

Performance versus Overhead for Fountain Codes over \mathbb{F}_q

Gianluigi Liva, *Member, IEEE*, Enrico Paolini, *Member, IEEE*, and Marco Chiani, *Senior Member, IEEE*

Abstract—Fountain codes for packet erasure recovery are investigated over Galois fields of order $q \geq 2$. It is shown through development of tight upper and lower bounds on the decoding failure probability under maximum likelihood decoding, that the adoption of higher order Galois fields is beneficial, in terms of performance, for linear random fountain codes. Moreover, it is illustrated how Raptor codes can provide performances very close to those of random fountain codes, with an affordable encoding and decoding complexity. Non-binary Raptor codes turn out to represent an appealing option for applications requiring severe constraints in terms of performance versus overhead, especially for small source block sizes.

Index Terms—Fountain codes, Raptor codes, maximum likelihood decoding.

I. INTRODUCTION

FOUNTAIN codes have been introduced in [1] as a possible solution for information delivery in broadcast and multicast networks. A fountain encoder is capable to produce an undefined amount of encoded symbols (or output symbols) out of a source block formed by k source symbols (or input symbols). In broadcast and multicast networks, each user collects symbols generated by the fountain encoder. Once a sufficiently large amount of symbols has been received, the user is able to recover the k input symbols. For an ideal fountain code this amount coincides with k : the decoder is able to recover the source block from any set of k output symbols. For real fountain codes, the source block is recovered with a probability that is non-decreasing with the number of symbols received in surplus with respect to (w.r.t.) k . This integer number is referred to as the *overhead*, here denoted by δ .

Fountain codes are usually adopted in communication networks to recover lost packets. Here, an object (e.g., a file) is divided into k source packets, all of the same length L [bits], out of which the encoder produces an undefined amount of encoded packets, each of length L [bits]. If a binary fountain code is used, each encoded packet may be obtained as a bit-wise exclusive-or of a subset of the source packets. Similarly, for a fountain code over a Galois field \mathbb{F}_q of characteristic two with $q > 2$, each source packet is regarded as a collection of $L/\log_2 q$ symbols in \mathbb{F}_q : each encoded packet is obtained as a symbol-wise sum (in \mathbb{F}_q) of a subset of the source packets. Hence, for a given object the encoding latency can be kept

constant, regardless the Galois field order used for performing the linear combinations.

In this letter, two classes of fountain codes are considered, namely, linear random fountain (LRF) codes and Raptor codes [2]. For both, maximum-likelihood (ML) decoding is adopted. The decoding error probability of LRF codes over Galois fields of order $q \geq 2$, as a function of the overhead, is investigated in Section II. It is shown through tight upper and lower bounds that, by adopting a code construction on non-binary fields, the probability of decoding success can be largely increased for the same overhead. In Section III, it is illustrated through simulation how Raptor codes constructed on Galois fields of order $q \geq 2$ are capable to closely approach the performance of LRF codes even for small overheads. Final remarks follow in Section IV.

II. LINEAR RANDOM FOUNTAIN CODES OVER \mathbb{F}_q

Let $\mathbf{c} = [c_i]_{i=0,\dots,k-1} \in \mathbb{F}_q^k$ be a vector of k input symbols.¹ A LRF code over \mathbb{F}_q is a random linear map $\mathbb{F}_q^k \rightarrow \mathbb{F}_q^{\mathbb{N}}$, where $\mathbb{F}_q^{\mathbb{N}}$ denotes the set of all sequences over \mathbb{F}_q . The encoder generates the output symbol e_j , $j \in \mathbb{N}$, as follows:

- for each input symbol c_i , a coefficient $g_{ji} \in \mathbb{F}_q$ is picked independently with uniform probability;
- the output symbol e_j is computed as $e_j = \sum_{i=0}^{k-1} g_{ji}c_i$, where all operations are performed in \mathbb{F}_q .

Assume the fountain encoder generates a stream of n output symbols. Denoting these symbols by $\mathbf{e}_{(0,\dots,n-1)}$, we have $\mathbf{e}_{(0,\dots,n-1)} = \mathbf{G}_{(0,\dots,n-1)} \mathbf{c}$ where

$$\mathbf{G}_{(0,\dots,n-1)} = \begin{bmatrix} g_{00} & \cdots & g_{0,k-1} \\ \vdots & & \\ g_{n-1,0} & \cdots & g_{n-1,k-1} \end{bmatrix}.$$

Note that, in general, $\mathbf{G}_{(0,\dots,n-1)}$ is a dense matrix.

