Finite-Length Analysis of BATS Codes

Shenghao Yang, Tsz-Ching Ng and Raymond W. Yeung

Abstract

BATS codes were proposed for communication through networks with packet loss. A BATS code consists of an outer code and an inner code. The outer code is a matrix generation of a fountain code, which works with the inner code that comprises random linear coding at the intermediate network nodes. In this paper, the performance of finite-length BATS codes is analyzed with respect to both belief propagation (BP) decoding and inactivation decoding. Our results enable us to evaluate efficiently the finite-length performance in terms of the number of batches used for decoding ranging from 1 to a given maximum number, and provide new insights on the decoding performance. Specifically, for a fixed number of input symbols and a range of the number of batches used for decoding, we obtain recursive formulae to calculate respectively the stopping time distribution of BP decoding and the inactivation probability in inactivation decoding. We also find that both the failure probability of BP decoding and the expected number of inactivations in inactivation decoding can be expressed in a power-sum form where the number of batches appears only as the exponent. This power-sum expression reveals clearly how the decoding failure probability and the expected number of inactivation decrease with the number of batches. When the number of batches used for decoding follows a Poisson distribution, we further derive recursive formulae with potentially lower computational complexity for both decoding algorithms. For the BP decoder that consumes batches one by one, three formulae are provided to characterize the expected number of consumed batches until all the input symbols are decoded.

Index Terms

Network coding, fountain code, LT code, Raptor code, BATS code, finite-length analysis, belief propagation, inactivation decoding, degree-distribution optimization, error probability, error exponent

I. Introduction

Proposed for communication through networks with packet loss, a BATS code consists of an outer code and an inner code [1], [2]. As a matrix generalization of a fountain code, the outer code generates a potentially unlimited number of *batches*, each of which consists of M coded symbols. The inner code comprises (random) linear network

This paper was presented in part at the 2013 IEEE International Symposium on Network Coding.

This work was partially supported by NSFC Grants No. 61471215, the University Grants Committee of the Hong Kong Special Administrative Region, China under Project No. AoE/E-02/08, and a project from the Beijing Institute of Tracking and Telecommunications Technology, Beijing, China.

S. Yang is with the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, China (e-mail: shyang@cuhk.edu.cn).

T.-C. Ng is with the Department of Applied Mathematics and Statistics, Stony Brook University, New York, USA (e-mail: tszching.ng@stonybrook.edu).

R. W. Yeung is with the Institute of Network Coding, The Chinese University of Hong Kong, Hong Kong, China (e-mail: whyeung@ie.cuhk.edu.hk).

coding [3]–[5] at the intermediate network nodes, which is applied on the symbols belonging to the same batch. When $M = 1$, the outer code becomes an LT code (or Raptor code if precode is applied), and network coding of the batches becomes forwarding. Note that if network coding is allowed for different symbols generated by an LT code, the degrees of the received symbols will be changed so that the efficient decoding algorithm of LT codes fails (see more discussion of this issue in [2]). BATS codes resolve this issue: By allowing only network coding for symbols belonging to the same batch, the degrees of batches are not changed by network coding at the intermediate nodes. Sufficient network coding gain can be obtained by using a reasonably large value of M.

BATS codes preserve the salient features of fountain codes, in particular, their ratelessness property and low encoding/decoding complexity. Compared with ordinary random linear network coding schemes [6]–[8], BATS codes not only have lower encoding/decoding complexity, but also smaller coefficient vector overhead and intermediate node caching requirement. Compared with other low-complexity random linear network coding schemes like EC codes [9], Gamma codes [10] and L-chunked codes [11], BATS codes generally achieve higher rates and have the extra feature that an unlimited number of batches can be generated. Applications of BATS codes in various network communication scenarios have been studied in [12]–[14].

The asymptotic performance of BATS codes with belief propagation (BP) decoding has been analyzed in [2]. A sufficient condition for the BP decoder to recover a given fraction of the input symbols with high probability was obtained. This sufficient condition enables us to design BATS codes with good performance for a large number of input symbols (e.g., tens of thousands). It has been verified theoretically for certain special cases and demonstrated numerically for general cases that BATS codes can achieve rates very close to optimality for a given rank distribution of the transfer matrices.

The performance of BATS codes for a relatively small number of input symbols is of important practical interest. For such codes, however, the error bound obtained in the asymptotic analysis is rather loose (if valid), and the degree distribution optimized asymptotically does not give a good performance. Towards designing better BATS codes for a relatively small number of input symbols (e.g., a few hundreds), we analyze in this paper BATS codes with a finite number of input symbols for two decoding algorithms: BP decoding and inactivation decoding.

Before presenting our results, we review some existing results on finite-length analysis of LT codes. For BP decoding, Karp, *et al*. provided a recursive formula to compute the error probability of LT codes for a given number of input symbols [15]. Maneva and Shokrollahi [16] used a random model of the number of coded symbols and obtained a simpler formula for BP decoding. Inactivation decoding has been used for LT/Raptor codes, and compared with BP decoding, it can significantly reduce the required the number of coded symbols for recovering all the input symbols [17], [18]. However, the design of the inactivation decoding of LT codes is mainly guided by heuristics [18].

A. Summary of results

In this paper, we present new results on finite-length analysis of BATS codes, which not only enable us to compute the exact decoding performance for certain pratical cases, but also provide new insights on the decoding performance of both BP decoding and inactivation decoding.

Specifically, for a fixed number of input symbols, recursive formulae are obtained to calculate the stopping time distribution of BP decoding and the inactivation probability in inactivation decoding. These formulae can evaluate efficiently the performance in terms of the number of batches used for decoding ranging from 1 to a given maximum number n, with almost the same computation cost as for evaluating only the decoding performance for n batches. Such mechanisms are interesting on their own. For example, evaluating the stopping time distribution for a range of the number of batches is required in the calculation of the expected coding overhead of BP decoding when the batches are consumed one by one.

We also find that both the probability that BP decoding stops at time t and the probability that an input symbol is inactivated at time t in inactivation decoding can be expressed exactly as the power-sum form $\sum_{i=0}^{2^t-1} c_i e_i^n$, where n is the number of batches, c_i is a function of the number of input symbols and t, and $0 \leq e_i \leq 1$ is a function of the number of input symbols, the degree distribution, the transfer matrix rank distribution and t. Note that i) both e_i and c_i do not depend on n, ii) for both decoders e_i are the same and only the coefficients c_i are different. This expression reveals clearly how the probability of decoding failure (for BP decoding) and the expected number of inactivation (for inactivation decoding) decrease with the number of batches. We obtain the error exponent for BP decoding and characterize the asymptotic behavior of the number of inactive symbols required when the number of received batches goes to infinity.

In network communications, the number of received packets in a time interval is random and typically modelled by a Poisson distribution. When the number of batches used for decoding follows a Poisson distribution, recursive formulae are obtained for calculating respectively the stopping time distribution of BP decoding and the inactivation probability in inactivation decoding, which may have lower computational cost than the corresponding formulae for a fixed number of batches. Our Poisson model of the number of batches is different from the model used for analyzing LT codes by Maneva and Shokrollahi [16], where the number of received coded (output) symbols is the sum of a set of binomial random variables.

The property that an unlimited number of batches can be generated enables another type of BP decoder that consumes the batches one by one. For such a BP decoder, we characterize the expected number of consumed batches until all the input symbols are decoded by three different formulae, which have an infinite-sum, a finite-sum and an integral form, respectively.

The analytical tools provided in this paper can readily be used in the design of BATS codes with a relatively small number of input symbols. We provide optimization examples to illustrate how to use our results for degree distribution optimization.

Our results also provide new analytical tools for LT codes. Detailed discussions on how to apply our results to LT codes are in Section II-E. As far as we know, except for Theorem 1, our results in this paper do not have corresponding results for LT codes in the literature. Subsequent to our work, Blasco, *et al*. obtained independently an iterative formula for computing the expected number of inactive symbols for LT codes [19], which is essentially the same as our formula in Theorem 11 when the batch size is one.

B. Paper Organization

The remainder of this paper is organized as follows. Section II introduces the notations and gives a review on BATS codes. In Section III, BATS codes are analyzed for BP decoding with a fixed number of batches and BP decoding that consumes the batches one by one. We first provide a basic recursive formula to calculate the stopping time distribution of BP decoding with a fixed number of batches (Theorem 1). Based on this formula, the following results about BP decoding of BATS codes are further obtained:

i) A recursive formula is derived to calculate the stopping time distribution of BP decoding with a fixed number n' of batches, where n' ranges from 1 to a given number n (Theorem 2).
ii) The power-sum formula of the stopping time distribution is derived (Theorem 3).
iii) For BP decoding with a fixed number n of batches, the BP decoding error exponent is obtained when n tends to infinity (Theorem 4 and Corollary 5).
iv) For the BP decoder that consumes the batches one by one, two formulae are provided for characterizing the expected number of consumed batches until all the input symbols are decoded (Theorem 6).

In Section IV, BATS codes are analyzed for BP decoding with a Poisson number of batches with mean $\bar{n}$. The following results are obtained.

i) A recursive formula is derived for calculating the stopping time distribution of BP decoding (Theorem 7).
ii) The probability of decoding failure is obtained. This probability decreases exponential with $\bar{n}$, and the rate of decrease is characterized (Theorem 8 and Corollary 9).
iii) For the BP decoder that consumes the batches one by one, the expected number of consumed batches until all the input symbols are decoded is alternatively expressed as an integral of the error probability of BP decoding with a Poisson number of batches (Theorem 10).

The inactivation decoding of BATS codes is analyzed in Section V. For the same number of batches, the probability that an input symbol is inactivated at time t during inactivation decoding shares very similar properties with the probability that BP decoding stops at time t. Therefore, except for the results about the BP decoder that consumes the batches one by one, the results we obtained for BP decoding with a fixed number of batches and with a Poisson number of batches all have corresponding versions for inactivation decoding.

The degree distribution optimizations of BATS codes are discussed in Section VI. Section VII provides the concluding remarks.

II. Preliminaries

After introducing some notations, we discuss the encoding and BP decoding processes of BATS codes.

A. Notations

In this paper, we use 0 as the starting index for vectors and matrices. For a vector $\mathbf{a}$ of length k, we denote by $\mathbf{a}[i{:}j]$ ($0 \le i \le j \le k-1$) the subvector of $\mathbf{a}$ from the i-th to the j-th component. We also write $\mathbf{a}[i] = \mathbf{a}[i{:}i]$, $\mathbf{a}[:] = \mathbf{a}[0{:}k-1]$ and $\mathbf{a}[i:] = \mathbf{a}[i{:}k-1]$ to simplify the notations.

For an $m \times n$ matrix $\mathbf{A}$, we denote by $\text{rk}(\mathbf{A})$ its rank and by $\mathbf{A}[i_1{:}i_2, j_1{:}j_2]$ ($i_1 \le i_2$, $j_1 \le j_2$) the submatrix of $\mathbf{A}$ formed by the entries between the i_1-th and i_2-th rows and the j_1-th to j_2-th columns. We also write $\mathbf{A}[i, j_1{:}j_2] = \mathbf{A}[i{:}i, j_1{:}j_2]$, $\mathbf{A}[i, j{:}] = \mathbf{A}[i{:}i, j{:}n-1]$, $\mathbf{A}[:, j] = \mathbf{A}[0{:}m-1, j{:}j]$, etc.

We use $\mathbf{e}_0$, $\mathbf{I}$ and $\mathbf{0}$ to denote a vector of the form $(1, 0, \ldots, 0)$, an identity matrix and a zero matrix, respectively, where the dimensions are determined by context.

For real numbers x and y, denote their minimum and maximum by $x \wedge y$ and $x \vee y$, respectively.

B. *Encoding of Batches*

Fix a finite field $\mathbb{F}_q$ with size q, called the *base field*. Suppose K input symbols of the base field[1] are transmitted from a source node to a sink node through a network employing linear network coding. Fix an integer $M \ge 1$ called the *batch size*. The outer code of a BATS code generates a potentially unlimited sequence of batches $\mathbf{X}_1, \mathbf{X}_2, \ldots$ formed by

$$\mathbf{X}_i = \mathbf{B}_i \cdot \mathbf{G}_i,$$

where $\mathbf{B}_i$ is a row vector consisting of dg_i input symbols, and $\mathbf{G}_i$ is a $\text{dg}_i \times M$ totally random matrix[2] over the base field, called the *generator matrix*. We call dg_i the *(batch) degree* of the i-th batch X_i. The degrees $\text{dg}_i, i = 1, \ldots$ are i.i.d. random variables with a given distribution $\mathbf{\Psi} = (\Psi_1, \ldots, \Psi_K)$, i.e., $\Pr\{\text{dg}_i = k\} = \Psi_k$. The dg_i input symbols of $\mathbf{B}_i$, called the contributors of batch i, are chosen uniformly at random from all the K input symbols. Denote by A_i the index set of the dg_i symbols in $\mathbf{B}_i$.

The batches are transmitted through a network where the nodes perform linear network coding only among symbols belonging to the same batch. So at the sink node, the received symbols of the i-th batch can be represented by a row vector

$$\mathbf{Y}_i = \mathbf{B}_i \cdot \mathbf{G}_i \cdot \mathbf{H}_i.$$

where $\mathbf{H}_i$ is an M-row random matrix called the *transfer matrix*. The number of columns of $\mathbf{H}_i$ corresponds to the number of symbols received for the i-th batch, which may vary for different batches and is finite. If no packets are received for a batch, $\mathbf{Y}_i$ is the empty vector. We assume that $\mathbf{H}_i$, $i = 1, 2, \ldots$ are independent and follow the same distribution, and $\mathbf{H}_i$, $i = 1, 2, \ldots$ are also independent of the encoding process. The network coding scheme at the intermediate network nodes is called the inner code of a BATS code.

