Fountain Coding via Multiplicatively Repeated Non-Binary LDPC Codes

Kenta Kasai, David Declercq, Kohichi Sakaniwa

► To cite this version:

Kenta Kasai, David Declercq, Kohichi Sakaniwa. Fountain Coding via Multiplicatively Repeated Non-Binary LDPC Codes. IEEE Transactions on Communications, 2012, pp.2077-2083. hal-00764122

HAL Id: hal-00764122

<https://hal.science/hal-00764122>

Submitted on 12 Dec 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Fountain Coding via Multiplicatively Repeated Non-Binary LDPC Codes

Kenta Kasai, *Member, IEEE*, David Declercq, *Member, IEEE*,
and Kohichi Sakaniwa, *Senior, IEEE*

Abstract—We study fountain codes transmitted over the binary-input symmetric-output channel. For channels with small capacity, receivers in fountain coding systems need to collect many channel outputs to recover information bits. Since a collected channel output yields a check node in the decoding Tanner graph, the channel with small capacity leads to large decoding complexity. In this paper, we introduce a novel fountain coding scheme with non-binary LDPC codes. The decoding complexity of the proposed fountain code does not depend on the channel. Numerical experiments show that the proposed codes exhibit better performance than conventional fountain codes, especially for moderate number of information bits.

Index Terms—fountain codes, rateless codes, non-binary LDPC codes

I. INTRODUCTION

In this paper, we study fountain codes transmitted over the binary-input symmetric-output channel with capacity C . Fountain codes are a class of erasure-recovering or error-correcting codes which produce limitless sequence of encoded bits from k information bits so that receivers can recover the k information bits from any $(1 + \epsilon)k/C$ encoded bits. We denoted *overhead* by ϵ . Fountain codes encode k information bits to infinitely long coded bits. The purpose of fountain coding is to realize reliable transmissions under reception of **arbitrary** $(1 + \epsilon)k/C$ received signals among coded bits of length ∞ . Even if there is no errors or erasures, this goal can not be achieved by conventional block codes.

The name *fountain* is after water fountains which endlessly produce water drops to entertain people. Designing fountain codes with small overhead is desirable. LT codes [2] and raptor codes [3] are fountain codes which achieve vanishing overhead $\epsilon \rightarrow 0$ in the limit of large information size over the

channel with $C = 1$. Such a channel is viewed as the binary erasure channel (BEC) with zero erasure probability in [2], [3]. By a nice analogy between the BEC and the packet erasure channel (e.g., Internet), fountain codes have been successfully adopted by several industry standards. In such packet erasure channels with many receivers (e.g., 10^6), ack-based protocol and re-transmissions can not be afforded due to high server load. Note that if the channel is not heavily impaired, the re-transmission scheme could be a good choice.

In [4], Etesami et al. investigated raptor codes used over the memoryless binary-input output-symmetric (MBIOS) channels. And they showed that over the binary-input additive white Gaussian noise (BIAWGN) channels with capacity $C \geq 0.49$, raptor codes achieve overhead $\epsilon \leq 0.08$ at bit error rate 10^{-7} with information size $k = 65536$. A raptor code can be viewed as concatenation of an outer high-rate LDPC code and infinitely many single parity-check codes of length d , where d is chosen randomly with probability Ω_d for $d \geq 1$. In [5], Venkiah et al. proposed a joint decoding of the concatenated codes and an optimization method for output degree distributions $\Omega(x) := \sum_{d \geq 1} \Omega_d x^d$ and showed that the optimized codes outperform the conventional ones.

The problems for constructing fountain codes used for general channels with finite inputs are summarized as follows.

- **Problem 1:** The output degree distribution $\Omega(x)$ needs to be optimized for each k . And large check node degree d leads to the large encoding and decoding complexity.
- **Problem 2:** The number of check nodes in the inner codes is given by $(1 + \epsilon)k/C$. This increases as the channel capacity C decreases. Since check node computation is dominant in decoding, the decoding complexity is high for small C .

- **Problem 3:** Conventional research has been focused on fountain codes with large information size and vanishing overhead. This leads to large size of memory devices and transmission latency. Good fountain codes with moderate information size is desired.

In this paper, we will propose a novel fountain coding scheme which is free of those drawbacks.

In this paper, we consider non-binary LDPC codes defined by sparse parity-check matrices over $\text{GF}(2^m)$ for $2^m > 2$. Non-binary LDPC codes were invented by Gallager [6] and, Davey and MacKay [7] found non-binary LDPC codes can outperform binary ones. Non-binary LDPC codes have captured much attention recently due to their decoding performance.

It is known that the irregularity of Tanner graphs helps in improving the decoding performance of binary LDPC codes. On the other hand, it is not the case for the non-binary LDPC codes. Interestingly, the $(2, d_c)$ -regular non-binary LDPC codes over $\text{GF}(2^m)$ are empirically known [8] as the best performing codes for $2^m \geq 64$, especially for short code length. This suggests that, for designing non-binary LDPC codes, one does not need to optimize the degree distributions of Tanner graphs, since $(2, d_c)$ -regular non-binary LDPC codes are best. Furthermore, sparsity of $(2, d_c)$ -regular Tanner graph makes efficient decoding possible.

