CAPACITY APPROACHING CODES DESIGN AND IMPLEMENTATION SPECIAL SECTION

Erasure-control coding for distributed networks

X.-H. Peng

Abstract: Channel coding has seen itself quickly emerge as a very promising technique in relatively new territory: distributed networks, where computing, communication and storage are extensively involved. In this environment, the distribution of rich media files to a large user population and the distribution of mission-critical data present special challenges for service providers and enterprises. Ensuring quality of service to these applications requires a combination of reliability, speed and scalability. Traditional packet-based networks endure poor end-to-end performance when packet-loss rates are high and a simple error-control strategy (through re-transmission) is applied. The paper will show how erasure-control coding, a special channel coding technique, plays an important part in promoting fault tolerance, and consequently quality of service across the network. Additionally, the construction of a class of maximum-distance-separable (MDS) array erasure codes is presented, which can be used in an adaptive way to efficiently control packet losses in a fluctuating network environment.

1 Introduction

Since Shannon's historical work [1], channel coding has played a remarkable role in modern telecommunications. Until very recently, most of the work in this paradigm has been devoted primarily to enhancing channel capacity over individual (point-to-point) noisy communication links, by applying appropriate error-control coding techniques such as error detection and error correction for random and/or burst errors. When computer and telecommunications technologies merge, problems in data transmission will be addressed in a multiple-device communication environment – namely the network, and channel coding will face challenges in transplanting its original role into this new environment.

In a point-to-point communication system that provides communication between, for example, a satellite and a ground station or between mobile terminals and the base station within a cellular cell, channel coding is extensively used to combat interference and error caused by signal attenuation and various types of noise. Powerful error-correcting codes, such as Reed–Solomon (RS) codes [2], convolutional codes [3], concatenated codes [4], turbo codes [5] and low-density-parity-check (LDPC) codes [6], are employed together with different error-control schemes, such as automatic-repeat-request (ARQ) and hybrid ARQ [7] in this system. This aims to achieve reliable communication at a maximum information rate over the specific link concerned.

When channel coding serves a network, it needs to integrate itself into the framework of the open systems interconnect (OSI) model [8]. This model represents a seven-layer protocol stack, as shown in Fig. 1*a*, with each layer responsible for certain tasks. Individual layers of the whole protocol stack contribute collaboratively to reliable and efficient communication across the network. Many practical networks, however, adopt a simplified version of this model, e.g. the five-layer model for the protocols defined in the Internet, as shown in Fig. 1*b*. Using the concept of layered protocols, coding schemes associated with different tasks are implemented separately at different layers including both lower (physical and data link) and higher (in particular the transport) layers [8], as indicated in Fig. 1*b*. The traditional error-control strategy for point-to-point communication can therefore be treated as a special case of the above, which is only responsible for lower layers, e.g. the link layer for wired systems [8] or both physical and link layers for wireless systems [9]. In most wired networks, error detection codes such as the cyclic redundancy check (CRC) codes are predominately used [8], as the random bit error rate is negligible in this type of networks.

In packet-based IP networks, including wired and wireless sections, the end-to-end performance depends on the protocols set at the transport layer, such as transmission control protocol (TCP) or user datagram protocol (UDP) [8]. TCP provides reliable end-to-end transmission by essentially re-transmitting the packets for which the source receives a negative acknowledgment or receives no acknowledgment within a transmission window. This protocol, equivalent to the ARQ strategy, could suffer long delays in the scenarios (quite common in a distributed network) such as poor channel conditions (particularly in wireless networks), multicast and long-distance transmission. UDP, in contrast to TCP, offers speedy data delivery as it has no re-transmission, but no guarantee for reliable services as it does not recover the lost or corrupted packets. Therefore, it is a big challenge for conventional IP-based networks to meet the increasing demand for supporting the multimedia distribution that requires both real-time and high-quality performances. This leads naturally to the consideration of employing forward-error-correction (FEC) channel coding techniques, combined with ARQ, to tackle these problems.

FEC schemes maintain constant throughput and have bounded time delay, which is well suited for real-time applications such as video transmission requiring a

IEE Proceedings online no. 20050291

doi:10.1049/ip-com:20050291

Paper first received 10th June 2005 and in final revised form 1st August 2005

The author is with Electronic Engineering, School of Engineering & Applied Science, Aston University, Birmingham B4 7ET, UK