C. *BP Decoding of BATS Codes*

For the BATS code described above and a given number $n \ge 1$, we first describe a BP decoding process that uses n batches. Consider the decoding of n batches $\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n$. Assume that the sink node knows $\mathbf{G}_i\mathbf{H}_i$ and A_i for $i = 1, \ldots, n$. The time index starts at 0 and increases by one after each decoding step. The decoding algorithm

[1] In general, we may consider K input packets, each of which is a vector in a vector space over the base field. But this generalization does not affect the analysis in this paper.

[2] A *totally random* matrix has uniform i.i.d. entries.

modifies $A_i, \mathbf{G}_i$ and $\mathbf{Y}_i$ in each step. For each batch i and time t, let $A_i^{(t)}, \mathbf{G}_i^{(t)}$ and $\mathbf{Y}_i^{(t)}$ be the versions of $A_i, \mathbf{G}_i$ and $\mathbf{Y}_i$ at time t, respectively. When $t = 0$, we have $A_i^{(0)} = A_i, \mathbf{G}_i^{(0)} = \mathbf{G}$ and $\mathbf{Y}_i^{(0)} = \mathbf{Y}_i$. Iterative formulae will be given for these variables at $t > 0$. We call $|A_i^{(t)}|$ and $\text{rk}(\mathbf{G}_i^{(t)}\mathbf{H}_i)$ the degree and the rank of batch i at time t, respectively.

We say a batch i is *decodable* at time t if $\text{rk}(\mathbf{G}_i^{(t)}\mathbf{H}_i) = |A_i^{(t)}|$ (i.e., its degree is equal to its rank), and an input symbol is *decodable* at time t if it contributes to a decodable batch at time t. Denote by $\mathbf{B}_i^{(t)}$ the row vector formed by the input symbols with indices in $A_i^{(t)}$. The associated linear system of batch i at time t is

$$\mathbf{Y}_i^{(t)} = \mathbf{B}_i^{(t)} \cdot \mathbf{G}_i^{(t)} \cdot \mathbf{H}_i.$$

Batch i at time t is decodable means that the above linear system, with $\mathbf{B}_i^{(t)}$ as the variable, has a unique solution.

The decoding algorithm operates as follows. For each time t, a decodable input symbol is selected (if there is more than one such symbols), substituted into the undecodable batches that it contributes to, and marked as *decoded*.[3] Suppose that the j-th input symbol b_j is decoded at time t. We then substitute the decoded input symbol into the batches it contributes to: For each batch i,

i) if $j \in A_i^{(t)}$, then $A_i^{(t+1)} = A_i^{(t)} \setminus \{j\}$, $\mathbf{G}_i^{(t+1)}$ is formed by removing the row g of $\mathbf{G}_i^{(t)}$ corresponding to the j-th input symbol b_j, and $\mathbf{Y}_i^{(t+1)} = \mathbf{Y}_i^{(t)} - b_j g \mathbf{H}_i$; and

ii) if $j \notin A_i^{(t)}$, then $A_i^{(t+1)} = A_i^{(t)}$, $\mathbf{G}_i^{(t+1)} = \mathbf{G}_i^{(t)}$ and $\mathbf{Y}_i^{(t+1)} = \mathbf{Y}_i^{(t)}$.

The decoding stops when there are no decodable input symbols.

The BATS code decoding algorithm described above uses a given number n of batches, and is denoted by $\text{BP}(n)$. For $\text{BP}(n)$, we are interested in the time when the decoding stops, which is equal to the number of input symbols that are decoded. For example, if $\text{BP}(n)$ stops at time zero, no input symbols are decoded; while if $\text{BP}(n)$ stops time time K, all the input symbols are decoded. We will characterize the distribution of the stopping time of $\text{BP}(n)$ in this paper.

Now let us see how to benefit from the unlimited number of batches. Suppose that the encoder generates n batches. When $\text{BP}(n)$ stops without all the input symbols decoded, the encoder can generate more batches to resume the BP decoding procedure. We define the following *rateless BP decoder* BP^* that consumes the batches one by one. BP^* starts by fetching the first batch. For n batches fetched ($n = 1$ to start with), $\text{BP}(n)$ is applied. If $\text{BP}(n)$ stops with all the input symbols decoded, BP^* stops; otherwise, one more batch is fetched and $\text{BP}(n+1)$ is applied. Since the number of batches is unlimited, BP^* will eventually stop with all the input symbols decoded.

For BP^*, we are interested in the number of batches consumed when the decoding stops. We will characterize the distribution of the number of batches consumed, as well as the expected number of batches consumed by BP^* in this paper.

[3]Note that in each step, the choice of the decodable input symbol to substitute does not affect the time when the decoding stops (see [2, Appendix B]).

D. Solvability of a Batch

Let us check the probability that a batch is decodable when its degree has a specific value. According to the algorithm of $\mathrm{BP}(n)$, if a batch is decodable at time t, it is decodable at all time $t' > t$ until the associated linear system has no variable left. We say a batch is decodable for the first time at time t if it is decodable at time t, but is not decodable at time $t-1$.

For $s = 0, 1, \ldots, M$, let $\mathbf{G}^{(s)}$ be an $s \times M$ totally random matrix over the base field $\mathbb{F}_q$. Let $\mathbf{H}$ be a random matrix with the same distribution of $\mathbf{H}_1$. Define

$$\hbar_s \triangleq \Pr\left\{\mathrm{rk}\left(\begin{bmatrix}\mathbf{G}^{(1)} \\ \mathbf{G}^{(s)}\end{bmatrix}\mathbf{H}\right) = \mathrm{rk}(\mathbf{G}^{(s)}\mathbf{H}) = s\right\}, \tag{1}$$

$$\hbar'_s \triangleq \Pr\{\mathrm{rk}(\mathbf{G}^{(s)}\mathbf{H}) = s\}, \tag{2}$$

where $\mathbf{G}^{(1)}$ and $\mathbf{G}^{(s)}$ are statistically independent. Note that $\hbar_s$ is the probability that a batch is decodable for the first time when its degree is s. Once a batch becomes decodable, it remains to be decodable until all its contributors are decoded. Note that $\hbar'_s = \sum_{k\geq s} \hbar_k$ for $0 \leq s \leq M$ and $\hbar_s = 0$ for $s > M$.

The explicit forms of $\hbar_s$ and $\hbar'_s$ will not be directly used in the analysis of this paper, but are useful in the numerical evaluation. According to [2],

$$\hbar_s = \sum_{k=s}^{M} \frac{\zeta_s^k}{q^{k-s}} h_k \quad \text{and} \quad \hbar'_s = \sum_{k=s}^{M} \zeta_s^k h_k,$$

where

$$\zeta_s^k \triangleq \begin{cases}(1-q^{-k})(1-q^{-k+1})\cdots(1-q^{-k+s-1}) & s > 0 \\ 1 & s = 0.\end{cases}$$

and $h_k \triangleq \Pr\{\mathrm{rk}(\mathbf{H}) = k\}$ is the rank distribution of $\mathbf{H}$. Henceforth, we assume the rank distribution $\mathbf{h} = (h_0, \ldots, h_M)$ of $\mathbf{H}$ is known. Let

$$\bar{h} = \sum_{i=1}^{M} i h_i.$$

Note that when the field size is large, e.g., $q = 2^8$, the difference between h_k and $\hbar_k$ becomes negligible.

We say that BP decoding can start if the probability that a batch is decodable at time 0 is nonzero. The following proposition is intuitive.

Proposition 1. *BP decoding can start if and only if there exists d, $1 \leq d \leq M$ such that $\Psi_d \sum_{k=d}^{M} h_k > 0$.*

Proof: A batch with degree $d \leq M$ is decodable at time 0 with probability $\Psi_d \hbar'_d$, and a batch with degree $d > M$ is not decodable at time 0 with probability one. The proposition is proved by noting that $\hbar'_d > 0$ if and only if $\sum_{k=d}^{M} h_k > 0$. ■

When $h_0 = 1$, for example, BP decoding cannot start. Since the case that BP decoding cannot start is trivial, we are primarily interested in the case that BP decoding can start, which is implied in the rest of this paper unless otherwise specified. When BP decoding can start, let r_{BP} be the smallest integer $s \in \{1, \ldots, M\}$ such that $\Psi_s \sum_{k=s}^{M} h_k > 0$. By Proposition 1, r_{BP} is well defined.

E. Special Case: LT Codes

When the batch size is one, BATS codes described above become LT codes. In this case, since each batch has only one coded symbol, network coding at the intermediate nodes becomes forwarding. Then h_0, the probability that the batch transfer matrix has rank zero, can be regarded as the end-to-end erasure rate.

Due to the random generator matrix, the degree of a batch may be larger than the degree of the coded symbol[4] in the batch because certain entries of the generator matrix may be equal to 0. For a batch with degree d, the degree of the coded symbol in the batch is k ($k \leq d$) with probability $\binom{d}{k}(1-q^{-1})^k q^{-(d-k)}$. Our analysis (to be provided) uses the degree distribution of batches, which can be converted into the degree distribution of coded symbols. But we have a simpler approach to apply our analytical results to LT codes with respect to the degree distribution of the coded symbols.

When $M = 1$, instead of a random generator matrix, we can use the generator matrix with all entries being the identity of the base field. Then the degree of a batch is the same as the degree of the coded symbol in the batch. Redefining (1) and (2) for $\mathbf{G}^{(s)}$ containing only the identity of the base field, we have

$$\hbar_0 = h_0, \hbar'_0 = 1 \text{ and } \hbar_1 = \hbar'_1 = h_1.$$

So when $M = 1$, substituting the above values of $\hbar_s$ and $\hbar'_s$ into the formulae to be obtained in this paper, we obtain the corresponding results for LT codes with respect to the degree distribution of the coded symbols.

III. Stopping Time of BP Decoding

In this section, we analyze the BP decoder for a fixed number of input symbols. We study the following performance measues for BP decoding:

i) The distribution of the stopping time of $\mathrm{BP}(n)$, which induces the error probability of $\mathrm{BP}(n)$ (i.e., the probability that $\mathrm{BP}(n)$ fails to recover all the input symbols);
ii) The decrease rate of the error probability of $\mathrm{BP}(n)$ when n increases;
iii) The distribution of the number of batches consumed by BP^*; and
iv) The expected number of batches consumed by BP^*.

A. Basic Recursive Formula

We start with the performance of $\mathrm{BP}(n)$. Let $R_n^{(t)}$ be the number of decodable input symbols at time t (which is also called the *input ripple size* in the literature of LT codes). The probability that $\mathrm{BP}(n)$ stops at time t is

$$P_{\text{stop}}(t|n) \triangleq \Pr\left\{R_n^{(t)} = 0, R_n^{(\tau)} > 0, \tau < t\right\}.$$

Let $C_n^{(t)}$ be the number of undecodable batches at time t. Define an $(n+1) \times (K-t+1)$ matrix $\mathbf{\Lambda}_n^{(t)}$ as

$$\mathbf{\Lambda}_n^{(t)}[c, r] \triangleq \Pr\left\{C_n^{(t)} = c, R_n^{(t)} = r, R_n^{(\tau)} > 0, \tau < t\right\}, \tag{3}$$

[4] A coded symbol can be expressed as a linear combination of the input symbols. The degree of the coded symbol is defined as the number of non-zero coefficients in the linear combination.

where $c = 0, 1, \ldots, n$ and $r = 0, 1, \ldots, K - t$. With the above equality we have

$$P_{\text{stop}}(t|n) = \sum_{c=0}^{n} \mathbf{\Lambda}_n^{(t)}[c, 0]. \tag{4}$$

We will express $\mathbf{\Lambda}_n^{(t)}$ in terms of $\mathbf{\Lambda}_n^{(t-1)}$, so that we can calculate $\mathbf{\Lambda}_n^{(t)}$ recursively for $t = 0, \ldots, K$.

Let

$$\mathrm{Bi}(k; n, p) \triangleq \binom{n}{k} p^k (1-p)^{n-k}$$

and

$$\mathrm{hyge}(k; n, i, j) \triangleq \begin{cases} \frac{\binom{i}{k}\binom{n-i}{j-k}}{\binom{n}{j}} & \max\{0, i + j - n\} \le k \le \min\{i, j\} \\ 0 & o.w. \end{cases}$$

be the p.m.f. of the binomial distribution and the hypergeometric distribution, respectively. We obtain the following recursion of $\mathbf{\Lambda}_n^{(t)}$, which together with (4) gives a formula to calculate $P_{\text{stop}}(t|n)$.

Theorem 1. *Consider a BATS code with K input symbols, n batches, degree distribution $\mathbf{\Psi}$, rank distribution $\mathbf{h}$ of the transfer matrix, and batch size M. When BP decoding can start, we have*

$$\mathbf{\Lambda}_n^{(0)}[c, :] = \mathrm{Bi}(c; n, 1 - \rho_0)\mathbf{e}_0 \mathbf{Q}_0^{n-c}, \tag{5}$$

and for $t > 0$,

$$\mathbf{\Lambda}_n^{(t)}[c, :] = \sum_{c'=c}^{n} \mathrm{Bi}(c; c', 1 - \rho_t)\mathbf{\Lambda}_n^{(t-1)}[c', 1{:}]\mathbf{Q}_t^{c'-c} \tag{6}$$

where ρ_t and $\mathbf{Q}_t$ are defined as follows:

i) $\rho_0 = \sum_{s=0}^{M} p_{0,s}$, *where* $p_{0,s} = \Psi_s \hbar'_s$.

ii) *For $t > 0$,*

$$\rho_t = \frac{\sum_{s=0}^{M} p_{t,s}}{1 - \sum_{\tau=0}^{t-1} \sum_{s=0}^{M} p_{\tau,s}}$$

and

$$p_{t,s} = \begin{cases} \hbar_s \displaystyle\sum_{d=s+1}^{s+t} \Psi_d \frac{d}{K} \mathrm{hyge}(d - s - 1; K - 1, d - 1, t - 1) & s + t \le K, \\ 0 & s + t > K. \end{cases}$$

iii) *For $t = 0, 1, \ldots, K$, $\mathbf{Q}_t$ is a $(K - t + 1) \times (K - t + 1)$ matrix with*

$$\mathbf{Q}_t[i, j] = \sum_{s=j-i}^{j \wedge M} \frac{p_{t,s}}{\sum_{s'=0}^{M} p_{t,s'}} \mathrm{hyge}(i + s - j; K - t, i, s) \tag{7}$$

for $0 \vee (j - M) \le i \le j \le K - t$, and $\mathbf{Q}_t[i, j] = 0$ otherwise.