The index $j \in \mathbb{N}$ assigned to the output symbol e_j is also known as the encoded symbol identifier (ESI). For an ESI j , we let $\Theta_j = \{g_{ji} : i = 0, \dots, k-1\}$.

Assume $k+\delta \geq k$ output symbols $\mathbf{e}_{(j_1,\dots,j_{k+\delta})}$ are collected at the receiver (the other transmitted symbols being erased by the channel) and let $J = \{j_1, \dots, j_{k+\delta}\}$ be the set of ESIs of these symbols. We have

$$\mathbf{G}_{(j_1,\dots,j_{k+\delta})} \mathbf{c} = \mathbf{e}_{(j_1,\dots,j_{k+\delta})} \quad (1)$$

where $\mathbf{G}_{(j_1,\dots,j_{k+\delta})}$ is the $((k+\delta) \times k)$ matrix composed of the $k+\delta$ rows of $\mathbf{G}_{(0,\dots,n-1)}$ whose indexes belong to J . ML decoding consists of solving (1) through Gaussian elimination to recover all k input symbols \mathbf{c} . Note that, to this purpose, for each collected output symbol e_j , the decoder needs the

Manuscript received October 22, 2009. The associate editor coordinating the review of this letter and approving it for publication was V. Stankovic.

G. Liva is with the Institute of Communication and Navigation of the Deutsches Zentrum für Luft- und Raumfahrt (DLR), 82234 Wessling, Germany (e-mail: Gianluigi.Liva@dlr.de).

E. Paolini and M. Chiani are with DEIS/WiLAB, University of Bologna, 47521 Cesena (FC), Italy (e-mail: {e.paolini, marco.chiani}@unibo.it).

Supported in part by the EC under Seventh Framework Program grant agreement ICT OPTIMIX n.INFSO-ICT-214625 and in part by the EC-IST SatNEX-II Project (IST-27393).

Digital Object Identifier 10.1109/LCOMM.2010.02.092080

¹Throughout the letter vectors will be intended as column vectors.

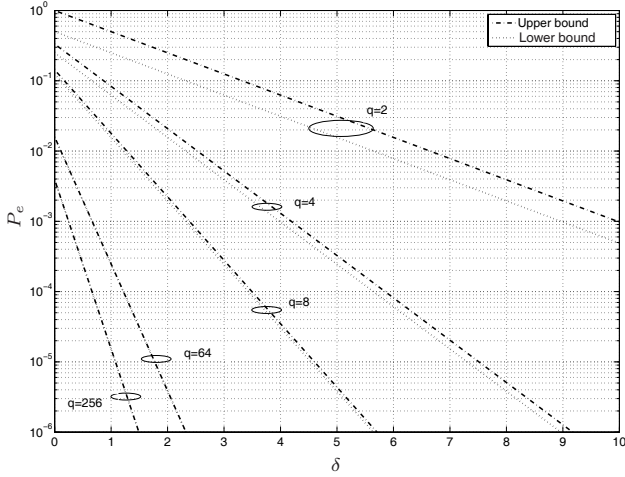


Fig. 1. Lower and upper bounds on the decoding error probability of LRF codes over \mathbb{F}_q , for $q = 2, 4, 8, 64, 256$. The bounds are independent of k .

corresponding Θ_j .² Decoding is successful if and only if $\text{rank}(\mathbf{G}_{(j_1, \dots, j_{k+\delta})}) = k$. The decoding error probability is then given by (see, e.g., [3])

$$P_e(k, \delta, q) = 1 - \prod_{i=1}^k \left(1 - \frac{q^{i-1}}{q^{k+\delta}}\right) \quad (2)$$

$$= 1 - (q^{-k-\delta}; q)_k \quad (3)$$

where the formulation (3) uses the q -Pochhammer symbol.

Proposition 1: The decoding failure probability of a LRF code over \mathbb{F}_q , under ML decoding, fulfills

$$q^{-\delta-1} \leq P_e(k, \delta, q) < \frac{1}{q-1} q^{-\delta} \quad (4)$$

with equality for the lower bound if and only if $k = 1$.³

Proof: The lower bound is obtained by observing that $1 - P_e(k, \delta, q) = \prod_{i=1}^k (1 - q^{i-1-k-\delta}) \leq (1 - q^{k-1-k-\delta}) = 1 - q^{-1-\delta}$, where the inequality is due to each factor being less than 1. Note that equality holds if and only if $k = 1$.