E-mail: x-h.peng@aston.ac.uk

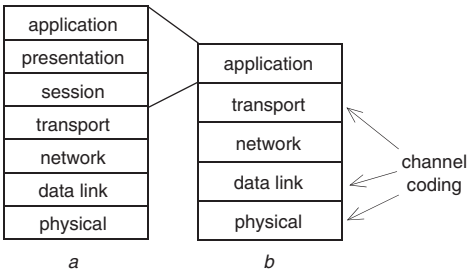

Fig. 1 *OSI model and five-layer model*
a OSI model
b Five-layer model

guaranteed maximum end-to-end delay. The reliability related problems in a modern distributed network are normally associated with packet losses. The packet-loss problem in wired networks is mainly caused by congestion that results in switches' buffers overflowing, so it is also called congestion erasure. Other contributors to this include 'destination unreachable' owing to various reasons and 'failure at error detection' that forces the destination node or intermediate routers to discard those packets. Although IPv4 and IPv6 have a powerful error-reporting scheme through using Internet control message protocol (ICMP) [10], the packet-loss problem will be left for the actions at the transport layer. As powerful error-detecting codes are exclusively used in most conventional IP-networks, the corrupted packets detected at the transport layer are normally discarded in both TCP and UDP modes, and all the packets accepted at this layer are treated as error free with a very high probability [11]. For this reason, the end-to-end transmission path between the transport layers at the source and the destination nodes in a network can be regarded as an erasure channel, and the FEC coding techniques designed for this type of channels are called erasure-control coding. Creating an efficient protocol for controlling erasures or packet losses in the network environment can provide robust fault tolerance and reduce the delay in data dissemination, and consequently increase system throughput to meet the quality of service (QoS) requirements.

The past decade has witnessed extensive research in this fast growing area and a wide range of work towards developing suitable erasure-control strategies and implementing them for various applications in the network domain. A variety of erasure codes with diversified features have been investigated for this purpose, including RS codes [11–13], array codes [14–16] and Fountain codes [17, 18]. They have also been applied with significant impact to multicast protocols [13, 19, 20], large-scale storage [21–23], backbone [24, 25], data broadcasting [26, 27] and distributed computing [22, 28, 29] in distributed networks.

For the main interest of this paper, the discussions will essentially focus on the coding strategies for ensuring reliable and efficient end-to-end data delivery, which mainly apply to the transport layer of a network protocol stack, because the end-to-end performance exhibits the overall quality of a network.

2 Erasure codes for distributed networks

Erasure codes are a class of FEC codes, i.e., no retransmission is required when they are employed. An erasure is a corrupted bit or symbol (packet) with an unknown value, but its location in the codeword is known to the decoder. An erasure code is designed to recover or correct the erasures, rather than to correct errors, from the encoded bits or packets correctly received. In a packet-based network, an (n, k) erasure code consists of k information packets and $n–k$ parity packets over a finite field $GF(q)$ (q is a power of a prime).

It is assumed that in an erasure channel, first introduced by Elias [30], packet-loss is an independent event with a fixed constant probability, p_l. In this channel model, packets are either correctly received or presented as erasures to the decoder. The Shannon capacity of an erasure channel is $(1-p_l)$, and transmission at any rate $R<(1(p_l)$ can be achieved with a random linear code [30]. The number of erasures that can be corrected by an erasure code, t_e, is bounded by

$$t_e \leq d - 1 \tag{1}$$

where d is the minimum distance of the code. Clearly, this doubles the number of errors that can be corrected by the same code when it is used solely for error correction.

Using a simple taxonomy, the construction of erasure codes can be classified into two categories: the maximum-distance-separable (MDS)-code approach and the sparse-graph-code approach. The construction methods and performance features of some useful erasure codes in both categories are summarised, as follows.

2.1 MDS erasure codes

Maximum distance separable is one of the desirable features of linear block codes, for achieving the maximum possible minimum distance for fixed n and k, i.e.

$$d = n - k + 1 \tag{2}$$

This result meets the Singleton bound [31] $d \leq n - k + 1$ with equality. Two trivial examples of MDS codes are the $(n, k, d) = (n, n-1, 2)$ single-parity-check code and $(n, 1, n)$ repetition code. The former has very limited error-control capacity though requiring a minimum redundancy, while the latter is equivalent to the retransmission of the same data, an error-control mechanism used in the ARQ schemes, thus inefficient in terms of bandwidth usage and throughput. Most non-trivial MDS codes are non-binary codes [32].

2.1.1 RS codes: RS codes are the special subclass of q-ary BCH codes, and form the most important class of MDS codes. They have been used widely for error control in both digital communication and storage systems, because of their powerful burst error correcting capacity. With this merit, RS codes can also be used as effective erasure codes, to correct multiple erasures of size q. In general, (n, k) RS codes can be constructed using the generator polynomial [32]

$$g(x) = (x - \alpha^b)(x - \alpha^{b+1}) \ldots (x - \alpha^{b+\delta-2}) \tag{3}$$

for some $\delta \geq 2$ and some $a \geq 1$, where $\alpha^i \in GF(q)$ for any i and $\alpha^n = 1$ but $\alpha^s \neq 1$ for any positive $s<n$. The codes generated have $n = q - 1$ symbols (or packets) in length, with each containing q bits.

As an MDS code, a RS code satisfies the condition $d = n - k + 1$ or, alternatively, $d - 1 = n - k$, which means that it can correct any combination of up to $n - k$ erasures, according to (1), when it is employed as the erasure code for reliable end-to-end communication in a distributed network. In this scenario, the transport layer of the source node generates n transport packets using the RS encoder for every k information packets passed on from the application layer. At the transport layer of the destination node, the RS decoder is able to recover the original

information as long as it can receive any k out of the n packets transmitted correctly. For this reason, the MDS erasure codes are also called the k-out-of-n codes.