Proof: The proof is left to Appendix I. The idea is to characterize the corresponding probability transition matrix between two consecutive decoding times. ■

The notations defined in the above theorem deserve some explanations. First, $p_{t,s}$ is the probability that a batch is decodable for the first time at time t and has batch degree s at time t assuming that the decoding can start (when $t=0$) or does not stop at the previous time (when $t>0$). Let

$$p_t \triangleq \sum_{s=0}^{M} p_{t,s}.$$

We know that p_t is the probability that a batch is decodable for the first time at time t assuming that the decoding can start (when $t=0$) or does not stop at the previous time (when $t>0$).

Lemma 1.

$$p_{t,s} \begin{cases} = 0, & \text{for } t+s < r_{\mathrm{BP}}, \\ > 0, & \text{for } t = 0, \text{ and } s = r_{\mathrm{BP}}, \\ > 0, & \text{for } t \geq 1, t+s \geq r_{\mathrm{BP}} \text{ and } s < r_{\mathrm{BP}}. \end{cases}$$

Proof: By Proposition 1, we have $\Psi_r = 0$ for $r = 1, \ldots, r_{\mathrm{BP}} - 1$, $\Psi_{r_{\mathrm{BP}}} > 0$ and $\hbar_s > 0$ for $s \leq r_{\mathrm{BP}}$. The lemma then follows from the definition of $p_{t,s}$. ∎

Lemma 1 implies that $p_t > 0$ for $t = 0, 1, \ldots, K$. So the denominators in the definitions of ρ_t for $t > 0$ and $\mathbf{Q}_t$ for $t \geq 0$ are all positive. We also note that ρ_t ($t > 0$) is the probability that a batch is decodable at time t under the condition that it is not decodable at time $t-1$. The following properties about ρ_t and p_t are straightforward and they are proved in Appendix II.

Lemma 2.

i) *For* $0 \leq t \leq K$

$$\prod_{\tau=0}^{t} (1-\rho_\tau) = 1 - \sum_{\tau=0}^{t} p_\tau.$$

ii) *For* $0 < t \leq K$

$$\rho_t \prod_{\tau=0}^{t-1} (1-\rho_\tau) = p_t.$$

Matrix $\mathbf{Q}_t$ can be regarded as a transition matrix. Suppose that k batches become decodable at time t and we generate new decodable input symbols from these k batches one batch after another. Define random variable $Z_0 = R_n^{(t-1)} - 1$ for $t > 0$ or $Z_0 \equiv 0$ for $t = 0$, and for $i = 1, \ldots, k$ define Z_i as the total number of decodable input symbols after having generated new decodable input symbols from the first i decodable batches. Note that $Z_k = R_n^{(t)}$. Then $Z_0, \ldots, Z_k$ forms a homogeneous Markov chain with the transition matrix $\mathbf{Q}_t$.

To evaluate the formulae in Theorem 1, we first calculate $p_{t,s}$ for $t = 0, 1, \ldots, K$ and $s = 0, 1, \ldots, M$, which takes $\mathcal{O}(K^2 M)$ real number operations. We then calculate ρ_t and $\mathbf{Q}_t$ for $t = 0, 1, \ldots, K$ using $\mathcal{O}(KM)$ and $\mathcal{O}(K^2 M^2)$ real number operations, respectively. Thus, it totally takes $\mathcal{O}(K^2 M^2)$ real number operations to calculate ρ_t and $\mathbf{Q}_t$. Note that $p_{t,s}$, ρ_t and $\mathbf{Q}_t$ do not depend on n, and are determined by K, $\boldsymbol{\Psi}$ and $\mathbf{h}$ only. Once they are calculated, we can use them in the evaluation of $\boldsymbol{\Lambda}_n^t$ for different values of n. Note that the matrix $\mathbf{Q}_t$ has at most

$M+1$ non-zero entries in each column. So the vector-matrix multiplication takes $\mathcal{O}(KM)$ real number operations. Since a total of $\mathcal{O}(Kn^2)$ such vector-matrix multiplications are used in the formulae, the complexity for computing $P_{\text{stop}}(t|n)$ using Theorem 1 is $\mathcal{O}(K^2M^2 + K^2n^2M)$ real number operations.

Example 1 ($\Psi_1 = 1$)**.** Consider a BATS code with $\Psi_1 = 1$ and $h_0 < 1$. In this special case, every batch has degree one. The condition $h_0 < 1$ means that BP decoding can start. It can be calculated that $p_{0,1} = \hbar_1'$ and $p_{0,s} = 0$ for $s \neq 1$. All the components of $\mathbf{Q}_0$ are zero except that $\mathbf{Q}_0[i,i] = i/K$ for $i = 1, \ldots, K$ and $\mathbf{Q}_0[i, i+1] = 1 - i/K$ for $i = 0, \ldots, K-1$. When $t > 0$, we have $p_{t,0} = \hbar_0/K$ and $p_{t,s} = 0$ for $s > 0$, and $\mathbf{Q}_t$ is the identity matrix.

Example 2 ($t = K$)**.** When $t = K$, all the input symbols are decoded so that all the batches have degree 0. We have $p_{K,0} = \hbar_0 \sum_{d=1}^{K} \Psi_d \frac{d}{K}$, $p_{K,s} = 0$ for $s > 0$, $\rho_K = 1$ and $\mathbf{Q}_K = [1]$.

Example 3 (LT Codes)**.** Letting $M = 1$, $\hbar_0 = h_0$, $\hbar_0' = 1$ and $\hbar_1 = \hbar_1' = h_1$ in Theorem 1, we obtain

$$p_{0,1} = \Psi_1 h_1 \text{ and } p_{0,0} = 0,$$

and for $t > 0$,

$$p_{t,0} = h_0 \sum_{d=1}^{t} \Psi_d \frac{d}{K} \frac{\binom{K-d}{t-d}}{\binom{K-1}{t-1}} = h_0 \sum_{d=1}^{t} \Psi_d \frac{\binom{t-1}{d-1}}{\binom{K}{d}},$$

$$p_{t,1} = \begin{cases} h_1 \sum_{d=2}^{t+1} \Psi_d \dfrac{d(d-1)}{K} \dfrac{\binom{K-d}{t-d+1}}{\binom{K-1}{t-1}} = h_1 \sum_{d=2}^{t+1} \Psi_d (K-t) \dfrac{\binom{t-1}{d-2}}{\binom{K}{d}} & t < K, \\ 0 & t = K. \end{cases}$$

The matrix $\mathbf{Q}_t$, $t = 0, 1, \ldots, K-1$ has the following expression: for $i = 0, \ldots, K-t$,

$$\mathbf{Q}_t[i,i] = \frac{p_{t,0}}{p_t} + \frac{p_{t,1}}{p_t} \frac{i}{K-t},$$

for $i = 0, \ldots, K-t-1$,

$$\mathbf{Q}_t[i, i+1] = \frac{p_{t,1}}{p_t} \left(1 - \frac{i}{K-t}\right),$$

and $\mathbf{Q}_t[i,j] = 0$ otherwise.

Karp *et al.* [15] has given a formula for LT codes to recursively calculate the joint distribution of the number of decodable received symbols (called *output ripple size*) and the number of undecodable received symbols at each decoding step. Note that the distribution of output ripple size determines the distribution of the input ripple size. Their formula is given in a polynomial form and has an evaluation bit-complexity $\mathcal{O}(n^3 \log^2(n) \log\log(n))$ based on polynomial evaluation and interpolation.

Note that it is possible to extend the approach in [15] for $M > 1$, i.e., recursively calculating the joint distribution of the number of decodable batches and the number of undecodable batches. When $M > 1$, decodable batches with different degrees must be considered separately and M recursive formulae must be provided for each positive degree value of the decodable batches. The evaluation complexity of this extension increases exponentially with M (see an outline of this extension in [20, Appendix]). Our approach here, which instead tracks the number of decodable input symbols and the number of undecodable batches at each step, gives a formula with complexity equal to a quadratic function of M. Further, our formula is given in a matrix form, which facilitates certain analyses as we will demonstrate in this paper.

B. Stopping Time Distribution

For a given number n, $\mathbf{\Lambda}_n^{(t)}$ can be calculated recursively for $t = 0, \dots, K$ using Theorem 1 and hence the stopping time distribution $P_{\text{stop}}(\cdot|n)$ can be calculated using (4). But for applications that will be discussed later in this section, we may want to calculate $P_{\text{stop}}(\cdot|n')$ for $n' = 0, 1, \dots, n$, where $n > 0$ is a given integer. Using the formula in Theorem 1, we have to run the program for each value of n'. In Theorem 2, we will propose a new formula that can simplify the calculation of $P_{\text{stop}}(\cdot|n)$ for a range of n.

Theorem 2. *For $n \geq 0$ and $t \geq 0$,*

$$P_{\text{stop}}(t|n) = \sum_{c=0}^{n} \binom{n}{c} \left(1 - \sum_{\tau=0}^{t} p_\tau\right)^c \mathbf{\Lambda}_{n-c}^{(t)}[0,0], \tag{8}$$

where the first row of the matrices $\mathbf{\Lambda}_{n'}^{(t)}$, $n' = 0, 1, \dots, n$ can be computed by the following recursion: For $n' = 0, 1, \dots, n$,

$$\mathbf{\Lambda}_{n'}^{(0)}[0,:] = (p_0 \mathbf{Q}_0)^{n'}[0,:], \tag{9}$$

and for $t > 0$

$$\mathbf{\Lambda}_{n'}^{(t)}[0,:] = \sum_{c=0}^{n'} \binom{n'}{c} \mathbf{\Lambda}_{n'-c}^{(t-1)}[0,1:](p_t \mathbf{Q}_t)^c. \tag{10}$$

Proof: The formula in Theorem 1 implies a relation between $\mathbf{\Lambda}_n^{(t)}[c,:]$ ($c > 0$) and $\mathbf{\Lambda}_{n-1}^{(t)}[0,:]$. See the details in Appendix III. ■

For a given number $n > 0$, the above theorem provides us a new representation of $P_{\text{stop}}(\cdot|n)$ in terms of $\mathbf{\Lambda}_{n'}^{(t)}[0,0]$ for $n' = 0, 1, \dots, n$, and a recursive formula given by (9) and (10) to calculate $\mathbf{\Lambda}_{n'}^{(t)}[0,:]$ for $t = 0, 1, \dots, K$ and $n' = 1, \dots, n$. To evaluate the formulae in the above theorem, we first use (9) to calculate $\mathbf{\Lambda}_i^{(0)}[0,:]$ for $i = 0, 1, \dots, n$. For $t > 0$, we use the following recursive formulae induced by (10) to calculate $\mathbf{\Lambda}_i^{(t)}[0,:]$ for $i = 0, 1, \dots, n$:

$$\begin{aligned}
\mathbf{\Lambda}_0^{(t)}[0,:] &= \binom{0}{0}\mathbf{\Lambda}_0^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^0 \\
\mathbf{\Lambda}_1^{(t)}[0,:] &= \binom{1}{0}\mathbf{\Lambda}_1^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^0 + \binom{1}{1}\mathbf{\Lambda}_0^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^1 \\
\mathbf{\Lambda}_2^{(t)}[0,:] &= \binom{2}{0}\mathbf{\Lambda}_2^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^0 + \binom{2}{1}\mathbf{\Lambda}_1^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^1 + \binom{2}{2}\mathbf{\Lambda}_0^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^2 \\
&\vdots \\
\mathbf{\Lambda}_n^{(t)}[0,:] &= \binom{n}{0}\mathbf{\Lambda}_n^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^0 + \binom{n}{1}\mathbf{\Lambda}_{n-1}^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^1 + \dots + \binom{n}{n}\mathbf{\Lambda}_0^{(t-1)}[0,1:](p_t\mathbf{Q}_t)^n.
\end{aligned}$$

This theorem is more convenient to use when we want to calculate $P_{\text{stop}}(\cdot|n')$ for $n' = 1, \dots, n$, which has the same complexity $\mathcal{O}(K^2M^2 + K^2n^2M)$ as calculating $P_{\text{stop}}(\cdot|n)$ only using Theorem 1.

C. Power-Sum Formula

Matrix $\mathbf{Q}_t$ defined in Theorem 1 is upper-triangular. The following lemma, proved in Appendix II, shows that $\mathbf{Q}_t$ is also diagonalizable.

Lemma 3. *Matrix $\mathbf{Q}_t$ is diagonalizable, i.e.,*

$$\mathbf{Q}_t = \mathbf{U}_t \mathbf{D}_t \mathbf{U}_t^{-1},$$

where $\mathbf{D}_t$ is a diagonal matrix with $\mathbf{D}_t[i,i] = \mathbf{Q}_t[i,i]$, $\mathbf{U}_t$ is an upper-triangular matrix with $\mathbf{U}_t[i,j] = \binom{K-t-i}{j-i}$ for $i \leq j$, and $\mathbf{U}_t^{-1}$ is an upper-triangular matrix with $\mathbf{U}_t^{-1}[i,j] = (-1)^{j-i}\binom{K-t-i}{j-i}$ for $i \leq j$.

In the above decomposition, the degree and rank distributions only affect $\mathbf{D}_t$, i.e., the eigenvalues of $\mathbf{Q}_t$. The matrix $\mathbf{U}_t$ depends only on K and t. We also notice that $\mathbf{U}_t[1{:},1{:}] = \mathbf{U}_{t+1}$ and $\mathbf{U}_t^{-1}[1{:},1{:}] = \mathbf{U}_{t+1}^{-1}$. Substituting the above decomposition of $\mathbf{Q}_t$ into Theorem 2, we obtain another formula for $P_{\text{stop}}(t|n)$ with an power-sum form.

