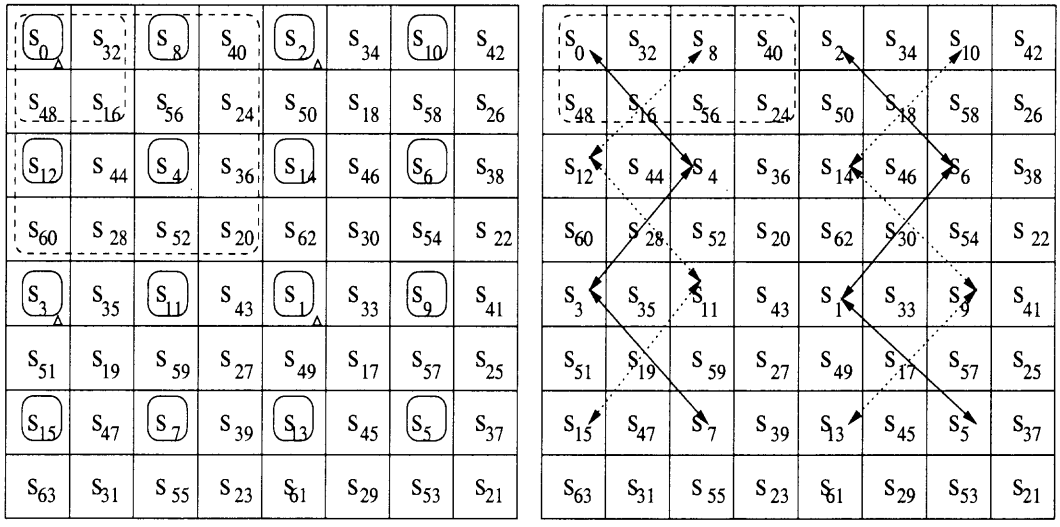


Figure 7.1 The square and rectangular equivalent element distribution patterns: (a) The 4-($s_0 - s_3$) and 16-($s_0 - s_{15}$) equivalent element distribution patterns, (b) The 8-equivalent element distribution patterns.

APPENDIX B

LEMMA AND THEOREM PROOF FOR CHAPTER 3

Proof of Lemma 3.2.1

Proof. The lemma is proved for the 3×3 bursts in 9×9 interleaving array first followed by a proof for the general case of $m_0^n \times m_1^n$ interleaving array. The mathematical induction method is employed.

I1. When $m_0 = 3$, $m_1 = 3$ and $n = 2$, A is of 9×9, as shown in Figure 7.2. $k < n$ implies that $k = 0, 1$. Since the proof for $k = 0$ is trivial, the lemma is only proved for $k = 1$.

I2. Consider $k = 1$. There are 9 elements (code symbols) in each of this type of square bursts. To correct this type of error bursts with a one-random-error-correction code, the size of the codewords is 9. Look at the burst of 3×3, located at the top-left block of A. In particular, look at the very northwest corner element, s_0. According to Definition 3.1.2, its 9-equivalent elements are: $s_1, s_2, \cdots, s_8$, which are marked in Figure 7.2. Similarly, each element in the burst is the initial element of a distinct 9-equivalent elements set. It has its own 9-equivalent elements, which follow the same distribution pattern as the 9-equivalent elements of s_0. For instance, the 9-equivalent elements set initialized by s_{18}. Therefore, if the burst of 3×3, located at the top-left block of A, is shifted toward the right by one column, it can be observed that the burst and its shifted version have two columns in common, while the first column in the burst (consisting of the elements s_0, s_{63}, s_{45}) has been replaced by the first column in the top-middle block (consisting of s_3, s_{66}, s_{48}). It is easy to see that elements in these two columns are 9-equivalent to each other. Hence, these two bursts are 9-equivalent according to Definition 3.2.1.

I3. This reasoning can be used for six more times (each times the burst of 3×3 is shifted towards the right by one column) and it is easy to see that the 3×3 burst located in the top-left block is equivalent to each of its right-shifted versions. The

conclusion that the 3×3 burst located in the top-left block is equivalent to each of its down-shifted versions can be obtained accordingly. After applying the same reasoning to all the down-shifted burst of 3×3 described in the previous sentence in considering its equivalence with its right-shifted versions, it can be obtained that any burst of 3×3 in A are 9-equivalent to each other. Thus, the proof of the lemma for any 3×3 bursts for 9×9 array is completed.