RS codes can provide a large number of MDS codes for different application requirements. Extended RS codes [33, 34] by adding one or two overall parity check(s) are also maximum distance separable. These features have made RS codes an attractive candidate in this field. The decoding complexity of RS codes is at the scale of $O(n \log^2 n)$, achieved using the fast Fourier transform technique [35].

RS codes have been considered for different applications in the network environment [11–13, 19, 20]. For example, employing the RS code in association with the ARQ technique can largely enhance the performance of reliable multicast in the IP network [13], which is essential for ensuring QoS in multimedia (video and audio) distribution across the network. In this scheme, RS coding can either be placed in a sub-layer beneath ARQ, called 'layered RS' as shown in Fig. 2*a*, or works interactively with ARQ, called 'integrated RS' as shown in Fig. 2*b*.

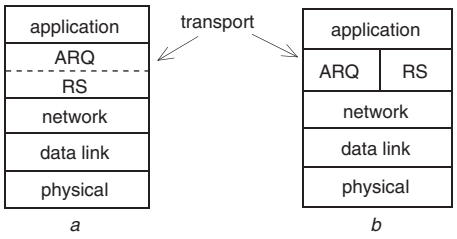

Fig. 2 *Layered RS and integrated RS*
a Layered RS
b Integrated RS

Layered RS is very similar to a hybrid ARQ strategy where two coding schemes, RS and ARQ, operate independently. The RS decoder at the destination recovers the original information packets once it has received any k out of n transmitted packets, and passes them to the ARQ sub-layer. If fewer than k packets are received, ARQ will then issue retransmission. In integrated RS, the packet-loss control can be done more efficiently as the packet-loss status and RS performance are monitored by ARQ in an attempt to avoid unnecessary retransmission. Given a multicast group of 10 000 and certain code parameter settings (e.g., $n = 9$ and $k = 7$), the average numbers of transmissions per correctly received packet can be reduced by 38% in layered RS and 46% in integrated RS, compared to the scheme without RS coding [13]. This means that the end-to-end throughput of the system can be increased substantially as a result of the introduction of RS erasure-control coding in conjunction with ARQ.

2.1.2 Other MDS codes: There exist other methods of constructing MDS erasure codes for correcting multiple erasures. Among them, the information dispersal algorithm (IDA) proposed by Rabin [29] presents a generic approach to establishing computationally efficient MDS coding schemes, for enabling fault-tolerant and efficient data transmission in distributed networks or between processors in parallel computers. In this approach, the encoded packets are transmitted through multiple paths between the source and the destination, so packet losses or even link failures can be tolerated. The MDS coding problem of this algorithm is to construct a specific set of n vectors, such that every subset of k different vectors are linearly independent. [29] suggests that such n vectors, $\alpha_i = (\alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,k})$ $1 \le i \le n$, can be formed by adopting either a method analogous to that used by RS codes, or the construction

$$g_i = \left(\frac{1}{x_i + y_1}, \ldots, \frac{1}{x_i + y_k}\right)^T \tag{4}$$

where $x_1, \ldots, x_n, y_1, \ldots, y_k \in GF(q)$ satisfy the conditions: for all i and j, $x_i + y_j \neq 0$; for $i \neq j$, $x_i \neq x_j$ and $y_i \neq y_j$. Here it is required that $n + k < q$. The implementation of IDA through this construction requires just $O(k^2)$ operations.

MDS array codes are also extensively investigated [36–39], for the applications mainly in storage but applicable to communications networks as well. The array code concerned is presented in a $(p-1) \times n$ array, with $p \ge 3$ being a prime. Each column of the array is treated as a symbol or a packet of $(p-1)$ bits, and the code contains p information packets and u parity packets ($p + u = n$), resulting in an (n, p) or $C(p, u)$ linear block code over $GF(2^{p-1})$. The parity check matrix in a systematic form of $C(p, u)$ is given by [39]

$$H(p,u) = \begin{pmatrix} 1 & 1 & \cdots & 1 & 1 & 0 & \cdots & 0 \\ 1 & \alpha & \cdots & \alpha^{p-1} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{u-1} & \cdots & \alpha^{(u-1)(p-1)} & 0 & 0 & \cdots & 1 \end{pmatrix} \tag{5}$$

where α^i is the element in the ring of binary polynomials modulo $M_p(x) = (x^p - 1)/(x - 1) = x^{p-1} + \cdots + x + 1$ [39], and α^i and $\alpha^i + \alpha^j$ are invertible modulo $M_p(x)$. In this case, the main interest of the investigation is to find how many parity packets can be added on for the fixed number of information packets, p. It has been shown, by exploiting the property of the Vandermonde matrix [40] within (5), that $C(p, u)$ have the guaranteed MDS property for $u = 2$ and 3, but not always so for $u > 3$ depending on the total number of packets considered.