Theorem 3. *For $n \geq 0$ and $t \geq 0$,*

$$P_{\text{stop}}(t|n) = \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \left(1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0]\right)^n,$$

where row vector $\mathbf{V}_{t,i}$ and diagonal matrix $\mathbf{\Delta}_{t,i}$ are defined as follows:

i) $\mathbf{V}_{0,0} \triangleq \mathbf{U}_0[0,:]$ *and* $\mathbf{\Delta}_{0,0} \triangleq p_0 \mathbf{D}_0$,

ii) *For $t \geq 0$ and $i = 0, 1, \ldots, 2^t - 1$,*

$$\begin{aligned}
\mathbf{V}_{t+1,i} &= \mathbf{V}_{t,i}[1{:}], \\
\mathbf{\Delta}_{t+1,i} &= \mathbf{\Delta}_{t,i}[1{:},1{:}] + p_{t+1}\mathbf{D}_{t+1}, \\
\mathbf{V}_{t+1,2^t+i} &= -\mathbf{V}_{t,i}[0]\mathbf{U}_t[0,1{:}], \\
\mathbf{\Delta}_{t+1,2^t+i} &= \mathbf{\Delta}_{t,i}[0,0]\mathbf{I} + p_{t+1}\mathbf{D}_{t+1}.
\end{aligned}$$

Proof: This theorem can be proved by substituting the diagonal decomposition of $\mathbf{Q}_t$ in Lemma 3 into Theorem 2. The details can be found in Section III. ■

The formula in Theorem 3 is a linear combination of 2^t n-th powers, where the number of batches n appears only in the power, but in neither $\mathbf{V}_{t,i}$ nor $\mathbf{\Delta}_{t,i}$. It is now easy to see that $P_{\text{stop}}(t|n)$ decreases exponentially with n, which will be made explicit in the next subsection. Note that $\mathbf{V}_{t,i}[0]$ are integers determined by K, t and i, but not n, and can be both positive and negative. According to the definition, we also know that for $t = 0, 1, \ldots, K-1$,

$$0 < \sum_{\tau=0}^{t} p_\tau - \mathbf{\Delta}_{t,i}[0,0] < 1.$$

We prefer Theorem 2 to Theorem 3 for numerical evaluation due to two reasons. First, due to the 2^t n-th power for $t = 0, 1, \ldots, K$, the computation complexity increases exponentially with K. Second, the absolute value of $\mathbf{V}_{t,i}[0]$ can be very large, so that the accuracy of the numerical evaluation is difficult to guarantee if we use a fixed number of significant digits.

D. Error Probability and Error Exponent

For $\mathrm{BP}(n)$, we say a decoding error occurs if the decoder cannot recover all the K input symbols, i.e., the decoder stops before time K. Hence, the corresponding *error probability* is

$$P_{\text{err}}(n) = \sum_{t=0}^{K-1} P_{\text{stop}}(t|n) = 1 - P_{\text{stop}}(K|n).$$

Using Theorem 2, we can calculate $P_{\text{err}}(n)$ efficiently.

The asymptotic decrease rate of the error probability of $\mathrm{BP}(n)$ with respect to n can be characterized using the *BP error exponent* of BATS codes defined as

$$\mathrm{EE}_{\mathrm{BP}} = \lim_{n\to\infty} \frac{-\log(P_{\text{err}}(n))}{n}.$$

Define for $0 \le t \le K$

$$q_t = 1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,0}[0,0]. \tag{11}$$

Recall the definition of r_{BP} following Proposition 1. The following theorem enables us to characterize the BP error exponent.

Theorem 4. *Suppose that BP decoding can start. We have*

i) $P_{\text{stop}}(0|n) = q_0^n$*;*

ii) *For* $1 \le t < r_{\mathrm{BP}}$*,* $P_{\text{stop}}(t|n) = 0$ *for all* $n \ge 1$*;*

iii) *For* $t \ge r_{\mathrm{BP}}$*,*

$$\lim_{n\to\infty} \frac{-\log P_{\text{stop}}(t|n)}{n} = -\log q_t.$$

Proof: This theorem is derived using Theorem 3. See the details in Appendix III. ■

Remark 1. The above theorem says that $\mathbf{V}_{t,0} q_t^n$ is the dominating term of $P_{\text{stop}}(t|n)$ when n is large.

Corollary 5. *The BP error exponent of BATS codes satisfies*

$$\mathrm{EE}_{\mathrm{BP}} = -\log q^*.$$

where $q^* \triangleq q_0 \vee (\vee_{t=r_{\mathrm{BP}}}^{K-1} q_t) = \vee_{t=0}^{K-1} q_t$*.*

Proof: The corollary follows the above theorem and $P_{\text{err}}(n) = \sum_{t=0}^{K-1} P_{\text{stop}}(t|n)$. The equality $q_0 \vee (\vee_{t=r_{\mathrm{BP}}}^{K-1} q_t) = \vee_{t=0}^{K-1} q_t$ follows by $q_0 \ge q_t$ for $t < r_{\mathrm{BP}}$. (By checking the proof of Theorem 4, we know $q_t = 1 - \sum_{\tau=0}^{t} p_\tau$ for $t < r_{\mathrm{BP}}$.) ■

We can obtain the maximum BP error exponent by solving the following linear program for given K and rank distribution:

$$\begin{aligned} &\min_{\Psi,x} x \\ &\text{s.t. } q_t \le x, \quad t = 0,1,\ldots,K-1. \end{aligned} \tag{12}$$

The variables in the above optimization are the degree distribution and x.

E. Number of Batches Consumed

We now consider the decoder BP^* described in Section II-C. We are interested in the number of batches consumed when BP^* decodes all the input symbols, which is denoted by N_{BP^*}. It is possible to characterize the distribution of N_{BP^*} using the error probability of $\mathrm{BP}(n)$. The event $N_{\mathrm{BP}^*} \geq n$ is the same as the event that $\mathrm{BP}(n-1)$ stops with less than K input symbols decoded. So we have for $n \geq 1$,

$$\Pr\{N_{\mathrm{BP}^*} \geq n\} = P_{\text{err}}(n-1). \tag{13}$$

The *coding overhead* of a BATS code is defined as

$$\text{CO} = \sum_{i=1}^{N_{\mathrm{BP}^*}} \text{rk}(\mathbf{H}_i) - K.$$

We are interested in the expected coding overhead

$$\mathbb{E}[\text{CO}] = \mathbb{E}[N_{\mathrm{BP}^*}]\,\mathbb{E}[\text{rk}(\mathbf{H})] - K = \mathbb{E}[N_{\mathrm{BP}^*}] \sum_r r h_r - K.$$

where the first equality holds by Wald's equality. Since both K and $\sum_r r h_r$ are given, we now calculate $\mathbb{E}[N_{\mathrm{BP}^*}]$.

Theorem 6.

$$\mathbb{E}[N_{\mathrm{BP}^*}] = \sum_{n=0}^{\infty} P_{\text{err}}(n) \tag{14}$$

$$= \sum_{t=0}^{K-1} \sum_{i=0}^{2^t-1} \frac{\mathbf{V}_{t,i}[0]}{\sum_{\tau=0}^{t} p_\tau - \mathbf{\Delta}_{t,i}[0,0]}. \tag{15}$$

Proof: We can write by (13) that

$$\mathbb{E}[N_{\mathrm{BP}^*}] = \sum_{n=1}^{\infty} n \Pr\{N_{\mathrm{BP}^*} = n\} = \sum_{n=1}^{\infty} \Pr\{N_{\mathrm{BP}^*} \geq n\} = \sum_{n=0}^{\infty} P_{\text{err}}(n) = \sum_{t=0}^{K-1} \sum_{n=0}^{\infty} P_{\text{stop}}(t|n).$$

The proof is completed by applying Theorem 3:

$$\sum_{n=0}^{\infty} P_{\text{stop}}(t|n) = \sum_{n=0}^{\infty} \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \left(1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0]\right)^n = \sum_{i=0}^{2^t-1} \frac{\mathbf{V}_{t,i}[0]}{\sum_{\tau=0}^{t} p_\tau - \mathbf{\Delta}_{t,i}[0,0]}.$$

■

The above theorem provides two formulae for $\mathbb{E}[N_{\mathrm{BP}^*}]$. We prefer (14) for numerical evaluations than (15). Fix a sufficiently large integer n_2, and we can approximate $\mathbb{E}[N_{\mathrm{BP}^*}]$ by

$$\mathbb{E}[N_{\mathrm{BP}^*}] \approx \sum_{n=0}^{n_2} P_{\text{err}}(n). \tag{16}$$

The approximation error is exponentially small in terms of n_2 (implied by Corollary 5).

F. Evaluation Example I

We use an example to demonstrate the evaluation results of the formulae in this section. Consider a BATS code with $K = 256$, $q = 256$, $M = 16$ and the rank distribution in Table I. The rank distribution is the one of the length-2 homogeneous line network with link erasure probability 0.2 (see [2, Section VII-A] for a formula for the

TABLE I
THE RANK DISTRIBUTION FOR EVALUATION EXAMPLES. HERE THE BATS CODE HAS $q = 256$ AND $M = 16$. THE VALUE OF h_0 IS 0 AND OMITTED IN THE TABLE.

h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8
0	0	0	0	0.0001	0.0004	0.0025	0.0110
h_9	h_{10}	h_{11}	h_{12}	h_{13}	h_{14}	h_{15}	h_{16}
0.0387	0.1040	0.2062	0.2797	0.2339	0.1038	0.0190	0.0008

rank distribution). Here $\bar{h} = 11.91$ is an upper bound on the achievable rates of BATS codes (in terms of packet per batch).

Three degree distributions $\boldsymbol{\Psi}^{\text{asy}}$, $\boldsymbol{\Psi}^{\text{BP}}$ and $\boldsymbol{\Psi}^{\text{mee}}$ are used in our evaluation (given in Table III in Appendix VI), where $\boldsymbol{\Psi}^{\text{asy}}$ is obtained by solving the degree-distribution optimization problem induced by the asymptotic analysis of BATS code in [2]; $\boldsymbol{\Psi}^{\text{mee}}$ is obtained by solving (12); and $\boldsymbol{\Psi}^{\text{BP}}$ is obtained by modifying $\boldsymbol{\Psi}^{\text{asy}}$ using an approach to be discussed in Section VI. We evaluate the error probability of $\mathrm{BP}(n)$, $n = 1, \ldots, 200$ for the three degree distributions. See Fig. 1 for an illustration of the evaluation results.

We first observe that for all the degree distributions, the error probability decreases exponentially fast in n when n is large, which matches the findings in Section III-D. For $\boldsymbol{\Psi}^{\text{mee}}$, the BP error decrease rate is the fastest asymptotically among these three distributions. We also observe that the error probability is almost one for small n. For the general case, the error probability for $n < K/\bar{h}$ is all close to one, which can be bounded as follows.

Proposition 2. *For any $n < K/\bar{h}$,*

$$P_{\text{err}}(n) \geq 1 - \exp\left(-\frac{1}{3}\left(\frac{K}{n\bar{h}} - 1\right)^2 \frac{\bar{h}}{M} n\right).$$

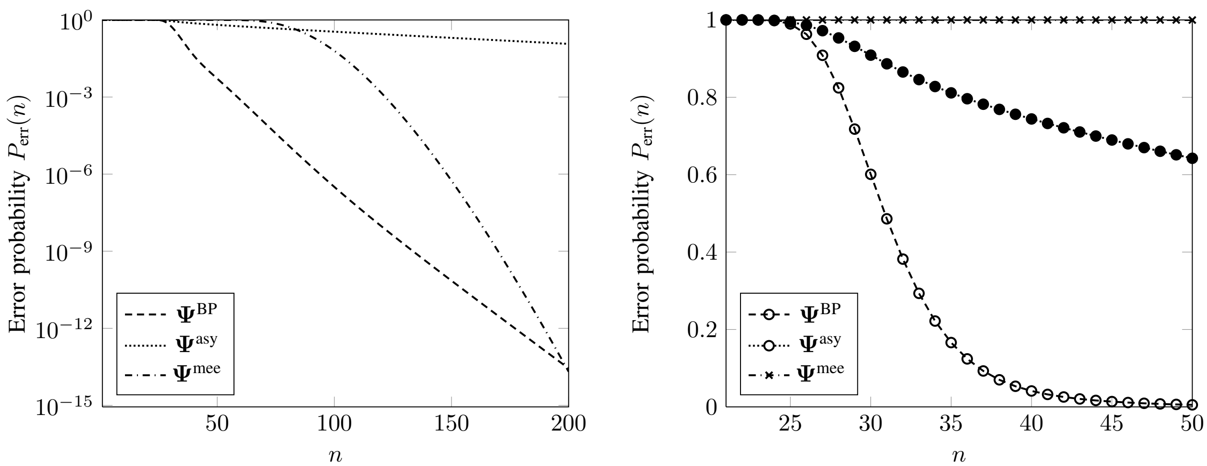
Proof: We have

$$P_{\text{err}}(n) \geq \Pr\left\{\sum_{i=1}^{n} \text{rk}(\mathbf{H}_i) < K\right\} = 1 - \Pr\left\{\sum_{i=1}^{n} \text{rk}(\mathbf{H}_i) \geq K\right\},$$

where $\text{rk}(\mathbf{H}_i), i = 1, \ldots, n$ are independent random variables with generic distribution $\mathbf{h}$. The proof is an application of the Chernoff bound. ■

For relatively small values of n, the lower bound is loose. In this example, $K/\bar{h} = 21.49$. The bound in the above proposition gives $P_{\text{err}}(21) \geq 0.0029$, but our evaluations show that $P_{\text{err}}(21) = 1.0000$ for all the three degree distributions.