The rest of this paper is organized as follows. Section II defines the proposed codes. Section III describes the decoding algorithm for the proposed codes. In Section IV, we investigate the asymptotic overhead for the BEC and BIAWGN channels by *density evolution* [9], [10] and Monte Carlo density evolution, respectively. In Section V, for the BIAWGN channels, we compare the decoding performance of proposed codes and the best known codes for short and moderate code length so far. In Section VI, we give a conclusion.

II. FOUNTAIN CODING WITH MULTIPLICATIVELY REPEATED NON-BINARY LDPC CODES

In this section we explain a new fountain coding scheme. The new coding scheme uses a non-binary LDPC code as a pre-code.

In [11], the authors introduced low-rate non-binary codes. The code is a concatenation of $(2, 3)$ -

regular non-binary LDPC code and inner multiplicative repetition codes. In general, low-rate LDPC codes have many check nodes and suffer from the high decoding complexity than high rate codes. One of the remarkable feature of the codes proposed in [11] is that the decoding complexity does not depend on the coding rate. The code exhibits excellent decoding performance for moderate code length and is rate-compatible. We will use the low-rate code [11] with vanishing rate as a fountain code.

We fix a Galois field $\text{GF}(2^m)$ with a primitive element α and its primitive polynomial $\pi(x)$. Once the primitive element is fixed, one can represent each symbol in the Galois field as a binary sequence of length m [12]. For example, with a primitive element $\alpha \in \text{GF}(2^3)$ such that $\pi(\alpha) = \alpha^3 + \alpha + 1 = 0$, each symbol is represented as $0 = (0, 0, 0)$, $1 = (1, 0, 0)$, $\alpha = (0, 1, 0)$, $\alpha^2 = (0, 0, 1)$, $\alpha^3 = (1, 1, 0)$, $\alpha^4 = (0, 1, 1)$, $\alpha^5 = (1, 1, 1)$ and $\alpha^6 = (1, 0, 1)$. We refer to elements in $\text{GF}(2^m)$ as symbols if $m \geq 2$ and bits if $m = 1$. In this setting, k information bits can be represented as k/m symbol sequence $(s_1, \dots, s_{k/m}) \in \text{GF}(2^m)^{k/m}$. We denote the j -th bit of the binary representation of $x_i \in \text{GF}(2^m)$ by $x_i^{(j)}$. As mentioned in Section I, fountain coding has an application for the packet erasure channel. In the analogous packet scenario, what we consider as a packet is not a symbol $s_i^{(j)} \in \text{GF}(2^m)$, but a bit $s_i^{(j)} \in \text{GF}(2)$. We denote the data size of packet by D [bits] which is set to very large, e.g., $D = 12000$ to mitigate the rate loss due to the header overhead in practice. In this paper, we assume $D = 1$ for simplicity. The number of information packets is given by k/m .

A non-binary LDPC code $\mathcal{C}$ over $\text{GF}(2^m)$ is defined by the null space of a sparse $M \times N$ parity-check matrix $H = \{h_{c,v}\}$ defined over $\text{GF}(2^m)$.

$$\mathcal{C} = \{\underline{x} \in \text{GF}(2^m)^N \mid H\underline{x}^T = 0 \in \text{GF}(2^m)^M\}$$

The c -th parity-check equation for $c = 1, \dots, M$ is written as

$$h_{c,1}x_1 + \dots + h_{c,N}x_N = 0 \in \text{GF}(2^m),$$

where $h_{c,1}, \dots, h_{c,N} \in \text{GF}(2^m)$ and $x_1, \dots, x_N \in \text{GF}(2^m)$.

The binary LDPC codes are represented by Tanner graphs with variable and check nodes [13, pp. 75]. The non-binary LDPC codes, in this paper, are also represented by bipartite graphs with

Fig. 1. An example of a pre-code $\mathcal{C}_1$. A non-binary (2,3)-regular LDPC code of rate 1/3 over $\text{GF}(2^m)$. Each variable node represents a symbol in $\text{GF}(2^m)$. Each check node represents a parity-check equation over $\text{GF}(2^m)$. The code length is 18 symbols in $\text{GF}(2^m)$ or equivalently 18m bits.

variable nodes and check nodes, which are also referred to as Tanner graphs. For a given sparse parity-check matrix $H = \{h_{c,v}\}$ over $\text{GF}(2^m)$, the graph is defined as follows. The v -th variable node and c -th check node are connected if $h_{c,v} \neq 0$. By $v = 1, \dots, N$ and $c = 1, \dots, M$, we also denote the v -th variable node and c -th check node, respectively.

A non-binary LDPC code with a parity-check matrix over $\text{GF}(2^m)$ is called (d_v, d_c) -regular if all the columns and all the rows of the parity-check matrix have weight d_v and d_c , respectively, or equivalently all the variable and check nodes have degree d_v and d_c , respectively. Let $\mathcal{C}_1$ be a (2,3)-regular LDPC pre-code defined over $\text{GF}(2^m)$ of length N symbols or equivalently mN bits and of rate 1/3. It can be seen that $N = 3k/m$. The pre-code $\mathcal{C}_1$ has a $2N/3 \times N$ sparse parity-check matrix $H = \{h_{c,v}\}$ over $\text{GF}(2^m)$. The matrix H has row weight 3 and column weight 2. Fig. 1 shows the Tanner graph of $\mathcal{C}_1$ of length $N = 18$ symbols. It can be shown that (2, d_c)-regular non-binary LDPC codes is linear-encodable by using a non-singular zig-zag sub-graph.