Other MDS array codes can be generated using different methods, but they share almost the same properties with $C(p, u)$, e.g. their encoding and decoding involve cyclic shifts and exclusive-or (XOR) operations only, thus computationally efficient, but their error-control capacity and number of codes available are limited, compared to RS codes and the IDA. Examples of these codes and those that can be developed straightforward into this category include complex rotary codes [41], X-code [37], B-code [38], cyclic-square codes [42] and augmented array codes [43, 44].

2.2 Sparse-graph erasure codes

The beauty of the LDPC code with its random generating style and sparsity [6] has inspired another important class of codes, aiming to meet the demands of both performance and complexity in data communications. The parity check matrix of the LDPC code is said to be sparse, as it has a low density of ones, and so is its associated graph used for encoding and decoding in terms of the small number of edges linked to each node in the graph. Based on this framework, a class of erasure codes, called Fountain codes [18], have been developed for addressing the needs especially for erasure channels in distributed networks.

The initial Fountain codes, called Tornado codes [45, 46], are very similar to Gallager's LDPC code and integrate some features from other codes [47, 48]. For given k information packets the source randomly generates,

according to the distribution function over the set of integers $\{1, \ldots, k\}$, a fixed number (n) of encoded packets for transmission, as shown in Fig. 3. At the destination the original k information packets can be recovered from a random set of $(1+\varepsilon)k(\varepsilon>0)$ packets received, with probability $1-O(k^{-3/4})$. Its encoding and decoding algorithms can be described with an irregular graph, rather than the regular one used in Gallager's approach, and the running times for encoding and decoding processes are both proportional to $n\ln(1/\varepsilon)$ since the operation is predominantly bit-wise XOR.

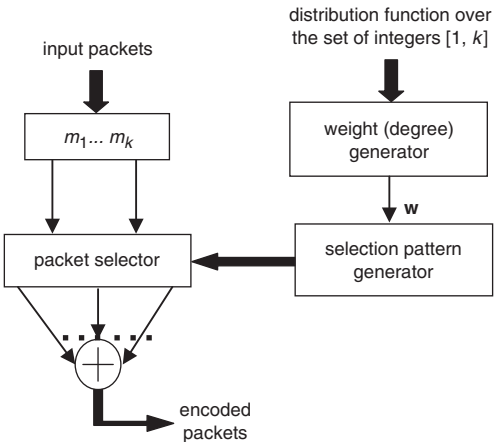

Fig. 3 *Fountain encoder*

Later, the key fountain idea was developed [17] and practically realised, leading to a class of so-called universal erasure codes or LT codes [18]. The fundamental innovation of these codes is the encoder that can generate a potentially limitless number of encoded packets on the fly and send them on demand to the destination until it has received a sufficient number of packets for data reconstruction. These features are particularly enviable for heavily impaired channels where packet-loss rate is very high and more packets than the normal number are required by the decoder at the destination. With LT codes, the k original information packets can be recovered at the destination from any $k+O(\sqrt{k}\ln^2(k/\delta))$ packets received with probability $1-\delta$. The encoding and decoding processes need $O(\ln(k/\delta))$ and $O(k\ln(k/\delta))$ operations, respectively. An extension of the conventional Fountain codes (or LT codes), called Raptor codes [49], relax the condition that all the original information packets need to be recovered. By concatenating a traditional erasure code with an LT code, a Raptor code can still recover all the information packets but with a constant cost for both encoding and decoding.

3 Construction of a class of MDS array erasure codes

MDS codes are optimal in the sense that they have the maximum possible minimum distance for the given code length and dimension. Array codes suit naturally the packet-based system in terms of data presentation and processing. However, the MDS array codes currently available are mostly designed for storage and have limited erasure-control capacity. In this Section, a class of MDS array erasure codes are introduced for packet-loss control in a distributed network. The erasure-correcting capacity of these codes varies as they can have different numbers of parity packets for a given information array. This feature allows the coding scheme to be adaptive to the changes of the environment and to achieve the required network performance such as throughput at a minimum resource (e.g. bandwidth) cost.

3.1 Encoding algorithm

Given an $(m\times k)$ information array, represented by a matrix B, encoding is carried out by directly multiplying B by a generator matrix G, resulting in an $(m\times n)$ code array or codeword V of code C. This process can be expressed by

$$B\times G=\begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,k} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,k} \\ \vdots & \vdots & & \vdots \\ b_{m,1} & b_{m,2} & \cdots & b_{m.k} \end{bmatrix}\times\begin{bmatrix} g_{11} & g_{1,2} & \cdots & g_{1,n} \\ g_{2,1} & g_{2,2} & \cdots & g_{2,n} \\ \vdots & \vdots & & \vdots \\ g_{k,1} & g_{k,2} & \cdots & g_{k,n} \end{bmatrix}$$
$$=\begin{bmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,n} \\ v_{2,1} & v_{2,2} & \cdots & v_{2,n} \\ \vdots & \vdots & & \vdots \\ v_{m,1} & v_{m,2} & \cdots & v_{m,n} \end{bmatrix}=V$$