From Fig. 1(b), we observe that $\boldsymbol{\Psi}^{\text{BP}}$ has the lowest error probability for n from 25 to 50. For example, if we want to achieve an error probability 0.01, it is sufficient to use $n = 47$ for $\boldsymbol{\Psi}^{\text{BP}}$. Unless we desire an extremely low error probability, e.g., 10^{-14}, $\boldsymbol{\Psi}^{\text{BP}}$ is preferred for BP decoding. It is not surprising that the degree distribution obtained from the asymptotic analysis does not perform well for short block lengths.



(a) n from 1 to 200

(b) Zoom-in for n from 21 to 50

Fig. 1. $P_{\text{err}}(n)$ for different degree distributions. Here $K = 256$, $q = 256$ and the rank distribution is given in Table I.

TABLE II

PERFORMANCE COMPARISON OF THE THREE DEGREE DISTRIBUTIONS GIVEN IN TABLE III.

Degree Distribution	Average Degree	EE_{BP}	$\mathbb{E}[N_{\text{BP}^*}]$	$\mathbb{E}$[CO]
Ψ^{asy}	53.8	0.0107	$>$ 97	$>$ 1154
Ψ^{BP}	49.3	0.1562	32.1	382.4
Ψ^{mee}	111.1	0.5692	82.5	983.1

The BP error exponents of the three degree distributions are given in Table II. Actually, Ψ^{mee} is the degree distribution that achieves the optimal value of (12) for $K = 256$, $q = 256$ and the rank distribution in Table I.

The values of $\mathbb{E}[N_{\text{BP}^*}]$ and the expected coding overhead of the three degree distributions can be found using the approximation in (16). The trend of $\sum_{n=0}^{n_2} P_{\text{err}}(n)$ when n_2 increases can be found in Fig. 2. We see that for both Ψ^{BP} and Ψ^{mee}, the approximation converges fast due to the fast decrease of the corresponding error probability $P_{\text{err}}(n)$. For the range of n_2 in the evaluation, the value of $\sum_{n=0}^{n_2} P_{\text{err}}(n)$ does not converge for Ψ^{asy}. But the value of $\sum_{n=0}^{n_2} P_{\text{err}}(n)$ for Ψ^{asy} provides a lower bound for $\mathbb{E}[N_{\text{BP}^*}]$ that is sufficient for us to compare these three degree distributions in terms of $\mathbb{E}[N_{\text{BP}^*}]$.

IV. POISSON NUMBER OF BATCHES

In this section, we study the stopping time of $\text{BP}(\tilde{N})$ where $\tilde{N}$ is a Poisson distributed random variable, i.e., we assume that the number of batches used by the BP decoder follows a Poisson distribution. In network communications, the number of received packets in a given time interval is usually modelled by a Poisson distribution. Therefore, the Poisson model for the number of the batches is useful for evaluating the performance of BATS code in such network models. In addition, the analysis of $\text{BP}(\tilde{N})$ will provide an alternative formula for

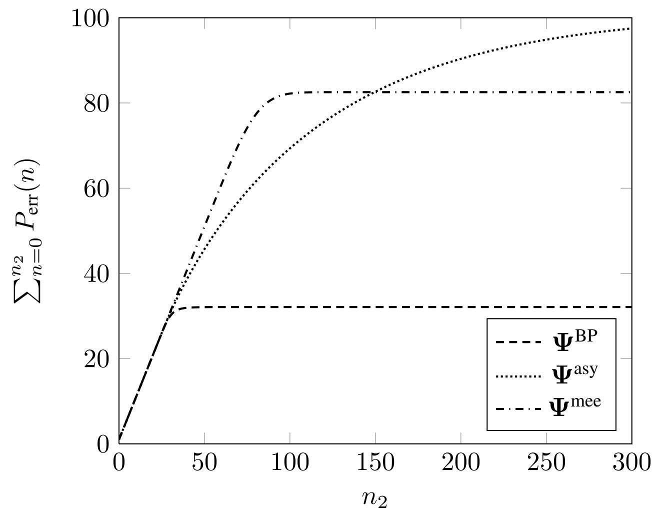


Fig. 2. The trends of $\sum_{n=0}^{n_2} P_{\text{err}}(n)$ when n_2 increases for the three degree distributions given in Table III.

calculating $\mathbb{E}[N_{\text{BP}^*}]$.

A. Recursive Formulae

The Poisson random variable $\tilde{N}$ can be represented by is expectation $\bar{n}$, with

$$\Pr\left\{\tilde{N} = n\right\} = \frac{\bar{n}^n}{n!} e^{-\bar{n}}.$$

For any integer t ($0 \le t \le K$) and real value $\bar{n} > 0$, define a row-vector $\tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t)}$ of length $K - t + 1$ as

$$\tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t)}[r] \triangleq \sum_n \Pr\left\{\tilde{N} = n\right\} \Pr\left\{R_n^{(t)} = r, R_n^{(\tau)} > 0, \tau < t\right\}, \quad r = 0, 1, \ldots, K - t.$$

According to the definition in (3), we have

$$\tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t)} = \sum_n \Pr\{\tilde{N} = n\} \sum_{c=0}^{n} \mathbf{\Lambda}_n^{(t)}[c, :]. \tag{17}$$

Denote by $\tilde{P}_{\text{stop}}(t|\bar{n})$ the probability that $\text{BP}(\tilde{N})$ stops at time t, where $\mathbb{E}[\tilde{N}] = \bar{n}$. We see that

$$\tilde{P}_{\text{stop}}(t|\bar{n}) = \tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t)}[0] = \sum_n \Pr\{\tilde{N} = n\} P_{\text{stop}}(t|n), \tag{18}$$

where the second equality follows from (4) and (17). The above formula of $\tilde{P}_{\text{stop}}(t|\bar{n})$ can be calculated using Theorem 2 with complexity $\mathcal{O}(K^2M^2 + K^2 n_{\max}^2 M)$ of real number operations, where we use the first $n_{\max}$ summands for approximation. Due to the fast decrease of $\Pr\{\tilde{N} = n\}$ when $n > \bar{n}$, we may choose $n_{\max}$ such that $\sum_{n=n_{\max}+1}^{\infty} \Pr\{\tilde{N} = n\}$ is small, which gives an upper bound on the approximation error tolerance.

In the following, we show that $\tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t)}$ can be expressed using a different formula, which provides a new perspective on the quantity $\tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t)}$ and a simpler method of evaluating $\tilde{P}_{\text{stop}}(t|\bar{n})$ than (18) for certain cases. Define the matrix exponential $\exp(\mathbf{A})$ for a square matrix $\mathbf{A}$ as

$$\exp(\mathbf{A}) \triangleq \sum_{i=0}^{\infty} \frac{\mathbf{A}^i}{i!}.$$

Theorem 7. *Consider BP decoding of a BATS code with K input symbols, degree distribution $\mathbf{\Psi}$, and transfer matrix rank distribution $\mathbf{h}$. When the number of batches used by BP decoding is Poisson distributed with expectation $\bar{n}$, for any integer $t \geq 0$,*

$$\tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t)} = \tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t-1)}[1{:}] \exp\left(\bar{n} p_t (\mathbf{Q}_t - \mathbf{I})\right), \tag{19}$$

where $\tilde{\mathbf{\Lambda}}_{\bar{n}}^{-1}[1{:}] \triangleq \mathbf{e}_0$.

Proof: We show the proof of (19) for $t = 0$ here. The remainder of the proof can be found in Appendix IV. Substituting $\Pr\{\tilde{N} = n\}$ and $\mathbf{\Lambda}_n^{(0)}[c, :]$ given in Theorem 1, we have

$$\begin{aligned}
\tilde{\mathbf{\Lambda}}_{\bar{n}}^{(0)} &= \sum_n \frac{\bar{n}^n}{n!} e^{-\bar{n}} \sum_{c \leq n} \mathrm{Bi}(c; n, 1 - \rho_0) \mathbf{e}_0 \mathbf{Q}_0^{n-c} \\
&= \sum_{c,n: c \leq n} \frac{\bar{n}^n}{n!} e^{-\bar{n}} \binom{n}{c} (1 - \rho_0)^c (\rho_0)^{n-c} \mathbf{e}_0 \mathbf{Q}_0^{n-c} \\
&= e^{-\bar{n}} \mathbf{e}_0 \sum_{c,n: c \leq n} \frac{(\bar{n}(1 - \rho_0))^c}{c!} \frac{(\bar{n}\rho_0 \mathbf{Q}_0)^{n-c}}{(n-c)!}.
\end{aligned}$$

By defining $m = n - c$ and using matrix exponential, we can further simplify the above formula as

$$\begin{aligned}
\tilde{\mathbf{\Lambda}}_{\bar{n}}^{(0)} &= e^{-\bar{n}} \mathbf{e}_0 \sum_c \frac{(\bar{n}(1 - \rho_0))^c}{c!} \sum_m \frac{(\bar{n}\rho_0 \mathbf{Q}_0)^m}{m!} \\
&= e^{-\bar{n}} \mathbf{e}_0 \exp(\bar{n}(1 - \rho_0)) \exp(\bar{n}\rho_0 \mathbf{Q}_0) \\
&= \mathbf{e}_0 \exp(-\bar{n}\rho_0) \exp(\bar{n}\rho_0 \mathbf{Q}_0) \\
&= \mathbf{e}_0 \exp(\bar{n}\rho_0(\mathbf{Q}_0 - \mathbf{I})),
\end{aligned} \tag{20}$$

where the last equality is obtained using the fact that $\exp(\mathbf{A}) \exp(\mathbf{B}) = \exp(\mathbf{A} + \mathbf{B})$ whenever $\mathbf{AB} = \mathbf{BA}$. ■

The formula provided in the above theorem involves only the distribution of the number of decodable input symbols at each time. In other words, for a Poisson number of batches, it is not necessary to consider the joint distribution of the number of decodable input symbols and the number of undecodable batches as in Theorem 1.

B. Evaluation Approaches

To evaluate the formula in Theorem 7, we need to calculate the matrix exponential efficiently. Using the decomposition of $\mathbf{Q}_t$ given in Lemma 3, we have

$$\begin{aligned}
\exp\left(\bar{n} p_t (\mathbf{Q}_t - \mathbf{I})\right) &= \mathbf{U}_t \exp(\bar{n} p_t (\mathbf{D}_t - \mathbf{I})) \mathbf{U}_t^{-1} \\
&= \mathbf{U}_t \begin{bmatrix} \exp(\bar{n} p_t (\mathbf{D}_t[0,0] - 1)) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \exp(\bar{n} p_t (\mathbf{D}_t[K-t, K-t] - 1)) \end{bmatrix} \mathbf{U}_t^{-1}.
\end{aligned}$$

However, this approach is not suitable for numerical calculation for moderately large K (e.g., $K > 60$) due to the loss of significance.

The calculation of matrix exponential has been extensively studied (see [21] for a survey). We will discuss two approaches for evaluating the formula in Theorem 7. One of the widely used approach for calculating matrix exponential is the scaling and squaring method [22], which has been implemented in many numerical computing environments (e.g., the **expm** function in Matlab). For a square matrix $\mathbf{A}$, the computational cost of the algorithm in [22] for computing $\exp(\mathbf{A})$ is $O(\log \|\mathbf{A}\|_1)$ matrix multiplications (of size $\mathbf{A}$) with the truncation error no larger than a specified tolerance (e.g., the unit roundoff or 2^{-32}). Recall that the complexity of computing the quantities $\{p_{t,s}, p_t\mathbf{Q}_t\}_{0\leq t\leq K, 0\leq s\leq M}$ is $\mathcal{O}(K^2M^2)$. Since each row of the matrix $\mathbf{Q}_t$ has at most $M+1$ non-zero entries, the computational cost of the algorithm in [22] for computing $\exp\left(\bar{n}p_t(\mathbf{Q}_t - \mathbf{I})\right)$ is $\mathcal{O}(KM\log\bar{n})$. Taking into account of the vector-matrix multiplication, the overall complexity for computing $\tilde{P}_{\text{stop}}(t|\bar{n})$, $t = 0, 1, \ldots, K$ is $\mathcal{O}(K^2M^2 + K^2M\log\bar{n} + K^3)$ real number operations.

Now we discuss another approach. What we are calculating in (19) is a vector multiplying the matrix exponential, also called an action of the matrix exponential. In general, for a row vector $\mathbf{v}$ and a square matrix $\mathbf{A}$, the computation of $\mathbf{v}\exp(\mathbf{A})$ can be done by $O(\|\mathbf{A}\|_1)$ multiplications of a vector with matrix $\mathbf{A}$, using the algorithm in [23]. So for our case, the overall complexity for computing $\tilde{P}_{\text{stop}}(t|\bar{n})$, $t = 0, 1, \ldots, K$ is $\mathcal{O}(K^2M^2 + K^2M\bar{n})$ real number operations, taking the structure of $\mathbf{Q}_t$ into consideration. When $\bar{n}$ is relatively small, we would prefer the approach using the action of the matrix exponential, while when $\bar{n}$ is large, we would choose the first approach to compute the matrix exponential directly.

We may want to evaluate $\tilde{P}_{\text{stop}}(t|\bar{n})$ for $\bar{n} \in \{i\bar{n}_0 : i = 1, \ldots, i_{\max}\}$, where $\bar{n}_0$ is a small number (e.g. 1 or 0.5). In this case, we calculate the matrix exponential $\exp\left(\bar{n}_0 p_t(\mathbf{Q}_t - \mathbf{I})\right)$ directly with complexity $\mathcal{O}(KM)$ using the algorithm in [22]. Then, we calculate $\exp\left(i\bar{n}_0 p_t(\mathbf{Q}_t - \mathbf{I})\right)$ for $i = 1, \ldots, i_{\max}$ recursively using

$$\exp\left(i\bar{n}_0 p_t(\mathbf{Q}_t - \mathbf{I})\right) = \left(\exp\left(\bar{n}_0 p_t(\mathbf{Q}_t - \mathbf{I})\right)\right)^i.$$

The overall complexity for computing $\tilde{P}_{\text{stop}}(t|\bar{n})$, $t = 0, 1, \ldots, K$, $\bar{n} \in \{i\bar{n}_0 : i = 1, \ldots, i_{\max}\}$ is $\mathcal{O}(K^2M^2 + K^3 i_{\max})$ real number operations.