We define a new fountain code $\mathcal{C}_\infty : \text{GF}(2)^k \rightarrow \text{GF}(2)^\infty$ by using the following encoding procedure.

- 1) First, map k information bits $(s_1^{(1)}, \dots, s_{k/m}^{(m)})$ to k/m information $\text{GF}(2^m)$ -symbols $(s_1, \dots, s_{k/m})$.
- 2) By the pre-code $\mathcal{C}_1$, encode the k/m information symbols to N symbols $(x_1, \dots, x_N) \in \text{GF}(2^m)^N$.
- 3) Repeat the followings endlessly from $i = 1$ to ∞ .
 - a) Pick uniformly at random $v_i \in [1, N]$, $w_i \in [1, m]$ and $r_i \in \text{GF}(2^m) \setminus \{0\}$.
 - b) Transmit w_i -th bit of $r_i x_{v_i} \in \text{GF}(2^m)$.

The proposed fountain code $\mathcal{C}_\infty$ can be viewed as a non-binary raptor code with a non-binary (2,3)-regular LDPC pre-code and an output degree distribution $\Omega(x) = x$ [3]. Note that $\Omega(x) = x$ does

not mean simple repetition of bits but multiplicative repetition of symbols in $\text{GF}(2^m)$ for the proposed non-binary setting.

It may be possible to improve the overhead by optimizing general output degree distributions $\Omega(x) = \sum_{d=1}^{d_{\max}} x^d$ with some bounded maximum degree $d_{\max}$. In this paper, we focus on the output degree distribution as $\Omega(x) = x$.

III. DECODING SCHEME

We assume that transmission takes place over the MBIOS channel. Specifically, the channel is specified by the transition probability $P(\cdot|\cdot)$ such that $P(y|x) = \Pr(Y = y|X = x)$ where X and Y are the random variable of an input bit x and the channel output y , respectively. We assume that the information bits are chosen with uniform probability.

The most important feature of the fountain coding system is that the decoder does not receive all the channel outputs but collects n channel outputs. The decoder recovers the k information bits from the n collected channel outputs. *Overhead* ϵ is defined [4], [5] by

$$\epsilon = C/R - 1, \quad R = k/n,$$

where C is the channel capacity. Then, the decoder has $n = (1 + \epsilon)k/C$ collected channel outputs.

Note that, in the original setting of fountain codes as in [2],[3], the capacity is set $C = 1$, i.e., all the collected bits are uncorrupted. The aim of fountain coding in this paper is to reliably recover the information bits with small overhead. The overhead $\epsilon = 0$ implies that the information bits are transmitted at rate $R = C$, which is our extreme aim. With infinitely many information bits, raptor codes can achieve $\epsilon = 0$ for the channel with $C = 1$, i.e., the BEC with zero erasure probability. And raptor codes optimized for the BEC exhibit a quite good performance with large information bits $k = 65536$. However, for both the BEC and the general MBIOS channels with $C < 1$, raptor codes exhibit high error floors [4], [14], [15], [5] for moderate information bits with $k \approx 1024$.

For the i -th transmitting bit, the sender picked $v_i \in [1, N]$, $w_i \in [1, m]$ and $r_i \in \text{GF}(2^m) \setminus \{0\}$ uniformly at random and transmitted w_i -th bit of $r_i x_{v_i} \in \text{GF}(2^m)$. This computation $r_i x_{v_i}$ can be viewed as concatenation with a rate-1 inner random code of length m bits. Let I be the set of

Fig. 2. An example of a Tanner graph used for decoding. Some variable nodes are of degree one. The variable nodes of degree one are corresponding to the transmitted symbols whose channel outputs are collected by the decoder. White dots represent bits corresponding to the received channel outputs. Each transmitted bit is encoded in the scheme explained in Section II. It can be seen that the decoder collected 22 channel outputs for this example.

transmitting indices that the receiver collected. It follows $\#I = n$, where $\#I$ denotes the cardinality of I . In other words, for $i \in I$, the receiver collects y_i that is the corrupted version of the i -th transmitted bits. We assume that the decoder knows not only y_i but also the indices v_i, w_i and the multiplicative coefficients r_i for $i \in I$. In practice, this is realized by embedding the indices in the header of packets or synchronization between the sender and the receivers [3]. If the data size of packet is sufficiently large (e.g., $D=12,000$ [bits]), the rate loss due to the header is negligible.

The proposed code $\mathcal{C}_\infty$ can be decoded by the sum-product (SP) decoding algorithm on the Tanner graphs. The SP decoder for the non-binary LDPC codes exchanges probability vectors in $\mathbb{R}^{2^m}$, called *messages*, between variable nodes and check nodes [10]. An example of the Tanner graph used by the decoder is shown in Fig. 2. The variable nodes of degree one with white dots in Fig. 2 represent collected channel outputs. If the SP decoding algorithm is immediately applied to the proposed codes, all the variable nodes and check nodes, including the variable nodes of those multiplicative repetition symbols, are activated, i.e. exchange the messages. However, the messages reached at the variable nodes of degree one do not change messages that sent back from the nodes. Therefore, after the initialization, the decoder does not need to pass the messages all the way to those variable nodes of degree 1 and their adjacent check nodes of degree 2. Consequently, the decoder uses only the Tanner graph of the pre-code $\mathcal{C}_1$. It follows that the complexity of the decoding algorithm does not depend on the number n of collected channel outputs and the channel capacity C . In contrast the

decoding complexity of the conventional fountain codes largely depends on n and C as explained in Section I.