I4. Applying the reasoning contained in I2 and I3 accordingly, it is derived that any burst of $m_0 \times m_1$, located in $m_0^2 \times m_1^2$ array A, is $m_0 m_1$-equivalent to each other.

I5. Having finished the proof for $n = 2$, now assume that the lemma holds for any mentioned bursts in the A when $n = N$ (A may be denoted by $S_{(m_0 m_1)^N}$). To complete the proof for the lemma, according to the mathematical induction, it only need to show that the lemma holds for any mentioned bursts in A when $n = N + 1$ (Note that at this time, A may be denoted by $S_{(m_0 m_1)^N}$).

I6. To complete the proof for $n = N + 1$, the lemma is proved first for $k < N$, then for $k = N$.

I7. Consider $k < N$. When $n = N$, each element in the very northwest rectangle burst of $m_0^k \times m_1^k$ has $(m_0 m_1)^{N-k} - 1$ equivalent elements in $S_{(m_0 m_1)^N}$ due to the assumption made in I6. When $S_{(m_0 m_1)^{N+1}}$ is successively packed by using $S_{(m_0 m_1)^N}$ according to Procedure 3.1.1, the total number of the equivalent elements of each element in the very northwest square burst of $m_0^k \times m_1^k$, located in the top-left block of $S_{(m_0 m_1)^{N+1}}$, has increased by $m_0 \times m_1$ times. Meanwhile, the difference in the subscripts of any two equivalent elements in the top-left block of $S_{(m_0 m_1)^{N+1}}$ has increased by $m_0 \times m_1$ times compared with that in the $S_{(m_0 m_1)^N}$. The increased equivalent elements are distributed evenly among the $m_0 \times m_1 - 1$ blocks of $S_{(m_0 m_1)^{N+1}}$ due to the successive packing (refer to Procedure 3.1.1). Together with Definition 3.1.2, it is easy to see that the equivalent elements of any specific elements in the very northwest square burst of $m_0^k \times m_1^k$ occupy the same positions in each of the

$m_0 \times m_1 - 1$ blocks of $S_{(m_0 m_1)^{N+1}}$. Furthermore, these positions are the same as that in the $S_{(m_0 m_1)^N}$. Obviously, this observation is true for any elements in the very northwest burst of $m_0^k \times m_1^k$. Figure 7.3 illustrates the distribution of the equivalent elements of s_0 in $m_0^n \times m_1^n$ array, where circle is used to mark the $m_0 \times m_1$ equivalent elements of s_0, and square is used to mark the $m_0^{n-1} \times m_1^{n-1}$ equivalent elements of s_0. Therefore, the same reasoning used to prove the case of 9×9 can be applied accordingly to complete the proof for $k < N$.

I8. When $k = N$, the total number of the equivalent elements of each elements in the square burst located in the top-left block is $m_0 \times m_1$. Hence, the situation is similar to the square bursts of $m_0^2 \times m_1^2$ when $n = 2$. The reasoning contained in I2-I4 can be used to prove that the lemma holds for $k = N$. The Lemma 3.2.1 is now proved. □

Proof of Theorem 3.2.1

Proof. Due to Lemma 3.2.1, which establishes the equivalent-invariance with respect to burst translation for the types of error bursts under consideration, it only need to prove the theorem for the types of bursts located at the very north-west corner of the 2-D interleaving array, A. According to the same reasoning in the proof of Lemma 3.2.1, the proof is started for $n = 2$ and $k = 1$.

A1. Consider $k = 1$. Look at the very northwest corner element, s_0. According to Definition 3.1.2, its $m_0 \times m_1$-equivalent elements are: $s_1, \cdots, s_{m_0 m_1 - 1}$, which are are all outside of the $m_0 \times m_1$ burst. Similarly, each element in the burst has its own $m_0 \times m_1$-equivalent elements, which follows the same distribution pattern as the $m_0 \times m_1$-equivalent elements of s_0. Hence, all elements in the burst will be spread in the de-interleaving 2-D array so that there is only one code symbol, which is in error, within each single codeword of size $m_0 \times m_1$. Therefore, it can be concluded that all

elements in the burst will be spread into each single codeword of size $m_0 \times m_1$ in the de-interleaving 2-D array.