C. Error Probability and Exponent

Similar to Theorem 4, we have the following characterization of $\tilde{P}_{\text{stop}}(t|\bar{n})$. Recall r_{BP} defined after Proposition 1, and q_t defined in (11).

Theorem 8. *Suppose that BP decoding can start. We have*

i) $\tilde{P}_{\text{stop}}(0|\bar{n}) = \exp(-\bar{n}(1-q_0))$;
ii) *For* $1 \leq t < r_{\text{BP}}$, $\tilde{P}_{\text{stop}}(t|\bar{n}) = 0$; *and*
iii) *For* $t \geq r_{\text{BP}}$,

$$\lim_{\bar{n}\to\infty} \frac{-\log \tilde{P}_{\text{stop}}(t|\bar{n})}{\bar{n}} = 1 - q_t.$$

Proof: Using Theorem 3 and (18), we get

$$\begin{aligned}\tilde{P}_{\text{stop}}(t|\bar{n}) &= \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \sum_n \frac{\bar{n}^n}{n!} e^{-\bar{n}} \left(1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0]\right)^n \\ &= \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \exp\left(-\bar{n}\left(\sum_{\tau=0}^{t} p_\tau - \mathbf{\Delta}_{t,i}[0,0]\right)\right).\end{aligned}$$

The proof then follows similarly as the one of Theorem 4 and the details are left to Appendix IV. ■

Let $\tilde{P}_{\text{err}}(\bar{n}) \triangleq 1 - \tilde{P}_{\text{stop}}(K|\bar{n})$, i.e., the probability that $\mathrm{BP}(\tilde{N})$ cannot recover all the input packets. Recall that $q^* = \vee_{t=0}^{K-1} q_t$.

Corollary 9.

$$\lim_{\bar{n}\to\infty} \frac{-\log \tilde{P}_{\text{err}}(\bar{n})}{n} = 1 - q^*.$$

Proof: The proof is similar to that of Corollary 5 except that Theorem 8 instead of Theorem 4 is applied, and hence it is omitted. ■

D. Another Formula for $\mathbb{E}[N_{\mathrm{BP}^*}]$

We can use $\tilde{P}_{\text{err}}(\bar{n})$ to characterize $\mathbb{E}[N_{\mathrm{BP}^*}]$, the expected number of batches consumed by BP^*.

Theorem 10.

$$\mathbb{E}[N_{\mathrm{BP}^*}] = \int_0^\infty \tilde{P}_{\text{err}}(x)\mathrm{d}x = \sum_{t=0}^{K-1} \int_0^\infty \tilde{\mathbf{\Lambda}}_x^{(t)}[0]\mathrm{d}x.$$

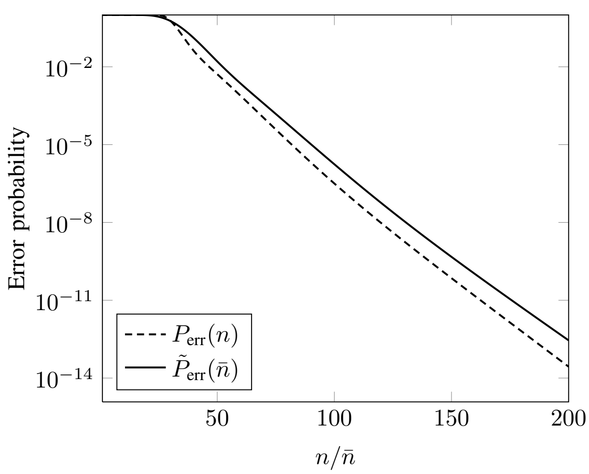
Proof: We have

$$\begin{aligned}\int_0^\infty \tilde{P}_{\text{err}}(x)\mathrm{d}x &= \int_0^\infty \textstyle\sum_{t=0}^{K-1} \tilde{P}_{\text{stop}}(t|x)\mathrm{d}x \\ &= \int_0^\infty \textstyle\sum_{t=0}^{K-1} \sum_n \displaystyle\frac{x^n}{n!} e^{-x} P_{\text{stop}}(t|n)\mathrm{d}x \\ &= \int_0^\infty \sum_n \frac{x^n}{n!} e^{-x} P_{\text{err}}(n)\mathrm{d}x \\ &= \sum_n \frac{P_{\text{err}}(n)}{n!} \int_0^\infty x^n e^{-x}\mathrm{d}x \\ &= \sum_n P_{\text{err}}(n) = \mathbb{E}[N_{\mathrm{BP}^*}]\end{aligned}$$

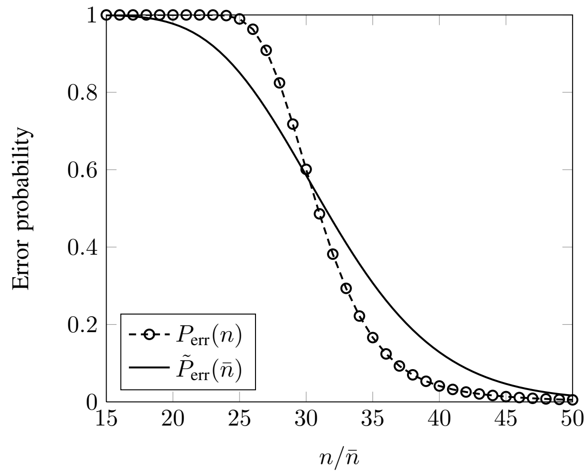
where the change of the order of the integral and the infinite sum follows from the monotone convergence theorem and the second last step follows because the integral is the Gamma function of order $n+1$ and is equal to $n!$. ■

Compared with the formulae for $\mathbb{E}[N_{\mathrm{BP}^*}]$ in (14) in the form of summation, the formula here is in the form of an integration. When $\tilde{P}_{\text{err}}(\bar{n})$ is easier to obtain than $P_{\text{err}}(n)$, the new formula may have certain advantage for numerical evaluation.

Checking the proof of the above theorem, we see that the equivalence of these two formulae depends only on the properties of the Poisson distribution, but not on the underlaying distribution of N_{BP^*}. In general, let b_n be an



(a) n from 1 to 200



(b) Zoom-in for n from 15 to 50

Fig. 3. Comparison of $\tilde{P}_{\text{err}}$ and P_{err} for degree distribution $\mathbf{\Psi}^{\text{BP}}$. Here $K = 256$, $q = 256$ and the rank distribution is given in Table I.

infinite sequence such that $b_n \geq 0$ and $\sum_{n=0}^{\infty} b_n$ exists. Define $\tilde{b}(x) = \sum_n \frac{x^n e^{-x}}{n!} b_n$. Then we have

$$\int_0^{\infty} \tilde{b}(x)\mathrm{d}x = \int_0^{\infty} \sum_n \frac{x^n e^{-x}}{n!} b_n \mathrm{d}x = \sum_n \frac{b_n}{n!} \int_0^{\infty} x^n e^{-x} \mathrm{d}x = \sum_n \frac{b_n}{n!} n! = \sum_n b_n.$$

E. Evaluation Example II

Following the example in Section III-F, we evaluate $\tilde{P}_{\text{err}}(\bar{n})$ for the degree distribution $\mathbf{\Psi}^{\text{BP}}$ given in Table III in Appendix VI and compare it with $P_{\text{err}}(n)$. From the illustration in Fig. 3, we first observe that the two curves are similar except for the different decrease rates. $\tilde{P}_{\text{err}}(\bar{n})$ decreases slightly slower than $P_{\text{err}}(n)$ which matches our characterization that

$$\lim_{n\to\infty} \frac{-\log(P_{\text{err}}(n))}{n} = -\log q^* \geq 1 - q^* = \lim_{\bar{n}\to\infty} \frac{-\log \tilde{P}_{\text{err}}(\bar{n})}{\bar{n}}.$$

Further, from the two formulae for $\mathbb{E}[N_{\text{BP}^*}]$ in terms of $P_{\text{err}}(n)$ and $\tilde{P}_{\text{err}}(\bar{n})$ respectively, we know that the areas below the two curves in Fig. 3 are roughly the same.

V. Analysis of Inactivation Decoding

In this section, we study *inactivation decoding*, which can reduce the coding overhead for relatively small K compared with BP decoding.

A. Introduction of Inactivation Decoding

Inactivation decoding was proposed for LT/Raptor codes [17], [18] and can be regarded as an efficient way to solve sparse linear systems [24], [25], and a similar algorithm [26] has been used for efficient encoding of LDPC codes. Here we describe how to use inactivation for BATS codes.

In the BP decoding algorithm discussed in Section II-C, the decoding stops when no decodable input symbols remain. Though BP decoding stops, Gaussian elimination can still be used to decode the remaining input symbols (by combining the linear systems associated with the undecoded batches to a single linear system involving all the undecoded input symbols). But the decoding complexity using Gaussian elimination is much higher than that of BP decoding. Inactivation decoding combines BP decoding with Gaussian elimination in a more efficient way.

We describe an inactivation decoding process for a given number n of batches, denoted by $\mathrm{INAC}(n)$. The decoding of $\mathrm{INAC}(n)$ is the same $\mathrm{BP}(n)$ until there are no decodable symbols. Instead of stopping the decoding as in $\mathrm{BP}(n)$, $\mathrm{INAC}(n)$ tries to resume the BP decoding process by "inactivating" certain undecoded input symbols. Specifically, suppose that there are no decodable input symbols at time t, $\mathrm{INAC}(n)$ randomly picks an undecoded symbol b and marks it as *inactive*. The decoder substitutes the inactive symbol b into the batches like a decoded symbol, except that b is an indeterminate, and increases the time by one. For example, if the k-th input symbol b_k is inactivated at time t and $k \in A_i^{(t)}$, each component of $\mathbf{Y}_i^{(t+1)} = \mathbf{Y}_i^{(t)} - b_k g \mathbf{H}_i$ will be expressed as a linear polynomial of b_k. Since the time is increased by one for each input symbol decoded or inactivated, the decoding process of $\mathrm{INAC}(n)$ is repeated until time K when all the input symbols are either decoded or inactive.

Denote by I the number of inactive symbols after $\mathrm{INAC}(n)$ stops, and denote by $b_1, \ldots, b_I$ the inactive input symbols. A decoded input symbol b now can be expressed as

$$b = \sum_{i=1}^{I} \alpha_i b_i + \alpha_0,$$

where α_i $(0 \leq i \leq I)$ are determined by the decoding process. Therefore, the inactivation decoding recovers a linear formula of each decoded input symbol in terms of the inactive symbols.

After $\mathrm{INAC}(n)$ stops, we need to recover the inactive symbols and substitute their values into the formulae of the decoded input symbols. To generat $K - I$ decoded input symbols, the decoder consumes $K - I$ of all the received symbols. The other received symbols are actually transformed into linear equations of the inactive symbols, and then used to solve the inactive symbols. For example, if all the input symbols of a batch is decoded (in terms of the inative symbols), the received symbols of this batch cannot be used to decode more input symbols, but they impose linear constraints on the inative symbols. Usually, this linear system of inactive symbols are solved by Gaussian elimination.

The inactive symbols are uniquely solvable if and only if the (global) linear system formed by the linear systems associated with all the batches is uniquely solvable. When being used with the precoding techniques of *high-density parity-check* and *permanent inactivation*, the decoding of the inactive symbols can be successful with high probability for a small coding overhead. Readers may find the detailed discussion of these precoding techniques in [18]. Our analysis to be provided is not associated with any specific precoding technique.

Inactivation decoding incurs extra computation cost that includes solving the inactive symbols using Gaussian elimination and substituting the values of the inactive symbols. Since both terms depend on the number of inactive symbols, knowing this number can help us to understand the tradeoff between computation cost and coding rate. In the remainder of this section, we provide methods to compute the expected number of inactive symbols.

B. Expected Number of Inactivation

Since the inactive input symbols are treated as decoded during the inactive decoding, the decodability of batches can be defined the same as for BP decoding. Let $\hat{R}_n^{(t)}$ and $\hat{C}_n^{(t)}$ be the number of decodable input symbols and the number of undecodable batches, respectively, at time t when using $\mathrm{INAC}(n)$. From the description of inactivation decoding, the probability that a symbol is inactivated at time $t < K$ is

$$P_{\text{inac}}(t|n) \triangleq \Pr\{\hat{R}_n^{(t)} = 0\}.$$

At time K, the decoding stops (all the input symbols are either decoded or inactive). The expectation of the number of inactive symbols can be expressed as

$$\mathbb{E}[I|n] = \sum_{t=0}^{K-1} P_{\text{inac}}(t|n).$$

Define an $(n+1) \times (K-t+1)$ matrix $\mathbf{\Gamma}_n^{(t)}$ as

$$\mathbf{\Gamma}_n^{(t)}[c,r] \triangleq \Pr\left\{\hat{C}_n^{(t)} = c, \hat{R}_n^{(t)} = r\right\}.$$

According to the definition, we can write

$$P_{\text{inac}}(t|n) = \sum_{c=0}^{n} \mathbf{\Gamma}_n^{(t)}[c,0]. \tag{21}$$

Define $\mathbf{N}_t$ as a $(K-t+2) \times (K-t+1)$ matrix of the form $\begin{bmatrix} \mathbf{e}_0 \\ \mathbf{I} \end{bmatrix}$, so that

$$\mathbf{\Gamma}_n^{(t-1)}[c,:]\mathbf{N}_t = (\mathbf{\Gamma}_n^{(t-1)}[c,0] + \mathbf{\Gamma}_n^{(t-1)}[c,1], \mathbf{\Gamma}_n^{(t-1)}[c,2:K-t+1]).$$

The following theorem provides an iterative formula for $\mathbf{\Gamma}_n^{(t)}$, $t = 0, 1, \ldots, K$.