The SP decoding involves mainly 4 parts, i.e. the initialization, the check to variable computation, the variable to check computation, and the tentative decision parts. Let X be the random variable of a transmitted bit x , and let Y be the random variable of the corresponding channel output y . The a posteriori probability $Q(x|y) := \Pr(X = x|Y = y)$, for $x = 0, 1$ and $y \in \mathcal{A}$ is assumed to be known to the decoder, where $\mathcal{A}$ is the receiving alphabet.

initialization :

The decoders collected $n = (1 + \epsilon)k/C$ channel outputs, y_i for $i \in I$, where $\#I = n$. Define $I_v := \{i \in I \mid v_i = v\}$. It follows that $I = \bigcup_{v=1}^N I_v$. For each variable node v in $\mathcal{C}_1$ for $v = 1, \dots, N$, calculate $p_v^{(0)}(x)$ for $x \in \text{GF}(2^m)$ as follows.

$$p_v^{(0)}(x) = \xi \prod_{i \in I_v} \tilde{p}_i^{(0)}(r_i x) \quad (1)$$

$$\tilde{p}_i^{(0)}(x) = \begin{cases} Q(0|y_i) & \text{if the } w_i\text{-th bit of } x \text{ is } 0 \\ Q(1|y_i) & \text{if the } w_i\text{-th bit of } x \text{ is } 1, \end{cases}$$

where ξ is the normalized factor such that $\sum_{x \in \text{GF}(2^m)} p_v^{(0)}(x) = 1$. Each variable node $v = 1, \dots, N$ in $\mathcal{C}_1$ sends the initial message $p_{vc}^{(0)} = p_v^{(0)} \in \mathbb{R}^{2^m}$ to each adjacent check node c . Set the iteration round as $\ell := 0$.

check to variable output :

For each check node $c = 1, \dots, M$ in $\mathcal{C}_1$, let ∂c be the set of the adjacent variable nodes. It holds that $\#\partial c = 3$, since the pre-code $\mathcal{C}_1$ is $(2, 3)$ -regular. Each c has 3 incoming messages $p_{vc}^{(\ell)}$ for $v \in \partial c$ from the 3 adjacent variable nodes. The check node c sends the following message $p_{cv}^{(\ell+1)} \in \mathbb{R}^{2^m}$ to each adjacent variable node $v \in \partial c$.

$$\tilde{p}_{vc}^{(\ell)}(x) = p_{vc}^{(\ell)}(h_{v,c}^{-1}x) \text{ for } x \in \text{GF}(2^m),$$

$$\tilde{p}_{cv}^{(\ell+1)} = \otimes_{v' \in \partial c \setminus \{v\}} \tilde{p}_{v'c}^{(\ell)}$$

$$p_{cv}^{(\ell+1)}(x) = \tilde{p}_{cv}^{(\ell+1)}(h_{v,c}x) \text{ for } x \in \text{GF}(2^m).$$

where $p_1 \otimes p_2 \in \mathbb{R}^{2^m}$ is convolution of $p_1 \in \mathbb{R}^{2^m}$ and $p_2 \in \mathbb{R}^{2^m}$. To be precise,

$$(p_1 \otimes p_2)(x) = \sum_{\substack{y, z \in \text{GF}(2^m) \\ x = y + z}} p_1(y) p_2(z) \text{ for } x \in \text{GF}(2^m).$$

The convolution seems the most complex part of the decoding. The convolutions are efficiently calculated via FFT and IFFT [16], [10]. Update the iteration round as $\ell := \ell + 1$.

variable to check output :

Each variable node $v = 1, \dots, N$ in $\mathcal{C}_1$ has 2 adjacent check nodes since the pre-code $\mathcal{C}_1$ is (2, 3)-regular. Let ∂v be the set of adjacent check nodes. The message $p_{vc}^{(\ell)} \in \mathbb{R}^{2^m}$ sent from v to $c \in \partial v$ is given by

$$p_{vc}^{(\ell)}(x) = p_v^{(0)}(x) \prod_{c' \in \partial v \setminus \{c\}} p_{c'v}^{(\ell)}(x) \text{ for } x \in \text{GF}(2^m).$$

tentative decision

For each $v = 1, \dots, N$, the tentatively estimated v -th transmitted symbol is given as

$$\hat{x}_v^{(\ell)} = \underset{x \in \text{GF}(2^m)}{\text{argmax}} p_v^{(0)}(x) \prod_{c' \in \partial v} p_{c'v}^{(\ell)}(x).$$

If $\hat{x}^{(\ell)} := (\hat{x}_1^{(\ell)}, \dots, \hat{x}_N^{(\ell)})$ forms a codeword of $\mathcal{C}_1$, i.e. $\hat{x}^{(\ell)}$ satisfies every parity-check equation of $\mathcal{C}_1$

$$\sum_{v \in \partial c} h_{c,v} \hat{x}_v^{(\ell)} = 0 \in \text{GF}(2^m)$$

for all $c = 1, \dots, M$, the decoder outputs $\hat{x}^{(\ell)}$ as the estimated codeword. Otherwise repeat the check to variable, variable to check and tentative steps. If the iteration round ℓ reaches at a pre-determined number, the decoder collects more channel outputs and start over the decoding.