A2. After the proof for $n = 2$, now assume that the theorem holds for any northwest square bursts in A when $n = N$ (denoted by $S_{(m_0 m_1)^N}$). That is, as $n = N$, all the northwest square bursts of size $m_0^k \times m_1^k$ with $k < N$ can be spread, where the corresponding codeword size becomes $(m_0 m_1)^{N-k}$. To complete the proof of the theorem for all the northwest square bursts, according to the mathematical induction, it only need to show that the theorem holds for all the northwest square bursts in A when $n = N + 1$ (denoted by $S_{(m_0 m_1)^{N+1}}$).

A3. The theorem is proved first for $k < N$, then for $k = N$. Consider $k < N$. When $n = N$, s_0 has $(m_0 m_1)^{N-k}$ equivalent elements (including itself) in $S_{(m_0 m_1)^N}$. Generate $S_{(m_0 m_1)^{N+1}}$ by applying the successive packing to $S_{(m_0 m_1)^N}$ according to Procedure 3.1.1, s_0 then has $(m_0 m_1)^{N-k+1}$ equivalent elements. Due to the successive packing, the minimum distance between s_0 and any other elements in the northwest rectangle incease by $m_0 m_1$ times. Hence, there is not any other $(m_0 m_1)^{N-k+1}$-equivalent element of s_0 within the northwest rectangle burst of $m_0^k \times m_1^k$ in $S_{(m_0 m_1)^{N+1}}$.

Obviously, the above observation is also true for any elements in the northwest burst of $m_0^k \times m_1^k$. As mentioned above (in Definition 3.1.2 and meaning of Theorem 3.2.1), all the $(m_0 m_1)^{N-k+1}$-equivalent elements form a codeword. The distribution pattern of all the $(m_0 m_1)^{N-k+1}$-equivalent elements discussed above indicates that there is only one error element in each codeword. Therefore, the proof for $k < N$ is finished.

A4. When $k = N$, the total number of the equivalent elements of each element in the northwest square burst is $m_0 \times m_1 - 1$. Hence, the situation is similar to the square bursts of 3×3 in 9×9 array when $n = 2$. The reasoning used in I3 can be used to prove the theorem for $k = N$.

A5. Up to this point, the theorem has been proved for all the northwest square bursts in A. The use of Lemma 3.2.1 at this point leads to the proof for all the square bursts in A.

Thus, the proof of the theorem is completed.

□

s_0	s_{27}	s_{54}	s_3	s_{30}	s_{57}	s_6	s_{33}	s_{60}
s_{63}	s_9	s_{36}	s_{66}	s_{12}	s_{39}	s_{69}	s_{15}	s_{42}
s_{45}	s_{72}	s_{18}	s_{48}	s_{75}	s_{21}	s_{51}	s_{78}	s_{24}
s_7	s_{34}	s_6	s_1	s_{28}	s_{55}	s_4	s_{31}	s_{58}
s_{70}	s_{16}	s_4	s_{64}	s_{10}	s_{37}	s_{67}	s_{13}	s_{40}
s_{52}	s_{79}	s_{25}	s_{46}	s_{73}	s_{19}	s_{49}	s_{76}	s_{22}
s_5	s_{32}	s_{59}	s_8	s_{35}	s_{62}	s_2	s_{29}	s_{56}
s_{68}	s_{14}	s_{41}	s_{71}	s_{17}	s_{44}	s_{65}	s_{11}	s_{38}
s_{50}	s_{77}	s_{23}	s_{53}	s_{80}	s_{26}	s_{47}	s_{74}	s_{20}

Figure 7.2 The 9-equivalent element distribution patterns in 9×9 interleaving array.