The upper bound is proved by induction on k . The bound holds for $k = 1$. In fact, $1 - P_e(1, \delta, q) = 1 - q^{-1-\delta} = 1 - \frac{1}{q} q^{-\delta} > 1 - \frac{1}{q-1} q^{-\delta}$. Assuming the bound is true for k , then it is true also for $k+1$. In fact, $1 - P_e(k+1, \delta, q) = \prod_{i=1}^{k+1} (1 - q^{i-1-k-1-\delta}) = [1 - P_e(k, \delta+1, q)](1 - q^{-1-\delta}) > (1 - \frac{1}{q-1} q^{-1-\delta})(1 - q^{-1-\delta}) > 1 - \frac{1}{q-1} q^{-\delta}$ where the first inequality is due to the bound for k , and the second inequality can be easily verified. ■

Remarkably, the upper bound and the lower bound in (4) are independent of the number k of input symbols, which allows to develop considerations valid for all k . The bounds are depicted in Fig. 1 as functions of δ for $q = 2, 4, 8, 64$ and 256 . The two bounds converge for large q and the gap between them is very small for all q . It can be verified that the upper bound is extremely tight even for $q = 2$ and k in the order of a few tens. Fig. 1 reveals an inherent advantage, in terms of

²In real systems, Θ_j is not usually transmitted as it is obtained by the decoder through the same pseudo-random generator used for encoding, starting from ESIs. Therefore, it is sufficient to transmit the ESI together with the corresponding output symbol.

³The upper bound for the binary case, $P_e(k, \delta, 2) < 2^{-\delta}$, appeared in [4].

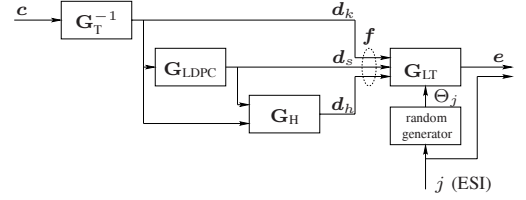


Fig. 2. Block diagram of the systematic Raptor encoder specified in [6].

performance for the same overhead, of constructing the code on higher order Galois fields for a given k . For example, with only one symbol of overhead, we have $P_e \simeq 2.5 \cdot 10^{-4}$ for all k over \mathbb{F}_{64} , while we have $P_e \geq 2.5 \cdot 10^{-1}$ for all k over \mathbb{F}_2 .

The independence of the two bounds from k and the small gap between them emphasize a weak dependence of the performance on k , for a given overhead and Galois field order. Note that using a large block size k increases the fountain code efficiency defined as $\eta = k/(k + \delta)$. However, LRF codes are not practical for large source blocks due to prohibitive $\mathcal{O}(k^3)$ complexity of ML decoding, in terms of both number of additions and number of multiplications in \mathbb{F}_q .

Given a value of error probability, the efficiency gain of a non-binary code w.r.t. a binary one becomes remarkable for small blocks (i.e., small k). Hence, the use of non-binary codes is appealing for small objects.

III. A CLASS OF RAPTOR CODES OVER \mathbb{F}_q

A Raptor code is obtained by concatenating an outer high rate code (pre-code) with an inner Luby-transform (LT) code [5]. We derive Raptor codes on \mathbb{F}_q from their binary counterparts. In the process, we focus on the class of binary Raptor codes specified in [6], whose encoder is depicted in Fig. 2. A non-systematic LT encoder generates the output symbols from $l = k + s + h$ symbols \mathbf{f} , known as the *intermediate symbols*. These latter symbols are generated by pre-coding the k symbols \mathbf{d}_k . We have $\mathbf{f}^T = [\mathbf{d}_k^T | \mathbf{d}_s^T | \mathbf{d}_h^T]$, where the s symbols \mathbf{d}_s are known as the *LDPC symbols* and the h symbols \mathbf{d}_h as the *half symbols*. The $(s \times k)$ and $(h \times (k + s))$ encoding matrices \mathbf{G}_{LDPC} and \mathbf{G}_H , the encoding matrix \mathbf{G}_{LT} of the inner LT code and the parameters s and h , depend on k and are specified in [6]. A systematic Raptor encoder is obtained through a rate-1 linear pre-coder that generates the k symbols \mathbf{d}_k from the k input symbols \mathbf{c} . This precoder can be represented as the product between \mathbf{c} and a properly chosen full-rank $(k \times k)$ matrix, denoted by \mathbf{G}_T^{-1} in Fig. 2. Adopting the same notation as Section II, we now have $\mathbf{e}_{(0, \dots, n-1)} = \mathbf{G}_{\text{LT}(0, \dots, n-1)} \mathbf{f}$. Note that, as opposed to $\mathbf{G}_{(0, \dots, n-1)}$ for a LRF code, $\mathbf{G}_{\text{LT}(0, \dots, n-1)}$ is a sparse matrix.