IV. ANALYSIS OF ASYMPTOTIC OVERHEAD

In this section, we investigate the overhead ϵ in the limit of many information bits $k \rightarrow \infty$ for the transmissions over the BEC, i.e., $C = 1$. Rathi developed the density evolution which enables the prediction of the decoding performance of the non-binary LDPC codes in the limit of large code length. The density evolution usually gives, for a given code ensemble, the maximum channel erasure probability, referred to as threshold, at which the average decoding erasure probability goes to zero. We will use the density evolution calculating the smallest overhead ϵ at which the average decoding erasure probability goes to zero in the limit of $k \rightarrow \infty$.

The density evolution used in this section was originally developed for the non-binary LDPC code

ensembles with parity-check matrices defined over the general linear group $\text{GL}(\text{GF}(2), m)$. However, Rathi observed that the threshold for the code ensemble defined over $\text{GF}(2^m)$ and $\text{GL}(\text{GF}(2), m)$ also have approximately the same threshold within the order of 10^{-4} . Consequently, we shall evaluate the threshold of the proposed codes by the density evolution for $\text{GL}(\text{GF}(2), m)$.

In the binary case, we can predict the asymptotic decoding performance of LDPC codes transmitted over the general MBIOS channels in the large code length limit by *density evolution* [9]. Density evolution is possible also for the non-binary LDPC codes [17] but computationally intensive and tractable only for the BEC. The analysis for the BEC often helps us to capture the universal properties of LDPC codes.

When the transmission is taken place over the BEC and all-zero codewords are assumed to be sent, the messages, described by probability vectors $(p(x))_{x \in \text{GF}(2^m)}$ of length 2^m in general, can be reduced to linear subspaces of $\text{GF}(2)^m$ [10]. To be precise, for each message in the SP decoding algorithm, a subset of $\{x \in \text{GF}(2)^m \mid p(x) \neq 0\}$ forms a linear subspace of $\text{GF}(2)^m$, where x is the binary representation of $x \in \text{GF}(2^m)$.

For messages in SP decoding, probability vectors $\underline{P} = (P_0, \dots, P_m)$ are used for the density evolution and referred to as *densities*. The i -th entry P_i is the probability that a message forms a subspace of dimension i for $i = 1, \dots, m$. Define two densities $\underline{P}^{(\ell)}$ and $\underline{Q}^{(\ell)}$ as the densities of messages sent from variable nodes and check nodes at the ℓ -th iteration round, respectively. In [18], Rathi proved that the density that outgoing messages from a variable (resp. check) node of degree 3 with two incoming messages of density $\underline{P}$ and $\underline{Q}$ is given by $\underline{P} \boxdot \underline{Q}$ (reps. $\underline{P} \boxtimes \underline{Q}$). The detail calculation of $[\underline{P} \boxdot \underline{Q}]_k$ and $[\underline{P} \boxtimes \underline{Q}]_k$ are given as follows, respectively,

$$\begin{aligned} & \sum_{i=k}^m \sum_{j=k}^{k+m-i} 2^{(i-k)(j-k)} \begin{bmatrix} i \\ k \end{bmatrix} \begin{bmatrix} m-i \\ j-k \end{bmatrix} / \begin{bmatrix} m \\ j \end{bmatrix} P_i Q_j, \\ & \sum_{i=0}^k \sum_{j=k-i}^k 2^{(k-i)(k-j)} \begin{bmatrix} m-i \\ m-k \end{bmatrix} \begin{bmatrix} i \\ k-j \end{bmatrix} / \begin{bmatrix} m \\ m-j \end{bmatrix} P_i Q_j, \end{aligned}$$

where $\begin{bmatrix} m \\ k \end{bmatrix} = \prod_{l=0}^{k-1} \frac{2^m - 2^l}{2^k - 2^l}$ is a 2-Gaussian binomial. Using these two operations of two densities, the

density evolution in [18] gives recursive update equations of $\underline{P}^{(\ell)}$ and $\underline{Q}^{(\ell)}$ for $\ell \geq 0$.

Rathi [10] developed the density evolution for the BEC that tracks probability densities of the dimension of the linear subspaces. For $\ell \geq 0$, the density evolution tracks the probability vectors $\underline{P}^{(\ell)}$ and $\underline{Q}^{(\ell)}$ which are referred to as *densities*. The initial messages in Eq. (1) can be seen as the intersection of d subspaces of the messages received as the channel outputs.