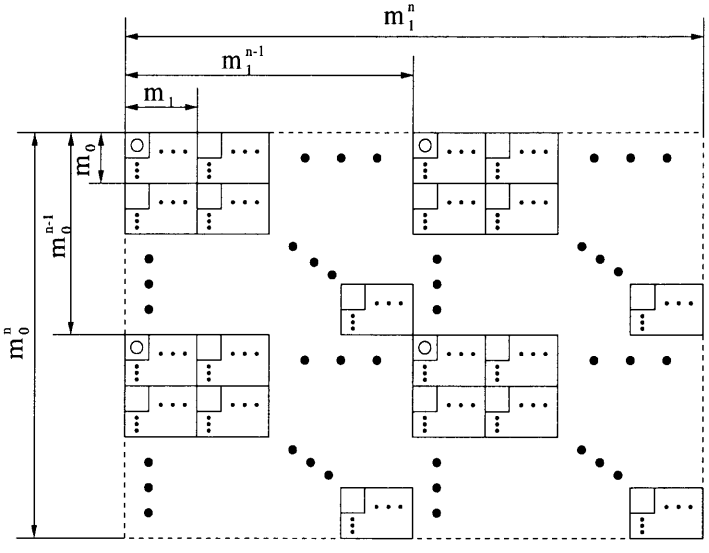


Figure 7.3 The general equivalent element distribution patterns in $m_0^n \times m_1^n$ interleaving array.

REFERENCES

1. N. Färber, B. Girod, and J. Villasenor, "Extensions of ITU-T Recommendation H.324 for Error-Resilient Video Transmission," *IEEE Communications Magazine*, pp. 120–128, June 1998.

2. S. B. Wicker, *Error Control System for Digital Communication and Storage.* Englewood Cliffs, NJ: Prentice-Hall, Inc, 1995.

3. J. F. Heanie, M. C. Bashaw, and L. Hesselink, "Volume holographic storage and retrieval of digital data," *Sci. Mag.*, vol. 265, pp. 749–752, 1994.

4. "Maxicode briefing document," *UPS Inc, http://www.maxicide.com*, March 1996.

5. N. L. Passos, E. H.-M. Sha, and L.-F. Chao, *Multi-Dimensional Interleaving for Synchronous Circuit Design Optimization.* Dept of ECE, Iowa State University: Research Report, CSE-TR-94-035, Nov. 1994.

6. D. Brady and D. Psaltis, "Control of volume holograms," *J. Opt. Soc. Amer.*, vol. 9A, pp. 1167–1177, 1992.

7. J. Huang and P. Schultheiss, "Block quantization of correlated Gaussian random variables," *IEEE Transaction on Communication Systems*, vol. 11, pp. 289–296, Sep.1963.

8. W. Ding and B. Liu, "Rate control of MPEG video coding and recording by rate-quantization modeling," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 6, no. 1, pp. 12–19, 1996.

9. K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders," *IEEE Transaction on Image Processing*, vol. 3, no. 5, pp. 533–545, 1994.

10. N. Jayant and P. Noll, *Digital Coding of Waveforms.* Englewood Cliffs, NJ: Prentice Hall, 1984.

11. W. Li, "Overview of Fine Granularity Scalability in MPEG-4 video standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 301–317, 2001.

12. M. van der Schaar and H. Radha, "A hybrid Temporal-SNR Fine-Granular Scalability for Internet video," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 318–331, 2001.

13. ISO/IEC 14496-2:1999/FDAM 4, *Information technology-Coding of audio-visual objects-Part 2:Visual Amendment 4: Streaming video profile.* N3904, Jan. 2001.

14. A. Puri and T. Chen, *Multimedia Systems, Standards, and Networks.* New York, NY: Marcel Dekker, Inc, 2000.

15. S. Lin, D. J. Costello, and M. J. Miller, "Automatic repeat error control achemes," *IEEE Communications Magazine*, pp. 5–17, 1984.

16. E. Steinbach, N. Färber, and B. Girod, "Standard Compatible Extensions of H.263 for Robust Video Transmission in Mobile Enviornments," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 7, no. 6, pp. 872–881, 1997.

17. H. Imai, "Two-Dimensional Fire Codes," *IEEE Transactions on Information Theory*, vol. 19, no. 6, pp. 796–806, 1973.

18. K. A. Abdel-Ghaffar, R. J. Mceliece, and H. C. Tilborg, "Two-Dimensional Burst Identification Codes and Their Use in Burst Correction," *IEEE Transactions on Information Theory*, vol. 34, no. 3, pp. 494–504, 1988.