We derive Raptor codes over \mathbb{F}_q by extending to non-binary fields the encoder structure depicted in Fig. 2, i.e., by replacing all component encoders with non-binary counterparts. Specifically, we replace each non-zero entry in \mathbf{G}_{LDPC} , \mathbf{G}_H and $\mathbf{G}_{\text{LT}(0, \dots, n-1)}$ with an element picked randomly in $\mathbb{F}_q \setminus \{0\}$.

Next, encoding and decoding are described. The set of constraints on the Raptor output symbols can be represented in a compact way, including the constraints imposed both by

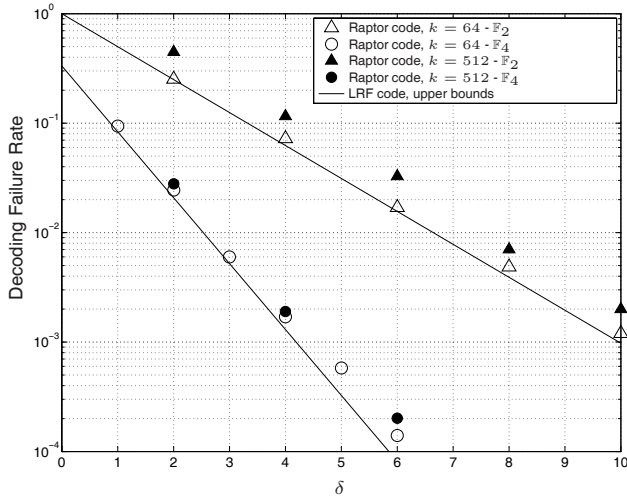


Fig. 3. Decoding failure rate vs. overhead for q -ary Raptor codes ($q = 2, 4$) with $k = 64$ and $k = 512$, compared to the upper bound (valid for all k) on the error probability of LRF codes over \mathbb{F}_2 and \mathbb{F}_4 .

the pre-coder and by the LT encoder, as

$$\mathbf{A}_{(0,\dots,n-1)} \mathbf{f} = \begin{bmatrix} \mathbf{0} \\ \mathbf{e}_{(0,\dots,n-1)} \end{bmatrix}$$

where $\mathbf{0}$ is the length- $(s+h)$ all-zero column vector and $\mathbf{A}_{(0,\dots,n-1)}$ is a $((s+h+n) \times l)$ matrix over \mathbb{F}_q called the *constraint matrix*, given by

$$\mathbf{A}_{(0,\dots,n-1)} = \begin{bmatrix} \mathbf{G}_{\text{LDPC}} & \mathbf{I}_s & \mathbf{Z} \\ & \mathbf{G}_H & \mathbf{I}_h \\ \mathbf{G}_{\text{LT}(0,\dots,n-1)} & & \end{bmatrix}.$$

Here, \mathbf{I}_s and \mathbf{I}_h are the $(s \times s)$ and $(h \times h)$ identity matrices, respectively, and \mathbf{Z} is the $(s \times h)$ all-zero matrix. In general, $\mathbf{A}_{(0,\dots,n-1)}$ is a sparse matrix. We use next the notation $\mathbf{A}_{(j_1, j_2, \dots, j_r)}$ to indicate the $((s+h+r) \times l)$ sub-matrix of $\mathbf{A}_{(0,\dots,n-1)}$ obtained by selecting only the rows of $\mathbf{G}_{\text{LT}(0,\dots,n-1)}$ corresponding to ESIs (j_1, j_2, \dots, j_r) .