With overhead ϵ , the decoder has $k(1 + \epsilon)/C$ channel outputs transmitted over the channel with capacity C . The number of variable nodes in $\mathcal{C}_1$ is N . It holds that

$$N = \frac{d_c}{d_c - 2} \frac{k}{m},$$

since $\mathcal{C}_1$ is of rate $(d_c - 2)/d_c$ and defined over $\text{GF}(2^m)$. The average number of collected channel outputs per variable node in $\mathcal{C}_1$ is given by

$$\beta := (1 + \epsilon)m \frac{d_c - 2}{d_c} \frac{1}{C}. \quad (2)$$

Denote the number of channel outputs by $n := k(1 + \epsilon)/C = \beta N$. It follows that the probability R_d that a randomly chosen variable node in $\mathcal{C}_1$ has d corresponding channel outputs is given by

$$R_d = \binom{n}{d} \left(\frac{1}{N} \right)^d \left(1 - \frac{1}{N} \right)^{n-d}.$$

It follows that

$$\begin{aligned} \sum_{d \geq 0} R_d x^d &= \left(\frac{1}{N} x + 1 - \frac{1}{N} \right)^n \\ &= \left(1 - \frac{\beta(1-x)}{n} \right)^n \\ &\stackrel{(n \rightarrow \infty)}{=} e^{-\beta(1-x)} \\ &= \sum_{d \geq 0} \frac{\beta^d e^{-\beta}}{d!} x^d. \end{aligned} \quad (3)$$

From this, we see the probability that a randomly chosen variable node in $\mathcal{C}$ has d corresponding channel outputs in the limit of $k \rightarrow \infty$ is

$$\frac{\beta^d e^{-\beta}}{d!}.$$

For the binary case, i.e., $m = 1$, the probability that a variable node has no channel outputs is given by $R_0 = e^{-\beta}$. The threshold of the binary $(2, d_c)$ -regular LDPC codes is given by $\frac{1}{d_c - 1}$ [13]. Therefore

the asymptotic overhead of the proposed scheme with binary $(2, d_c)$ -regular LDPC codes is given as ϵ such that

$$e^{-\beta} = \frac{1}{d_c - 1}.$$

From this and (2), it follows that

$$\epsilon = \frac{d_c}{d_c - 2} \ln(d_c - 1) - 1.$$

The density of the initial messages is given by $\underline{P}^{(0)}$ as follows,

$$\underline{P}^{(0)} = \sum_{d \geq 0} \frac{\beta^d e^{-\beta}}{d!} \overbrace{\underline{E} \boxtimes \dots \boxtimes \underline{E}}^{d \text{ times}},$$

where $\underline{E}$ is a density such that the subspace is of dimension $m - 1$ with probability 1. To be precise, $\underline{E} := (E_0, \dots, E_m)$ with

$$E_i := \begin{cases} 1 & \text{if } i = m - 1 \\ 0 & \text{if } i \neq m - 1. \end{cases}$$

Since the pre-code is a $(2, 3)$ -regular LDPC codes, we have recursive update equations of densities as follows.

$$\begin{aligned} \underline{Q}^{(\ell+1)} &= \underline{P}^{(\ell)} \boxtimes \underline{P}^{(\ell)}, \\ \underline{P}^{(\ell+1)} &= \underline{P}^{(0)} \boxtimes \underline{Q}^{(\ell+1)}. \end{aligned}$$

Since the messages of dimension 0 corresponds to the successful decoding, the asymptotic overhead ϵ^* is defined as follows.

$$\epsilon^* := \inf_{\epsilon \in [0, 1]} \{ \epsilon \in [0, 1] \mid \lim_{\ell \rightarrow \infty} P_0^{(\ell)} = 1 \}.$$

It follows that, in the limit of many information bits $k \rightarrow \infty$, with overhead $\epsilon > \epsilon^*$ the reliable transmissions are possible with the proposed $\mathcal{C}_\infty$.

Table I shows the asymptotic overhead ϵ^* of the proposed code $\mathcal{C}_\infty$ over $\text{GF}(2^m)$ for different $m = 1, \dots, 19$. Table I also lists the asymptotic overheads with $(2, d_c)$ -regular non-binary LDPC pre-code for $d_c = 4, 5$ and 6. It can be seen that the best overhead $\epsilon^* = 0.079$ is attained at $d_c = 3$ and $m = 9$ and the fountain code $\mathcal{C}_\infty$ exhibit very poor overhead if defined on $\text{GF}(2^m)$ with $m = 1$, i.e., the binary field. We will employ $m = 8$ in the next section for its good asymptotic overhead $\epsilon^* = 0.0809$ and friendliness for byte-oriented processors.

TABLE I

ASYMPTOTIC OVERHEAD ϵ^* OF THE PROPOSED CODES $\mathcal{C}_\infty$ WITH A PRE-CODE $(2, d_c)$ -REGULAR NON-BINARY LDPC CODES OVER $\text{GF}(2^m)$ TRANSMITTED OVER THE BEC, I.E., $C = 1.0$. THE BEST OVERHEADS FOR EACH d_c ARE WRITTEN IN BOLD FACE TYPE.

m	$d_c = 3$	$d_c = 4$	$d_c = 5$	$d_c = 6$	$d_c = 7$
1	1.0799	1.1972	1.3104	1.4140	1.5082
2	0.5747	0.6637	0.7492	0.8275	0.8987
3	0.3295	0.4017	0.4709	0.5342	0.5917
4	0.2075	0.2670	0.3244	0.3769	0.4247
5	0.1422	0.1908	0.2384	0.2823	0.3225
6	0.1069	0.1455	0.1846	0.2213	0.2553
7	0.0888	0.1180	0.1495	0.1803	0.2091
8	0.0809	0.1013	0.1262	0.1517	0.1763
9	0.0792	0.0913	0.1104	0.1313	0.1522
10	0.0813	0.0858	0.0996	0.1166	0.1343
11	0.0856	0.0831	0.0921	0.1057	0.1207
12	0.0913	0.0822	0.0871	0.0976	0.1102
13	0.0976	0.0826	0.0837	0.0915	0.1019
14	0.1044	0.0838	0.0815	0.0868	0.0953
15	0.1112	0.0856	0.0801	0.0833	0.0902
16	0.1179	0.0877	0.0795	0.0806	0.0859
17	0.1245	0.0901	0.0792	0.0786	0.0825
18	0.1309	0.0925	0.0793	0.0770	0.0797
19	0.1371	0.0951	0.0797	0.0759	0.0774