Theorem 11. *Consider a BATS code with K input symbols, n batches, degree distribution $\mathbf{\Psi}$, rank distribution $\mathbf{h}$ of the transfer matrix, and batch size M. We have for inactivation decoding*

$$\mathbf{\Gamma}_n^{(0)}[c,:] = \mathrm{Bi}(c; n, 1-\rho_0)\mathbf{e}_0\mathbf{Q}_0^{n-c}, \tag{22}$$

and for $t > 0$,

$$\mathbf{\Gamma}_n^{(t)}[c,:] = \sum_{c'=c}^{n} \mathrm{Bi}(c; c', 1-\rho_t)\mathbf{\Gamma}_n^{(t-1)}[c',:]\mathbf{N}_t\mathbf{Q}_t^{c'-c}. \tag{23}$$

Proof: The proof is similar to that of Theorem 1. See Appendix V. ∎

If we replace $\mathbf{N}_t$ by $\begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix}$ of proper dimension, the above theorem becomes Theorem 1. Due to this similarity, many discussions about BP decoding based on Theorem 1 apply to inactivation decoding as well. For example, the following formula is simpler for evaluating $P_{\text{inac}}(t|n)$ of a range of n.

Theorem 12. *For $n \geq 0$ and $t \geq 0$,*

$$P_{\text{inac}}(t|n) = \sum_{c=0}^{n} \binom{n}{c} \left(1 - \sum_{\tau=0}^{t} p_\tau\right)^c \mathbf{\Gamma}_{n-c}^{(t)}[0,0], \tag{24}$$

where the first row of the matrices $\mathbf{\Gamma}_{n'}^{(t)}$, $n' = 0, 1, \ldots, n$ can be computed by the following recursion: For $n' = 0, 1, \ldots, n$,

$$\mathbf{\Gamma}_{n'}^{(0)}[0,:] = (p_0 \mathbf{Q}_0)^{n'}[0,:], \tag{25}$$

and for $t > 0$

$$\mathbf{\Gamma}_{n'}^{(t)}[0,:] = \sum_{c=0}^{n'} \binom{n'}{c} \mathbf{\Gamma}_{n'-c}^{(t-1)}[0,:] \mathbf{N}_t (p_t \mathbf{Q}_t)^c. \tag{26}$$

Proof: The proof is similar to that of Theorem 2. See Appendix V. ∎

The formula in the above theorem can be evaluated similarly as the one in Theorem 2. Similar to $P_{\text{stop}}(t|n)$, $P_{\text{inac}}(t|n)$ can also be expressed as the linear combination of 2^t n-th powers.

Theorem 13. *For $n \geq 0$ and $t \geq 0$,*

$$P_{\text{inac}}(t|n) = \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0] \left(1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0]\right)^n,$$

where matrix $\mathbf{\Delta}_{t,i}$ is defined in Theorem 3, and row vector $\mathbf{V}'_{t,i}$ is defined as follows:

i) $\mathbf{V}'_{0,0} \triangleq \mathbf{U}_0[0,:]$,

ii) *For $t \geq 0$ and $i = 0, 1, \ldots, 2^t - 1$,*

$$\mathbf{V}'_{t+1,i} = \mathbf{V}'_{t,i}[1:],$$
$$\mathbf{V}'_{t+1,2^t+i} = \mathbf{V}'_{t,i}[0](\mathbf{U}_{t+1}[0,:] - \mathbf{U}_t[0,1:]).$$

Proof: The proof is similar to that of Theorem 3. See Appendix V. ∎

Recalling that $q_t = 1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,0}[0,0]$ (see (11)) and the definition of r_{BP} following Proposition 1. Applying Theorem 13, we can further obtain the following asymptotic behavior of $P_{\text{inac}}(t|n)$ when n is large.

Theorem 14. *When $t < r_{\text{BP}}$, $P_{\text{inac}}(t|n) = q_t^n$, and when $t \geq r_{\text{BP}}$,*

$$\lim_{n\to\infty} \frac{-\log P_{\text{inac}}(t|n)}{n} = -\log q_t.$$

Proof: See Appendix V. ∎

Corollary 15.

$$\lim_{n\to\infty} \frac{-\log \mathbb{E}[I|n]}{n} = -\log q^*.$$

where $q^* = \vee_{t=0}^{K-1} q_t$.

C. Poisson Number of Batches

In this subsection, we assume that the number of received batches is a Poisson distributed random variable $\tilde{N}$ with mean $\bar{n}$. Denote by $\tilde{I}$ the number of inactive symbols after $\text{INAC}(\tilde{N})$ stops.

Define a row vector $\tilde{\mathbf{\Gamma}}_{\bar{n}}^{(t)}$ of size $K-t+1$ as

$$\tilde{\mathbf{\Gamma}}_{\bar{n}}^{(t)}[r] \triangleq \Pr\left\{\hat{R}_{\tilde{N}}^{(t)} = r\right\} = \sum_n \frac{\bar{n}^n}{n!} e^{-\bar{n}} \Pr\left\{\hat{R}_n^{(t)} = r\right\}.$$

Thus,

$$\tilde{\mathbf{\Gamma}}_{\bar{n}}^{(t)} = \sum_n \frac{\bar{n}^n}{n!} e^{-\bar{n}} \sum_{c=0}^{n} \mathbf{\Gamma}_n^{(t)}[c, :]. \tag{27}$$

The probability that an input symbol is inactive at time t is

$$\tilde{P}_{\text{inac}}(t|\bar{n}) = \tilde{\mathbf{\Gamma}}_{\bar{n}}^{(t)}[0] = \sum_n \Pr\{\tilde{N} = n\} P_{\text{inac}}(t|n), \tag{28}$$

and hence the expected number of inactive symbols is given by

$$\mathbb{E}[\tilde{I}|\bar{n}] = \sum_{t=0}^{K-1} \tilde{P}_{\text{inac}}(t|\bar{n}) = \sum_{t=0}^{K-1} \tilde{\mathbf{\Gamma}}_{\bar{n}}^{(t)}[0]. \tag{29}$$

The next theorem provides a formula for calculating $\tilde{\mathbf{\Gamma}}_{\bar{n}}^{(t)}$.

Theorem 16. *Consider inactivation decoding of a BATS code with K input symbols, degree distribution $\mathbf{\Psi}$, and transfer matrix rank distribution $\mathbf{h}$. When the number of batches used by BP decoding is Poisson distributed with expectation $\bar{n}$, for any integer $t \geq 0$*

$$\tilde{\mathbf{\Gamma}}_{\bar{n}}^{(t)} = \tilde{\mathbf{\Gamma}}_{\bar{n}}^{(t-1)} \mathbf{N}_t \exp\left(\bar{n} p_t (\mathbf{Q}_t - \mathbf{I})\right),$$

where $\tilde{\mathbf{\Gamma}}_{\bar{n}}^{-1} \triangleq \mathbf{e}_0$.

Proof: Theorem 16 can be proved similarly as Theorem 7. See Appendix V. ∎

Recall that $q_t = 1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,0}[0,0]$ (see (11)) and the definition of r_{BP} following Proposition 1.

Theorem 17. *When $t < r_{\text{BP}}$, $\tilde{P}_{\text{inac}}(t|n) = \exp(-\bar{n}(1-q_t))$, and when $t \geq r_{\text{BP}}$,*

$$\lim_{n\to\infty} \frac{-\log \tilde{P}_{\text{inac}}(t|\bar{n})}{n} = 1 - q_t.$$

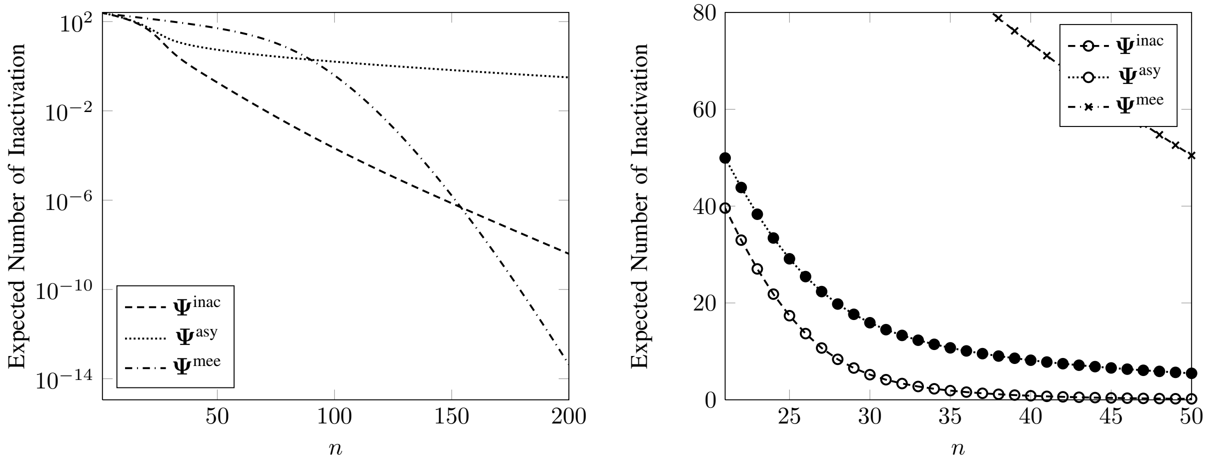
Proof: Using Theorem 13 and (28), we get

$$\tilde{P}_{\text{inac}}(t|\bar{n}) = \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0] \exp\left(-\bar{n}\left(\sum_{\tau=0}^{t} p_\tau - \mathbf{\Delta}_{t,i}[0,0]\right)\right).$$

The remainder of the proof is similar to that of Theorem 14 and can be found in Appendix V. ∎

Corollary 18.

$$\lim_{\bar{n}\to\infty} \frac{-\log \mathbb{E}[\tilde{I}|\bar{n}]}{\bar{n}} = 1 - q^*.$$



(a) n from 1 to 200

(b) Zoom-in with n from 21 to 40

Fig. 4. Expected number of inactivation for different degree distributions. Here $K = 256$, $q = 256$ and the rank distribution is given in Table I.

D. Evaluation Example III

Following the example in Section III-F, we further evaluate the inactivation decoding performance of three degree distributions $\mathbf{\Psi}^{\text{asy}}$, $\mathbf{\Psi}^{\text{inac}}$ and $\mathbf{\Psi}^{\text{mee}}$ given in Table III in Appendix VI, where $\mathbf{\Psi}^{\text{inac}}$ is obtained by modifying $\mathbf{\Psi}^{\text{asy}}$ using an approach to be introduced in Section VI. We evaluate $\mathbb{E}[I|n]$, $n = 1, \ldots, 200$ for the three degree distributions. See Fig. 4 for an illustration of the evaluation results.

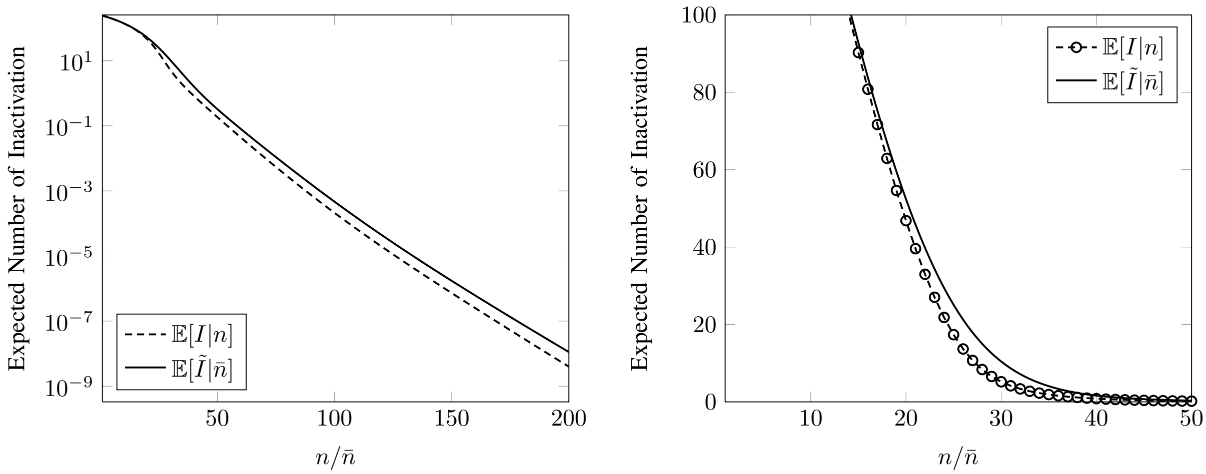
We first observe that for all the degree distributions, the expected number of inactivation decreases exponentially fast when n is large. For $\mathbf{\Psi}^{\text{mee}}$, the asymptotic decrease rate of the expected number of inactivation is the fastest among these three distributions. From Fig. 1(b), we observe that $\mathbf{\Psi}^{\text{inac}}$ has the smallest expected number of inactivation for n from 20 to 50. For example, if we use $n = 25$ for $\mathbf{\Psi}^{\text{inac}}$, the expected number of inactivation is about 17.

We also evaluate $\mathbb{E}[\tilde{I}|\bar{n}]$ and compare it with $\mathbb{E}[I|n]$ for degree distribution $\mathbf{\Psi}^{\text{inac}}$. From the illustration in Fig. 5, we observe that the two curves are similar except for the different decrease rates. $\tilde{P}_{\text{err}}(\bar{n})$ decreases slightly slower than $P_{\text{err}}(n)$ which matches our characterization that

$$\lim_{n\to\infty} \frac{-\log \mathbb{E}[I|n]}{n} = -\log q^* \geq 1 - q^* = \lim_{\bar{n}\to\infty} \frac{-\log \mathbb{E}[\tilde{I}|\bar{n}]}{\bar{n}}.$$

VI. Degree-Distribution Optimization Examples

In this section, we demonstrate how to use the formulae in the previous sections to optimize the degree distribution for finite block lengths. Note that our purpose here is to illustrate the applications of the formulae obtained in this paper, but not to propose an optimization approach for practical use. How to optimize the degree distribution for practical applications is beyond the scope of this paper.