19. C. de Almeida and R. Palazzo Jr., "Two-Dimensional Interleaving Using the set Partitioning Technique," *Proceedings. IEEE Int. Symp. Information Theory*, p. 505, 1994.

20. K. A. Abdel-Ghaffar, "Achieving the Reiger Bound for Burst Errors Using Two-Dimensional Interleaving Schemes," *IEEE ISIT 1997*, p. 425, 1997.

21. M. Blaum, J. Bruck, and A. Vardy, "Interleaving Schemes for Multidimensional Cluster Errors," *IEEE Transactions on Information Theory*, vol. 44, no. 2, pp. 730–743, 1998.

22. J. Pjatek, "Automotive industry recommends 2-D standards," *Automatic I. D. News*, pp. 54–56, 1994.

23. M. Blaum, J. Bruck, and P. G. Farrell, "Two-Dimensional Interleaving Schemes with Repetitions," *Proceedings. IEEE Int. Symp. Information Theory*, p. 342, 1997.

24. B. Chor, C. E. Leiserson, and R. L. Rivest, "An Application of Number Theory to the Organization of Raster-Graphics Memory," *Journal of the Association for Computing Machinery*, vol. 33, no. 1, pp. 86–104, 1986.

25. S. Golomb and L. Welch, "Perfect Codes in the Lee metric and the packing of polyominnoes," *SIAM J.Appl. Math.*, vol. 18, no. 2, pp. 302–317, 1970.

26. G. F. Elmasry and Y. Q. Shi, "An interleaving technique for error correction in 2-D bar coding," *Proceedings of Thirty-Sixth Annual Allerton Conference on Communication, Control, and Computing*, pp. 49–58, 1998.

27. G. F. Elmasry, *Detection and Robustness of Digital Image Watermarking Signals: A Communication Theory Approach.* Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ: Ph.D Dissertation, 1999.

28. Y.-Q. Shi and X. M. Zhang, "Two-Dimensional Interleaving by Using Successive Packing," *38th Annual Allerton Conference on Communication, Control and Computing*, pp. 945–954, Oct. 2000.

29. Y. Q. Shi and X. M. Zhang, "A New Two-dimensional Interleaving Technique Using Successive Packing," *IEEE Trans. Circuit and Systems: Part I.* , vol. 49, no. 6, pp. 779–789, 2002.

30. X. M. Zhang, Y. Shi, and S. Basu, "Basis array based successive packing approach for M-D interleaving," *Fifteenth International Symposium on Mathematical Theory of Networks and Systems (MTNS) 2002*, University of Notre Dame, Aug. 2002.

31. C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes," *Proceedings of ICC'93*, pp. 1064–1070, May 1993.

32. J. Hagenauer, E. Offer, and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Trans. Information Theory*, vol. 42, no. 2, pp. 429–445, 1996.

33. R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo Decoding as an Instance of Pearl's Belief Propagation Algorithm," *IEEE Journal on Select Areas of Communications*, vol. 16, no. 2, pp. 140–152, 1998.

34. M. Moher and T. Gulliver, "Cross-Entropy and Iterative Decoding," *IEEE Trans. Information Theory*, vol. 44, no. 7, pp. 3097–3104, 1998.

35. J. Yuan, B. Vucetic, and W. Feng, "Combined Turbo Code and Interleaver Design," *IEEE Trans. Commun*, vol. 47, pp. 484–487, Apr. 1999.

36. H. R. Sadjadpour, N. J. A. Sloane, M. Salehi, and G. Nebe, "Interleaver Design for Turbo Codes," *IEEE Journal on Select Areas of Communications*, vol. 19, no. 5, pp. 831–837, 2001.

37. J. Hokfelt, O. Edfors, and T. Maseng, "Interleaver design for turbo codes based on the performance of iterative decoding," *Proc. IEEE ICC*, vol. 1, pp. 93–97, June 1999.

38. S. Benedetto and G. Monotorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Trans. Information Theory*, vol. 42, no. 2, pp. 409–428, 1996.

39. A. Barbulescu and S. Pietrobon, "Interleaver Design for Turbo Codes," *Electron. Lett*, vol. 30, no. 25, pp. 2107–2108, 1994.