A. Monte Carlo Density Evolution

The thresholds for noisy channels ($C=0.5$ and $C=0.1$) are computed with Monte Carlo approximation of density evolution (DE). In the case of non-binary codes, it is no longer possible to rely on histograms to get numerical approximation of the thresholds since the densities of the messages are multi-variable functions. The only way to get thresholds is then to turn to Monte Carlo approximation of the densities of messages, that is using samples drawn from the densities. More details on the algorithm used to compute the thresholds can be found in [19]

In this paper, we have used an extra variance reduction technique in order to get accurate values for the thresholds. A posteriori averaging is used based on 200 threshold computations initialized with different random seeds. The values for $C=1.0$ indicated in Table II are computed with our Monte Carlo algorithm with variance reduction, and we can see that the thresholds match perfectly the exact thresholds of Table II. We therefore have a good confidence in the threshold values for the other capacities $C=0.5$ and $C=0.1$.

TABLE II

THRESHOLD VALUES CALCULATED BY MONTE CARLO DENSITY EVOLUTION FOR $C = 1.0$ (LEFT), $C = 0.5$ (CENTER), $C = 0.1$ (RIGHT).

$C = 1.0$			$C = 0.5$		
m	$d_c = 3$	$d_c = 4$	m	$d_c = 3$	$d_c = 4$
4	0.2075	0.2665	4	0.2907	0.3582
6	0.1069	0.1452	6	0.1516	0.1973
8	0.0809	0.1011	8	0.1150	0.1410
10	0.0813	0.0856	10	0.1144	0.1214
12	0.0912	0.0820	12	0.1274	0.1169

$C = 0.1$		
m	$d_c = 3$	$d_c = 4$
4	0.4047	0.4816
6	0.2073	0.2641
8	0.1376	0.1783
10	0.1155	0.1408
12	0.1129	0.1244

V. NUMERICAL RESULTS

In this section, we present demonstrations of $\mathcal{C}_\infty$ defined over $\text{GF}(2^8)$ with moderate number of information bits. Transmission over the BEC and the BIAWGN channels are considered.

Figure 3 shows the histograms of overheads of $\mathcal{C}_\infty$ defined over $\text{GF}(2^8)$. The number of the information bits is set $k = 192, 512, 1024, 2048, 8192, 16384$ and 32768 from the top to bottom. It can be seen that the overhead is getting concentrated at the asymptotic overhead 0.0809 as increasing k . This was predicted in Section IV.

Figure 4 shows the decoding performance of the proposed fountain code transmitted over the BIAWGN channels with capacity $C = 1.0, 0.5$ and 0.1 . The horizontal axis describes the overhead and the vertical axis describes the block error rate. The performance of best known raptor codes [14], [15], [5] optimized for $k = 1024$ are drawn for comparison. The proposed codes exhibit the better performance than the best known raptor codes.

VI. CONCLUSION

In this paper, we proposed a new simple fountain coding scheme. The conventional design of fountain codes faced with problems posed in Section I. Despite no optimization of the output degree distribution is needed, the proposed fountain codes outperform the best know optimized raptor codes for moderate code length. Furthermore, the decoding complexity does not depends on channels.

Fig. 3. Histograms of the overheads at which the proposed fountain code over $GF(2^8)$ successfully recovers k information bits over the channel with $C = 1.0$. The number of the information bits is set $k = 192, 512, 1024, 2048, 8192, 16384$ and 32768 from the top to bottom. The horizontal axis describes the overhead ϵ . It can be seen that it is getting concentrated at overhead 0.0809 as predicted in Tab. I at $m = 8$.

REFERENCES

- [1] K. Kasai and K. Sakaniwa, "Fountain codes with multiplicatively repeated non-binary LDPC codes," in *Proc. 6th Int. Symp. on Turbo Codes and Related Topics*, Sep. 2010.
- [2] M. Luby, "LT codes," in *Proc. 40th Annual Allerton Conf. on Commun., Control and Computing*, 2002, pp. 271 – 280.
- [3] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, Jun. 2006.
- [4] O. Etesami and A. Shokrollahi, "Raptor codes on binary memoryless symmetric channels," *IEEE Trans. Inf. Theory*, vol. 52, no. 5, pp. 2033 – 2051, May 2006.
- [5] A. Venkiah, C. Poulliat, and D. Declercq, "Jointly decoded raptor codes: analysis and design for the BIAWGN channel," *EURASIP J. Wirel. Commun. Netw.*, vol. 2009, pp. 1–11, 2009.
- [6] R. G. Gallager, *Low Density Parity Check Codes*. in Research Monograph series, MIT Press, Cambridge, 1963.
- [7] M. Davey and D. MacKay, "Low-density parity check codes over $GF(q)$," *IEEE Commun. Lett.*, vol. 2, no. 6, pp. 165–167, Jun. 1998.
- [8] C. Poulliat, M. Fossorier, and D. Declercq, "Design of regular $(2, d_c)$ -LDPC codes over $GF(q)$ using their binary images," *IEEE Trans. Commun.*, vol. 56, no. 10, pp. 1626–1635, Oct. 2008.
- [9] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.
- [10] V. Rathi and R. Urbanke, "Density Evolution, Threshold and the Stability Condition for non-binary LDPC Codes," *IEE Proceedings - Communications*, vol. 152, no. 6, pp. 1069–1074, 2005.
- [11] K. Kasai, D. Declercq, C. Poulliat, and K. Sakaniwa, "Multiplicatively repeated non-binary LDPC codes," *IEEE Trans. Inf. Theory*, Sep. 2011, to appear.
- [12] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: Elsevier, 1977.
- [13] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, Mar. 2008.