If we take the columns of the code array as the elements of the code, i.e., $\boldsymbol{V}=[\boldsymbol{v_1}\,\boldsymbol{v_2}\,\ldots\,\boldsymbol{v_n}]$ and $\boldsymbol{v_i}=[\boldsymbol{v_{1,i}}\,\boldsymbol{v_{2,i}}\ldots\boldsymbol{v_{m,i}}]^{\boldsymbol{T}}$, code C can be regarded as an (n, k)-column code over Z_q^m, where k is the number of the columns of the information array and $q=p^r$ (p is a prime and r is a positive integer). The generation of each column of C, C_i $(1\le i\le n)$, involves all the information characters $b_{i,j}\in Z_q$ $(1\le i\le m, 1\le j\le k)$ in the array. Code C can also be viewed as an interleaving code if the rows of the information array are taken as individual information vectors. When this code is used in a network for packet-loss control, the columns of the code array, $\boldsymbol{v_i}$, may be treated as the packets generated by the source.

To be an MDS erasure code, C must satisfy the condition that it can recover the original information array at the destination with any k out of n packets correctly received. In other words, code C can correct up to $n-k$ erasures. The structure of the generator matrix G designed for meeting this condition will be revealed through the description of the decoding algorithm.

3.2 Decoding algorithm

Given any k out of n packets $\boldsymbol{v}_{l_1}, \boldsymbol{v}_{l_2}, \ldots, \boldsymbol{v}_{l_k}$ correctly received at the destination, where $l_i\in\{1,2,\ldots,n\}$ and all l_i $(1\le i\le n)$ are distinct, the relationship between the k received packets and the original information array is given by

$$[\boldsymbol{v}_{l_1}, \boldsymbol{v}_{l_2}, \ldots, \boldsymbol{v}_{l_k}]=B\times G',$$

where

$$G'=\begin{bmatrix} g_{1,l_1} & g_{1,l_2} & \cdots & g_{1,l_k} \\ g_{2,l_1} & g_{2,l_2} & \cdots & g_{2,l_k} \\ \vdots & \vdots & & \vdots \\ g_{k,l_1} & g_{k,l_2} & \cdots & g_{k,l_k} \end{bmatrix}$$

Thus the information array can be recovered by

$$B=[\boldsymbol{v}_{l_1}, \boldsymbol{v}_{l_2}, \ldots, \boldsymbol{v}_{l_k}]\times(G')^{-1}$$

For G' to be invertible, it is required that

$$\det(G')\neq 0 \qquad (6)$$

Condition (6) implies that G' is non-singular, or that any k columns in G are linearly independent. To meet this condition, the elements of G, $g_{i,j}$, are selected to be the elements of a cyclic group $U(Z_q)=\{\beta_1,\beta_2,\ldots,\beta_s\}$, which is a set of invertible elements of Zq and s is the size or order

of $U(Z_q)$, and the n columns of G are constructed as

$$\boldsymbol{g_i} = (1, \beta_i, \ldots, \beta_i^{k-1})^T \quad 1 \le i \le n \tag{7}$$

As a result, any k columns of the G composed form a Vandemonde-like matrix. For a fixed q, the code length n can be chosen between $k < n < q$, i.e., the code parameters will be in the form of $(k+u, k)$, where $1 \le u \le q-k-1$. This can enable an adaptive coding scheme that allows the efficient utilisation of network resources. Note that $t_e = u$ for MDS codes according to (1) and (2). The code can have variable erasure-tolerant capacity in term of u for fixed q and k, depending on network conditions and the requirement on packet-loss control. The adaptation process can operate easily when the packet-loss status is properly monitored by the destination and timely feedback to the source. If the network conditions are varying in a great scale such as in wireless networks, a large q can be considered for the code to offer a wide range of erasure tolerance. The implementation of the encoding and decoding algorithms of the code is demonstrated in the following example.

Example 1: Consider an information array of $2 \times 3 = 6$ characters or blocks: $\begin{pmatrix} b_{1,1}, b_{1,2}, b_{1,3} \\ b_{2,1}, b_{2,2}, b_{2,3} \end{pmatrix}$, so $m = 2$, $k = 3$. By choosing $q = p^r = 7^1 = 7$, the code length is in the range $4 \le n \le 6$, implying that the code can tolerate u $(1 \le u \le 3)$ erasures. If we choose $n = 6$, the cyclic group concerned is $U(Z_7) = \{1, 2, \ldots, 6\}$, and codedwords are generated from the source by

$$\begin{aligned} B \times G &= \begin{bmatrix} b_{1,1} & b_{1,2} & b_{1,3} \\ b_{2,1} & b_{2,2} & b_{2,3} \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ \beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6 \\ \beta_1^2 & \beta_2^2 & \beta_3^2 & \beta_4^2 & \beta_5^2 & \beta_6^2 \end{bmatrix} \\ &= \begin{bmatrix} v_{1,1} & v_{1,2} & v_{1,3} & v_{1,4} & v_{1,5} & v_{1,6} \\ v_{2,1} & v_{2,2} & v_{2,3} & v_{2,4} & v_{2,5} & v_{2,6} \end{bmatrix} \\ &= [\boldsymbol{v_1} \quad \boldsymbol{v_2} \quad \boldsymbol{v_3} \quad \boldsymbol{v_4} \quad \boldsymbol{v_5} \quad \boldsymbol{v_6}] \end{aligned}$$

where $\beta_i = i$ for $1 \le i \le 6$ and

$$v_{1,i} = b_{1,1} + \beta_i b_{1,2} + \beta_i^2 b_{1,3}$$

$$v_{2,i} = b_{2,1} + \beta_i b_{2,2} + \beta_i^2 b_{2,3} \quad \text{for } 1 \le i \le 6$$