40. H. Herzberg, "Multilevel turbo coding with short interleavers," *IEEE Journal on Select Areas of Communications*, vol. 16, no. 2, pp. 303–309, 1998.

41. M. Eroz and A. R. Hammons Jr., "On the design of prunable interleavers for Turbo Codes," *Proceedings of VTC'99*, pp. 1669–1673, May 1999.

42. O. Y. Takeshita and D. J. Costello, "New Determinitic Interleaver Designs for Turbo Codes," *IEEE Trans. Information Theory*, vol. 46, no. 6, pp. 1988–2006, 2000.

43. EIA/TIA IS-2000. 1-6., *Standards for CDMA2000 Spread Spectrum Systems.*

44. M. Z. Wang, A. Sheikh, and F. Qi, "Interleaver Design For Short Turbo Codes," *Global Telecommunications Conference*, vol. 1b, pp. 894–898, 1999.

45. X. M. Zhang, Y. Shi, A. M. Haimovich, H. Chen, A. Vetro, and H. Sun, "Successive Packing Based Interleaver Design for Turbo Codes," *IEEE International Symposium on Circiuts and Systems (ISCAS) 2002*, Phoenix, 2002.

46. T. Chiang and Y.-Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 186–199, 1999.

47. A. Vetro, H. Sun, and Y. Wang, "MPEG-4 rate control for coding multiple video objects," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 186–199, 1999.

48. H. Zheng and K. J. R. Liu, "The subband Modulation: A Joint Power and Rate Allocation Framework for subband Image and Video Transmission," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 9, no. 5, pp. 823–838, 1999.

49. M. Gallant and F. Kossentini, "Rate-Distortion optimized layered coding with unequal error protection for robust Internet video," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 357–372, 2001.

50. J. Li and S. Lei, "An embedded still image coder with rate-distortion optimization," *Proceedings of SPIE: VCIP98*, vol. 1, pp. 36–47, 1998.

51. L.-J. Lin and A. Ortega, "Bit-Rate Control Using Piecewise Approximated Rate-Distortion Characteristics," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, no. 4, pp. 446–459, 1998.

52. L. Zhang and H. Fu, "A novel scheme of transporting pre-stored MPEG video to support video-on-demand (VoD) services," *Computer Communications*, vol. 23, pp. 133–148, 2000.

53. M. Hamdi, J. W. Robers, and P. Rolin, "Rate control for VBR video coders in Broad-Band networks," *IEEE Journal on Select Areas of Communication*, vol. 16, no. 6, pp. 1040–1051, 1997.

54. J. Shapiro, "Embeded image coding using zerotree of wavelet coefficients," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3445–3462, 1993.

55. D. Taubman and A. Zakhor, "Multirate 3-D Subband Coding of Video," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 572–588, 1994.

56. F. Wu, S. Li, and Y.-Q. Zhang, "A Framework for Efficient Progressive Fine Granularity Scalable Video Coding," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 332–344, 2001.

57. Q. Wang, F. Wu, S. Li, Z. Xiong, Y.-Q. Zhang, and Y. Zhong, "A New Rate Allocation Scheme for Progressive Fine Granular Scalable Coding," *The IEEE International Symposium on Circuits and Systems*, 2001.

58. Y.-Q. Shi and H. Sun, *Image and Video Compression for Multimedia Engineering - Fundamentals, Algorithms, and Standards.* Boca Raton, FL: CRC Press LLC, 1999.

59. L. Wang and A. Vincent, "Joint rate control for multi-program video coding," *IEEE Trans. on Consumer Electronics*, vol. 42, no. 3, pp. 300–305, 1996.

60. H. Sorial, W. Lynch, and A. Vincent, "Joint transcoding of multiple MPEG video bitstreams," *Proc. Int'l Symp. Circuits and Syst.*, 1999.

61. D. Wu, Y. T. Hou, W. Zhu, Y.-Q. Zhang, and J. M. Peha, "Streaming Video over the Internet: Approaches and Directions," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 282–300, 2001.