Fig. 4. Decoding performance of the proposed fountain codes for the BIAWGN channels with capacity $C=1.0, 0.5$ and 0.1 . The information size is $k = 1024$. The performance of best known raptor codes [14], [15], [5] optimized for $k = 1024$ are drawn for comparison.

- [14] A. Venkiah, C. Poulliat, and D. Declercq, "Rate splitting issue for finite length Raptor codes," in *IEEE Sarnoff Symposium*, Apr. 2008.
- [15] A. Venkiah, "Analyse et optimisation de codes Raptor pour les canaux multicast sans fil," Ph.D. dissertation, Université de Cergy-Pontoise, Nov. 2008, direction: D. Declercq, C. Poulliat.
- [16] D. Declercq and M. Fossorier, "Decoding algorithms for nonbinary LDPC codes over $GF(q)$," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 633–643, Apr. 2007.
- [17] A. Bennatan and D. Burshtein, "On the application of LDPC codes to arbitrary discrete-memoryless channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 3, pp. 417–438, Mar. 2004.
- [18] V. Rathi, "Non-binary LDPC codes and EXIT like functions," Ph.D. dissertation, Lausanne, 2008. [Online]. Available: <http://library.epfl.ch/theses/?nr=4111>
- [19] L. Sassatelli and D. Declercq, "Nonbinary hybrid LDPC codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 5314 – 5334, Oct. 2010.

PLACE
PHOTO
HERE

Kenta Kasai received B.E., M.E., and Ph.D. degrees from the Tokyo Institute of Technology, Tokyo Japan, in 2001, 2003 and 2006, respectively. Since 2006, he has been an assistant professor in the Department of Communications and Integrated Systems, Graduate School of Science and Engineering, Tokyo Institute of Technology. From 2008 to 2009, he was a visiting researcher at ETIS laboratory, Ecole Nationale Supérieure de l'Electronique et de ses Applications, Cergy-Pontoise, France. His current interests include coding theory.

PLACE
PHOTO
HERE

David Declercq was born in June 1971. He graduated his PhD in Statistical Signal Processing 1998. He worked on a new Gaussianity test based on Hermite polynomials properties, and the characterization and the blind identification of non-linear time series. After his PhD, he oriented his researchs towards digital communications, and especially coding theory and iterative decoder design. He started to

work on LDPC codes in 1999, both from the code and decoder design aspects.

Since 2003, he focused on studying and developping LDPC codes and decoders in high order Galois fields $GF(q)$, with $q \gg 2$. A large part of his research projects are related to non binary LDPC codes. He mainly investigated two directions: (i) the design of $GF(q)$ LDPC codes for short and moderate lengths, and (ii) the simplification of the iterative decoders for $GF(q)$ LDPC codes with complexity/performance tradeoff constraints.

David Declercq published more than 20 papers in major journals (IEEE-Trans. Commun., IEEE-Trans. Inf. Theo., Commun. Letters, EURASIP JWCN), and more than 70 papers in major conferences in ICT. He is currently full professor at the ENSEA in Cergy-Pontoise, France, a graduate school in Electrical Engineering. He is a member of the ETIS laboratory, general secretary of the National GRETSI association, and member of the GdR-ISIS direction team. He is currently the recipient of junior position at the "Institut Universitaire de France".

PLACE
PHOTO
HERE

Kohichi Sakaniwa received B.E., M.E., and Ph.D. degrees all in electronic engineering from the Tokyo Institute of Technology, Tokyo Japan, in 1972, 1974 and 1977, respectively.

He joined the Tokyo Institute of Technology in 1977 as a research associate and served as an associate professor from 1983 to 1991. Since 1991 he has been a professor in the Department of Electrical and Electronic En-

gineering, and since 2000 in the Department of Communications and Integrated Systems, Graduate School of Science and Engineering, both in the Tokyo Inst. of Tech. From November 1987 to July 1988, he stayed at the University of Southwestern Louisiana as a Visiting Professor. He received the Excellent Paper Award from the IEICE of Japan in 1982, 1990, 1992 and 1994. His research area includes Communication Theory, Error Correcting Coding, (Adaptive) Digital Signal Processing and so on.

Dr. Sakaniwa is a member of IEEE, IEICE Japan, Information Processing Society of Japan, and Institute of Image Information and Television Engineers of Japan.