All the calculations above and thereafter are made over Z_7 or modulo 7. In this example, the destination is able to recover the original information array as long as it has correctly received any three out of six packets. Suppose that the first three coded packets are correctly received. The information array can then be recovered through

$$\begin{aligned} B &= [\boldsymbol{v_1}\, \boldsymbol{v_2}\, \boldsymbol{v_3}] \times (G')^{-1} \\ &= \begin{bmatrix} b_{1,1} + b_{1,2} + b_{1,3} & b_{1,1} + 2b_{1,2} + 4b_{1,3} \\ b_{2,1} + b_{2,2} + b_{2,3} & b_{2,1} + 2b_{2,2} + 4b_{2,3} \\ b_{1,1} + 3b_{1,2} + 2b_{1,3} & \\ b_{2,1} + 3b_{2,2} + 2b_{2,3} & \end{bmatrix} \times \begin{bmatrix} 3 & 1 & 4 \\ 4 & 4 & 6 \\ 1 & 2 & 4 \end{bmatrix} \\ &= \begin{bmatrix} b_{1,1} & b_{1,2} & b_{1,3} \\ b_{2,1} & b_{2,2} & b_{2,3} \end{bmatrix} \end{aligned}$$

The computational cost on determining the inverse of G' can be reduced to $O(k^2)$ [50]. If $q = 2^r$, the implementation can be accomplished mainly by XOR operations [36].

4 Discussion

The codes discussed above all have the potential to be used for tackling the problems in relation to packet losses, delay and throughput of erasure channels in packet-based networks. However, it is desirable that the codes used are capable of correcting or tolerating as many erasures as possible and have fast encoding and decoding algorithms. Other criteria for choosing suitable codes may include: scalability, adaptability, and costs of network resources (bandwidth, storage, etc.).

MDS codes such as RS codes and the IDA are capable of correcting a large number of erasures and are highly efficient in terms of the utilisation of parity checks. However, they both have quadratic computational complexity, which could lead to a substantial increase in processing cost when the size of the field becomes large. MDS array codes have low implementation costs owing to special code structure. They are flexible in nature and can adapt well to channel conditions at minimum cost of system resources. The codes presented in Section 3, for example, are able to correct multiple erasures and offer two levels (based on q and u) of adaptation strategies, according to environment conditions and packet-loss control requirements.

Fountain codes and their extensions appear to be a versatile class of erasure codes whose abilities in erasure resilience and saving network resources can scale well in a highly distributed network. Fountain codes are not generally MDS codes, as indicated by their code parameters. Also, as the codewords are generated randomly, it is essential that the information regarding the encoding rule (or the distribution function used by the encoder) be reliably delivered to the destination for ensuring the correct decoding algorithm in place. Further investigations may include evaluating the realistic performance of the encoder for different distribution functions applied, feasible and efficient communication for making the encoding rule known to the destination, and environment awareness for effectively operating adaptive distribution of the encoded packets in both unicast and multicast scenarios.

FEC channel coding, in particular using erasure codes, can play a crucial role in distributed networks for reducing the packet-loss rate and promoting speedy data delivery. This paper addresses the features of erasure codes that can be further exploited towards achieving optimal network performances. The potentials of erasure-control coding or channel coding in general have been recognised particularly for wireless networks, as the conventional TCP protocol performs poorly in a wireless environment [51–53]. This is because the packet-loss situation is much worse in this environment than in a wired network, owing to high bit error rate, unstable channel characteristics and user mobility. Using simple packet-loss detection plus retransmission to ensure the required quality for wireless networks would cause significant delay and reduction in throughput. This problem can be effectively dealt with by adopting well-integrated FEC schemes including forward error-control and erasure-control coding, in combination with ARQ and other network protocols. At the same time, other related issues in the context of the distributed network should also be investigated jointly with the development of the coding scheme itself. There are some topics in this area that could lead to, in the author's view, the development of suitable technologies for future-generations of wired/wireless communication systems. A selection of these topics are listed below.

- A cross-layer consideration on coding issues, for example, communication between transport and link layers.
- Find MDS codes with large length n for given dimension k and field size q.
- Application-driven coding scheme design to meet specific QoS requirements.
- Soft-decision decoding for mixed error/erasure channels.
- Erasure-control coding at lower layers for connection-oriented networks.
- Erasure-control coding in multi-hop wireless networks, e.g., *ad hoc* or sensor networks.