Encoding exploits the $(l \times l)$ sub-matrix $\mathbf{A}_{(0,\dots,k-1)}$ formed by the first l rows of $\mathbf{A}_{(0,\dots,n-1)}$. Since encoding is systematic, we have $\mathbf{c} = \mathbf{e}_{(0,\dots,k-1)}$ from which

$$\mathbf{A}_{(0,\dots,k-1)} \mathbf{f} = \begin{bmatrix} \mathbf{0} \\ \mathbf{c} \end{bmatrix}. \quad (5)$$

Encoding consists of first solving (5) through Gaussian elimination to calculate the intermediate symbols $\mathbf{f} \in \mathbb{F}_q^l$, and then performing LT encoding of \mathbf{f} to obtain $\mathbf{e}_{(0,\dots,n-1)}$.

Assume now $k + \delta \geq k$ output symbols with set of ESIs $\{j_1, \dots, j_{k+\delta}\}$ are collected at the decoder. ML decoding is performed by first solving the system

$$\mathbf{A}_{(j_1, j_2, \dots, j_{k+\delta})} \mathbf{f} = \begin{bmatrix} \mathbf{0} \\ \mathbf{e}_{(j_1, j_2, \dots, j_{k+\delta})} \end{bmatrix} \quad (6)$$

through Gaussian elimination to obtain the intermediate symbols \mathbf{f} . Once \mathbf{f} has been recovered, the input symbols are obtained as $\mathbf{c} = \mathbf{G}_{\text{LT}(0,\dots,k-1)} \mathbf{f}$.

Raptor codes present advantages in terms of encoding and decoding complexity w.r.t. LRF counterparts. More specifically, efficient methods for the solution of (5) and (6) exist, which exploit the sparseness of system of equations [7] [8]. Originally proposed for solving sparse systems of equations in \mathbb{F}_2 , the extension of these algorithms to \mathbb{F}_q is straightforward. Although exploiting such approaches the number of required additions and multiplications in \mathbb{F}_q remains cubic (in l), the cubic cost function is multiplied by a very small constant, making the overall complexity affordable.

In Fig. 3 the decoding failure rate under ML decoding of binary Raptor codes from [6], with $k = 64$ and $k = 512$, and of their extension to \mathbb{F}_4 are depicted, as functions of the overhead. The (tight and valid for all k) upper bounds on the performance of LRF codes over \mathbb{F}_2 and \mathbb{F}_4 are also shown. Raptor codes approach closely the upper bounds, and the same was observed for codes on higher order fields. This example shows that Raptor codes over \mathbb{F}_q obtained with the simple proposed technique achieve a performance very close to that of random codes, sharing the same performance advantages of adopting higher order Galois fields.

IV. CONCLUSIONS

In this letter, the performance of LRF codes over \mathbb{F}_q has been analyzed through tight upper and lower bounds, and the advantage of adopting higher-order Galois fields in the code construction illustrated. A class of Raptor codes over \mathbb{F}_q has been then presented showing, through numerical simulation, how their performance is very close to that of LRF codes, while offering a manageable encoding and ML decoding complexity. Non-binary Raptor codes represent a very appealing option in the presence of severe performance versus overhead requirements, especially for small source block sizes. The bounds derived in Proposition 1 can be confidently used to estimate their performance down to moderate error rates.

REFERENCES

- [1] J. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data," *SIGCOMM Comput. Commun. Rev.*, vol. 28, no. 4, pp. 56–67, Oct. 1998.
- [2] M. Shokrollahi, "Raptor codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, June 2006.
- [3] R. Lidl and H. Niederreiter, *Finite Fields*. Cambridge, UK: Cambridge Univ. Press, 1997.
- [4] E. R. Berlekamp, "The technology of error-correcting codes," *Proc. IEEE*, vol. 68, no. 5, pp. 564–593, May 1980.
- [5] M. Luby, "LT codes," in *Proc. 43rd Annual IEEE Symp. on Foundations of Computer Science*, Nov. 2002, pp. 271–282.
- [6] 3GPP TS 26.346 V9.0.0, "Technical specification group services and system aspects; multimedia broadcast/multicast service (MBMS); protocols and codecs (Release 8)," Oct. 2009.
- [7] D. Burshtein and G. Miller, "An efficient maximum likelihood decoding of LDPC codes over the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2837–2844, Nov. 2004.
- [8] E. Paolini, G. Liva, B. Matuz, and M. Chiani, "Pivoting algorithms for maximum-likelihood decoding of LDPC codes over erasure channels," in *Proc. 2009 IEEE Global Telecommunications Conference*, Honolulu, HI, USA, Nov. 2009.