62. X. M. Zhang, A. Vetro, Y. Shi, and H. Sun, "Constant quality constrained rate allocation for FGS coded video," *IEEE Trans. Circuits and Systems for Video Technology, Accepted.*

63. Y. Wang, S. Wenger, J. Wen, and A. Katsaggelos, "Error Resilient Video Coding Techniques," *IEEE Signal Processing Magazine*, pp. 61–82, July 2000.

64. J. Wen and J. Villasenor, "Reversible variable length codes for robust image and video transmission," *1997 Asilomar Conference*, Pacific Grove, CA, Nov. 1997.

65. P. Haskell and D. Messerschmitt, "Resynchronization of motion compensated video affected by ATM cell loss," *Proc. ICASSP*, vol. 3, no. 545-548, San Francisco, CA, 1992.

66. S. Shirani, F. Kossentini, and R. Ward, "A concealment method for video communications in an error-prone environment," *IEEE Journal on Select Areas of Communications*, vol. 18, no. 6, pp. 1122–1128, 2000.

67. P. Salama, N. Shroff, and E. Delp, "Error concealment in MPEG video streams over ATM networks," *IEEE Journal on Select Areas of Communications*, vol. 18, no. 6, pp. 1129–1144, 2000.

68. B. Girod and N. Färber, "Feedback-based error control for mobile video transmission," *Proc. IEEE*, vol. 87, no. 1707-1723, Oct. 1999.

69. G. Reyes, A. Reibman, S. Chang, and J. Chuang, "Error-Resilient Transcoding for Video over Wireless Channels," *IEEE Journal on Select Areas of Communications*, vol. 18, no. 6, pp. 1063–1074, 2000.

70. J. Rosenberg and H. Schulzrinne, *An RTP payload format for generic forward error correction.* IETF Audio/Video Transport Working Group, Aug. 1999.

71. A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Unequal Loss Protection: Graceful degradation of image quality over packet erasure channels through forward error protection," *IEEE Journal on Select Areas of Communications*, vol. 18, no. 6, pp. 818–828, June 2000.

72. P. C. Cosman, J. K. Rogers, P. Greg Sherwood, and K. Zeger, "Combined Forward Error Control and Packetized Zerotree Wavelet Encoding for Transmission of Images Over Varying Channels," *IEEE Tran. on Image Processing*, vol. 9, no. 6, pp. 982–993, June 2000.

73. S. Wicker, *Erro Control Systems for Digital Communications and Storage.* Englewood Cliffs, NJ: Prentice-Hall, 1995.

74. M. Gallant and F. Kossentini, "Rate-Distortion Optimized Layered Coding with Unequal Error Protection for Robust Internet Video," *IEEE Tran. on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 357–372, Mar. 2001.

75. X. M. Zhang, A. Vetro, Y. Shi, and H. Sun, "Constant Quality Rate Allocation for FGS Video Coded Bitstreams," *IS&T/SPIE symposium on Visual Communications and Image Processing*, pp. 817–827, CA, 2002.

76. K. Stuhlmüller, N. Färber, M. Link, and B. Girod, "Analysis of Video Transmission over Lossy Channels," *IEEE Journal on Select Areas of Communications*, vol. 18, no. 6, pp. 1012–1032, June 2000.

77. W. Kwok and H. Sun, "Multi-directional interpolation for spatial error concealment," *IEEE Trans. Consumer Electron.*, vol. 39, pp. 455–460, Aug. 1993.

78. Y. Wang, Q. Zhu, and L. Shaw, "Maximally smooth image recovery in transform coding," *IEEE Trans. Commun.*, vol. 39, pp. 1544–1551, Oct. 1993.

79. H. Sun and W. Kwok, "Concealment of damaged block transform coded images using projections onto convex sets," *IEEE Trans. Image Processing*, vol. 4, pp. 470–477, Apr. 1995.

80. X. M. Zhang and Y. Shi, "Equal Contribution Based Unequal Error Protection for FGS Coded Video," *The 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI) 2002*, Orlando, Florida.

81. X. M. Zhang, A. Vetro, H. Sun, and Y. Q. Shi, "Robust Decoding for Reduced Error Propagation of DC/AC Prediction Errors," *IEEE 2nd Workshop and Exhibition on MPEG-4*, pp. 91–94, CA, 2001.