5 Acknowledgments

The author wishes to thank Professor P.G. Farrell and the anonymous reviewers for their valuable suggestions and comments for improving the paper.

6 References

1 Shannon, C.E.: 'A mathematical theory of communication', *Bell Syst. Tech. J.*, 1948, **27**, pp. 379–423 (part 1), pp. 623–656 (Part 2)
2 Reed, I.S., and Solomon, G.: 'Polynomial codes over certain finite fields', *J. Soc. Ind. Appl. Math.*, 1960, **8**, pp. 300–304
3 Elias, P.: 'Coding for noisy channels', *IRE Conv. Rec.*, 1955, **3**, pp. 37–47
4 Forney, G.D. Jr.: 'Concatenated codes' (MIT Press, Cambridge, MA, 1966)
5 Berrou, C., Glavieux, A., and Thitimajshima, P.: 'Near Shannon limit error-correcting coding and decoding: turbo codes'. Proc. IEEE Int. Conf. Communications, Geneva, Switzerland, May 1993, pp. 1064–1070
6 Gallager, R.G.: 'Low density parity check codes', *IRE Trans. Inf. Theory*, 1962, **IT-8**, pp. 21–28
7 Lin, S., and Costello, D.J. Jr.: 'Error control coding' (Pearson Prentice Hall, 2004, 2nd edn.)
8 Tanenbaum, A.S.: 'Computer networks' (Prentice Hall PTR, 2003, 4th edn.)
9 Holma, H., and Toskala, A.: 'WCDMA for UMTS: radio access for third generation mobile communications' (Wiley, 2001)
10 Conta, A., and Deering, S.: 'Internet control message protocol (ICMPv6) for the Internet protocol version 5 (IPv6) specification'. RFC 2463, Dec. 1998
11 McAuley, A.J.: 'Reliable broadband communications using a burst erasure correcting code'. Proc. ACM SIGCOMM'90, Philadelphia, PA, Sept. 1990, pp. 287–306
12 Rizzo, L.: 'Effective erasure codes for reliable computer communication protocols', *Comput. Commun. Rev.*, 1997, **27**, pp. 24–36
13 Nonnenmacher, J., Biersack, E.W., and Towsley, D.: 'Parity-based loss recovery for reliable multicast transmission', *IEEE Trans. Commun.*, 1998, **6**, pp. 349–361
14 Farrell, P.G.: 'A survey of array error control codes', *Eur. Trans. Telecom.*, 1992, **3**, pp. 441–454
15 Blaum, M., Farrell, P.G., and VanTilborg, H.C.A.: 'Array codes' in Pless, V.S. and Huffman, W.C. (Eds.), 'Handbook of coding theory' (North-Holland, 1998)
16 Blaum, M., Bruck, J., and Vardy, A.: 'MDS array codes with independent parity symbols', *IEEE Trans. Inf. Theory*, 1996, **42**, pp. 529–542
17 Byers, J., Luby, M., Mitzenmacher, M., and Rege, A.: 'A digital fountain approach to reliable distribution of bulk data'. Proc. ACM SIGCOMM'98, Vancouver, Canada, 1998, pp. 55–67
18 Luby, M.: 'LT codes'. Proc. IEEE Symp. on Foundations of Computer Science, 2002, pp. 271–280
19 Sakakibara, K., and Kasahara, M.: 'A multicast hybrid ARQ scheme using MDS codes and GMD decoding', *IEEE Trans. Commun.*, 1995, **43**, pp. 2933–2939
20 Rubenstein, D., Kurose, J., and Towsley, D.: 'Real-time reliable multicast using proactive forward error correction'. Proc. NOSSDAV, Cambridge, UK, July 1998, pp. 279–194
21 Byers, J.W., Luby, M., and Mitzenmacher, M.: 'Accessing multiple mirror sites in parallel: Using Tornado codes to speed up downloads'. Proc. IEEE INFOCOM, pp. 275–283
22 Bohossian, V., Fan, C.C., LeManheu, P.S., Riedel, M.D., Xu, L., and Bruck, J.: 'Computing in the RAIN: A reliable array of independent nodes', *IEEE Trans. Parallel Distrib. Syst.*, 2001, **12**, pp. 99–113
23 Cooly, J.A., Mineweaser, J.L., Servi, L.D., and Tsung, E.T.: 'Soft-based erasure codes for scalable distributed storage'. Proc. 20th IEEE/11th NASA Goddard Conf. Mass Storage Syst. and Tech, San Diego, CA, USA, Aril 2003, pp. 157–164
24 Ohta, H., and Kitemi, T.: 'Cell loss recovery method using FEC in ATM networks', *IEEE J. Sel. Areas Commun.*, 1991, **9**, pp. 1471–1483
25 Kousa, M.A., Elhakeem, A.K., and Yang, H.: 'Performance of ATM networks under hybrid ARQ/FEC error control scheme', *IEEE/ACM Trans. Netw.*, 1999, **7**, pp. 917–925
26 Bestavros, A.: 'AIDA-based real-time fault-tolerant broadcast disks'. Proc. 16th IEEE Real-Time Tech. App. Symp., 1996, pp. 49–58
27 Peng, X.-H.: 'Fault-tolerant scheduling for asymmetric communications'. Proc. IEE 5th European. Personal Mobile Communications. Conf, Glasgow, Scotland, April 2003, pp. 575–579
28 LeMahieu, P.S., Bohossian, V.Z., and Bruck, J.: 'Fault-tolerant switched local area networks'. Proc. Int. Parallel Processing Symp., 1998, pp. 747–751
29 Rabin, M.O.: 'Efficient dispersal of information for security, load balancing and fault tolerance', *J. ACM*, 1989, **36**, pp. 335–348
30 Elias, P.: 'Coding for two noisy channels'. Proc. 3rd London Symp. Information Theory, 1955, pp. 61–76
31 Joshi, D.D.: 'A note on upper bounds for minimum distance bounds', *Inf. Control*, 1958, **1**, pp. 289–295
32 MacWilliams F., J., and Sloane, N.J.A.: 'The theory of error-correcting codes' (North-Holland, Amsterdam, 1998)
33 Tanaka, H., and Nishida, F.: 'A construction of a polynomial code', *Electron. Commun. Jpn.*, 1970, **53-A**, pp. 24–31
34 Gross, A.J.: 'Some augmentations of Bose-Chaudhuri error correcting codes' in Srivastava, J.N., (Ed.): 'A survey of combinatorial theory' (North-Holland, Amsterdam, 1973)
35 Justesen, J.: 'On the complexity of decoding Reed-Solomon codes', *IEEE Trans. Inf. Theory*, 1976, **22**, pp. 237–238
36 Blaum, M., Brady, J., Bruck, J., and Menon, J.: 'Evenodd: an efficient scheme for tolerating double disk failures in RAID architectures', *IEEE Trans. Comput.*, 1996, **44**, pp. 192–202
37 Xu, L., and Bruck, J.: 'X-code: MDS array codes with optimal encoding', *IEEE Trans. Inf. Theory*, 1999, **45**, pp. 272–276
38 Xu, L., Bohossian, V., Bruck, J., and Wagner, D.: 'Low-density MDS codes and factors of complete graphs', *IEEE Trans. Inf. Theory*, 1999, **45**, pp. 1817–1826
39 Blaum, M., and Roth, R.: 'New array codes for multiple phased burst correction', *IEEE Trans. Inf. Theory*, 1993, **39**, pp. 66–77
40 Hoffman, K., and Kunze, R.: 'Linear algebra' (Prentice-Hall, Englewood Cliffs, 1971, 2nd edn.)
41 Fan, J.: 'An investigation on new complex-rotary codes'. Proc. Int. Symp. Information. Theory, Brighton, England, 1985, pp. 1–8
42 Peng, X.-H., and Farrell, P.G.: 'Cyclic square and its coding characteristics', *Electron. Lett.*, 1991, **27**, pp. 1706–1708
43 Peng, X.-H., and Farrell, P.G.: 'Optimal augmentation of product codes', *Electron. Lett.*, 2004, **40**, pp. 750–752
44 Honary, B., Markarian, G.S., and Farrell, P.G.: 'Generalised array codes and their trellis structure', *Electron. Lett.*, 1993, **29**, pp. 541–543
45 Luby, M., Mitzenmacher, M., Shokrollahi, A., Speilman, D., and Stemann, V.: 'Practical loss-resilient codes'. Proc. 9th Ann. ACM Symp. Theory of Computing, 1997
46 Luby, M., Mitzenmacher, M., Shokrollahi, A., and Spielman, D.: 'Efficient erasure correcting codes', *IEEE Trans. Inf. Theory*, 2001, **47**, pp. 569–584
47 MacKay, D.J.C., and Neal, R.M.: 'Good codes based on very sparse matrices'. Cryptography and Coding, 5th IMA Conf., Vol. 1025, 1995, pp. 110–111
48 Spielman, D.: 'Linear-time encodable and decodable error-correcting codes', *IEEE Trans. Inf. Theory*, 1996, **42**, pp. 1723–1731
49 Shokrollahi, A.: 'Raptor codes'. Digital Fountain Technical Report, June 2003
50 Press, W.H., Flannery, B.P., Tenkolsky, S.A., and Vetterling, W.T.: 'Numerical recipes in fortran: The art of scientific computing' (Cambridge University Press, 1992, 2nd edn.)
51 Kim, S., Fonseca, R., and Culler, D.: 'Reliable transfer on wireless sensor networks'. Proc. 1st Int. Conf. on Sensor and Ad Hoc Communications and Networks, Oct. 2004, pp. 449–459
52 Tian, Y., Xu, K., and Ansari, N.: 'TCP in wireless environments: Problems and solutions', *IEEE Radio Commun.*, 2005, pp. s27–s32
53 Fu, Z., Luo, H., Zerfos, P., Zhang, L., and Gerla, M.: 'The impact of multihop wireless channel on TCP performance', *IEEE Trans. Mob. Comput.*, 2005, **4**, pp. 209–221