(a) n from 1 to 200

(b) Zoom-in with n from 1 to 50

Fig. 5. Comparison of $\mathbb{E}[I|n]$ and $\mathbb{E}[\tilde{I}|\bar{n}]$ for degree distribution $\mathbf{\Psi}^{\text{inac}}$. Here $K = 256$, $q = 256$ and the rank distribution is given in Table I.

A. General Framework

We want to optimize the degree distribution of BATS codes to minimize the expected coding overhead (for BP decoding) or the expected number of inactivation (for inactivation decoding). A general approach has the following two steps with an initial degree distribution $\mathbf{\Psi}^{(0)}$ (which can be trivial).

i) Find one or multiple new degree distributions which may be potentially better than $\mathbf{\Psi}^{(0)}$.
ii) Evaluate the BP decoding performance of these new degree distributions in terms of an objective function, and select the degree distribution that outperforms $\mathbf{\Psi}^{(0)}$ the most.

These two steps can be applied repeatedly.

The above framework has been used in the design of LT/Raptor codes. For example, in the design of finite-length Raptor codes discussed in [27], the first step is achieved by a heuristic bound on the input ripple size, and the second step is performed by means of the exact calculation of the error probability. In one of the optimizations performed in [28], a robust soliton distribution is sampled at the first step, and a heuristic formula of the expected number of inactivation is evaluated at the second step.

In this paper, we adapt this framework in the following way. We use the degree distribution obtained from the asymptotic analysis of BATS codes as the initial degree distribution $\mathbf{\Psi}^{(0)}$. In the first step, a new degree distribution $\mathbf{\Psi}^{(1)}$ is obtained by perturbing the degree distribution $\mathbf{\Psi}^{(0)}$ at certain degree d so that

$$\mathbf{\Psi}^{(1)} = (\mathbf{\Psi}^{(0)} + \delta \mathbf{e}_{d-1})/(1+\delta),$$

where δ is a real number and $\mathbf{e}_{d-1}$ is the all-zero vector expect that the $(d-1)$-th component is 1. In the second step, we compare the performance of $\mathbf{\Psi}^{(1)}$ and $\mathbf{\Psi}^{(0)}$ based on our finite-length results of BATS codes. If $\mathbf{\Psi}^{(1)}$ is better than $\mathbf{\Psi}^{(0)}$, we replace $\mathbf{\Psi}^{(0)}$ by $\mathbf{\Psi}^{(1)}$. We repeat these two steps for a number of iterations and output $\mathbf{\Psi}^{(0)}$.

B. Optimization for BP Decoding

For BP decoding, we want to have a degree distribution that has a smaller expected coding overhead than $\mathbf{\Psi}^{\text{asy}}$. To compare the two degree distributions in the second step of our optimization framework, we may use (16) to evaluate $\mathbb{E}[N_{\text{BP}^*}]$ which is accurate enough if a large value of n_2 is used. But it is indeed not necessary to accurately evaluate $\mathbb{E}[N_{\text{BP}^*}]$ in the second step. To make the evaluation in the second step fast, we instead use $\tilde{P}_{\text{err}}(\bar{n})$ (with a properly chosen value of $\bar{n}$) as a proxy of $\mathbb{E}[N_{\text{BP}^*}]$.

As hinted by Proposition 2 and observed in numerical evaluations, $P_{\text{err}}(n)$ is very close to 1 when $n < K/\bar{h}$. So we have the approximation that

$$\mathbb{E}[N_{\text{BP}^*}] \gtrapprox n_1 + \sum_{n=n_1}^{n_2} P_{\text{err}}(n),$$

where $n_1 = \lceil K/\sum_i h_i \rceil$ and n_2 is a sufficiently large integer. We only need to pick n_2 such that $\sum_{n=n_2+1}^{\infty} P_{\text{err}}(n)$ is sufficiently small for the desired degree distributions. For other degree distributions such that $\sum_{n=n_2+1}^{\infty} P_{\text{err}}(n)$ is large, the above approximation is roughly a lower bound on the expected coding overhead, which is sufficient for our purpose of comparison. Similarly, we have the approximation

$$\tilde{P}_{\text{err}}(\bar{n}) \gtrapprox \sum_{n=0}^{n_1-1} \frac{\bar{n}^n e^{-\bar{n}}}{n!} + \sum_{n=n_1}^{n_2} \frac{\bar{n}^n e^{-\bar{n}}}{n!} P_{\text{err}}(n).$$

The first terms in both approximations are constants. Since the pmf of the Poisson distribution exhibits relatively small changes for the probability masses around its expectation, we can choose $\bar{n} = (n_1 + n_2)/2$ and expect that $\mathbb{E}[N_{\text{BP}^*}]$ and $\tilde{P}_{\text{err}}(\bar{n})$ share a similar trend when the degree distribution changes.

For the example in Section III-F, $\mathbf{\Psi}^{\text{BP}}$ is the obtained degree distribution, as given in Table III-(b). The comparison of this degree distribution with $\mathbf{\Psi}^{\text{asy}}$ and $\mathbf{\Psi}^{\text{mee}}$ for BP decoding can be found in Fig. 1 and Table II.

C. Optimization for Inactivation Decoding

For inactivation decoding, we want to have a degree distribution that has a smaller expected number of inactivations than $\mathbf{\Psi}^{\text{asy}}$. To compare two degree distributions in the second step of our optimization framework, we compare $\mathbb{E}[\tilde{I}|n]$ instead of $\mathbb{E}[I|n]$ to reduce the evaluation time. For the example in Section III-F, $\mathbf{\Psi}^{\text{inac}}$ is the obtained degree distribution, as given in Table III-(c). A comparison of this degree distribution with $\mathbf{\Psi}^{\text{asy}}$ and $\mathbf{\Psi}^{\text{mee}}$ for inactivation decoding can be found in Fig. 4.

VII. Concluding remarks

Our results in this paper significantly advances the analysis of finite-length BATS/LT codes. The recursive formulae in this paper can easily be evaluated numerically using matrix operations. Without heavy simulation, we can directly calculate the error probability of BP decoding and the expected number of inactive symbols. Based on the examples provided in this paper, it is possible to derise sophisticated finite-length degree distribution optimization methods for various applications of BATS codes. Further research is needed in the analysis of the power-sum formulae towards more explicit finite-length results.

APPENDIX I
PROOF OF THEOREM 1

The subscripts of $R_n^{(t)}$ and $C_n^{(t)}$ are omitted in this proof. Let $\bar{\Theta}_s^{(t)}$ be the set of indices of batches that both the degree and the rank at time t equal to s. In other words, a batch with index in $\bar{\Theta}_s^{(t)}$, $s > 0$, is decodable and can decode s symbols. Let $\Theta^{(t)}$ be the set of indices of batches that are not in $\bar{\Theta}^{(t)} \triangleq \cup_{s=0}^{M} \bar{\Theta}_s^{(t)}$. We see that $R^{(t)} = |\cup_{i \in \bar{\Theta}^{(t)}} A_i^{(t)}|$, which is valid since $A_i^{(t)} = \emptyset$ for $i \in \bar{\Theta}_0^{(t)}$. Also, we see that $C^{(t)} = |\Theta^{(t)}|$.

A. Initial status

We first calculate $\mathbf{\Lambda}_n^{(0)}[c, r] = \Pr\{C^{(0)} = c, R^{(0)} = r\}$. When $t = 0$, a batch with degree s has the probability Ψ_s and is decodable with probability $\hbar'_s$ (see (2) for the definition of $\hbar'_s$). Therefore, the probability that a batch is in $\bar{\Theta}_s^{(0)}$ is $\Psi_s \hbar'_s$, i.e., for $1 \leq i \leq n$ and $0 \leq s \leq M$,

$$\Pr\left\{i \in \bar{\Theta}_s^{(0)}\right\} = p_{0,s} \triangleq \Psi_s \hbar'_s.$$

Hence,

$$\Pr\left\{i \in \bar{\Theta}^{(0)}\right\} = \sum_{s=0}^{M} p_{0,s} \triangleq \rho_0. \tag{30}$$

Since all batches are independently generated, we have

$$\Pr\left\{C^{(0)} = k\right\} = \Pr\left\{|\Theta^{(0)}| = k\right\} = \mathrm{Bi}(k; n, 1 - \rho_0). \tag{31}$$

When $\rho_0 = 0$, $\Pr\{C^{(0)} = n, R^{(0)} = 0\} = 1$ and the formula in (5) holds. Henceforth in this subsection, we assume $\rho_0 > 0$. Recall $\mathbf{Q}_0$ defined in (7).

Lemma 4. *We have for $k = 0, 1, \ldots, n$,*

$$\left(\Pr\left\{R^{(0)} = j | C^{(0)} = n - k\right\} : j = 0, \ldots, K\right) = \mathbf{e}_0 \mathbf{Q}_0^k,$$

where $\mathbf{e}_0 = (1, 0, \ldots, 0)$.

Proof: Fix n. If $k = 0$, then $\bar{\Theta}^{(0)} = \emptyset$, and hence $\Pr\{R^{(0)} = 0 | C^{(0)} = n\} = 1$, i.e., the lemma with $k = 0$ is proved. Henceforth, we assume $k > 0$. The condition $C^{(0)} = n - k$ means that k batches becomes decodable at time 0. Suppose that $\bar{\Theta}^{(0)} = \{1, \ldots, k\}$, which does not change the distribution of $R^{(0)}$. Define $Z_0 \equiv 0$ as a constant random variable on $\{0, 1, \ldots, K\}$, and for $r = 1, \ldots, k$ define $Z_r = | \cup_{m=1}^{r} A_m|$. These random variables are defined under the condition $\{\bar{\Theta}^{(0)} = \{1, \ldots, k\}\} \triangleq E$. Note that $Z_k = R^{(0)}$. Since the contributors of each batch are independently chosen, $Z_0, \ldots, Z_k$ forms a Markov chain. Specifically, for $j < i$, $\Pr\{Z_r = j | Z_{r-1} = i\} = 0$

and for $j \geq i$,

$$\begin{aligned}\Pr\{Z_r = j | Z_{r-1} = i\} &= \Pr\left\{ |\cup_{m=1}^{r} A_m| = j \big| |\cup_{m=1}^{r-1} A_m| = j, E \right\} \\ &= \sum_{s=j-i}^{j} \underbrace{\Pr\left\{ |\cup_{m=1}^{r} A_m| = j \big| |\cup_{m=1}^{r-1} A_m| = i, |A_r| = s, E \right\}}_{(a)} \\ &\quad \times \underbrace{\Pr\left\{ |A_r| = s \big| |\cup_{m=1}^{r-1} A_m| = i, E \right\}}_{(b)}.\end{aligned}$$

Term (a) is a hypergeometric distribution $\mathrm{hyge}(s-j+i; K, i, s)$. Term (b) is equal to $\Pr\left\{|A_r| = s \big| r \in \bar{\Theta}^{(0)}\right\} = \frac{p_{0,s}}{\rho_0}$ for $s \leq M$ and zero otherwise. Overall, we have $\Pr\{Z_r = j | Z_{r-1} = i\} = \mathbf{Q}_0[i, j]$, independent of r. Therefore, $Z_0, \ldots, Z_k$ forms a homogeneous Markov chain with transition matrix $\mathbf{Q}_0$. The proof is completed by noting that $\mathbf{e}_0$ is the probability vector corresponding to the distribution of Z_0. ■

By (31) and Lemma 4, we have

$$\begin{aligned}\mathbf{\Lambda}_n^{(0)}[c, :] &= (\Pr\{C^{(0)} = c, R^{(0)} = j\} : j = 0, \ldots, K) \\ &= \Pr\{C^{(0)} = c\}(\Pr\{R^{(0)} = j | C^{(0)} = c\} : j = 0, \ldots, K) \\ &= \mathrm{Bi}(c; n, 1-\rho_0)\mathbf{e}_0 \mathbf{Q}_0^{n-c},\end{aligned}$$

which proves (5).

B. Recursive formula

Consider $t > 0$ and we prove the recursion of $\mathbf{\Lambda}_n^{(t)}$ in (6). Define event E_t as $\{R^{(\tau)} > 0, \tau < t\}$, i.e.,

$$E_t = \left\{ \cup_{s=1}^{M} \bar{\Theta}_s^{(\tau)} \neq \emptyset, \tau < t \right\}.$$

We have for $t > 0$

$$\begin{aligned}\mathbf{\Lambda}_n^{(t)}[c, r] &= \Pr\left\{ C^{(t)} = c, R^{(t)} = r, R^{(\tau)} > 0, \tau < t \right\} \\ &= \sum_{c', r' > 0} \Pr\left\{ C^{(t)} = c, C^{(t-1)} = c', R^{(t)} = r, R^{(t-1)} = r', R^{(\tau)} > 0, \tau < t \right\} \\ &= \sum_{c', r' > 0} \Pr\left\{ C^{(t)} = c, R^{(t)} = r, | C^{(t-1)} = c', R^{(t-1)} = r', R^{(\tau)} > 0, \tau < t-1 \right\} \mathbf{\Lambda}_n^{(t-1)}[c', r']. \\ &= \sum_{c', r' > 0} \underbrace{\Pr\left\{ R^{(t)} = r | C^{(t)} = c, C^{(t-1)} = c', R^{(t-1)} = r', E_{t-1} \right\}}_{(c)} \times \\ &\quad \times \underbrace{\Pr\left\{ C^{(t)} = c | C^{(t-1)} = c', R^{(t-1)} = r', E_{t-1} \right\}}_{(d)} \mathbf{\Lambda}_n^{(t-1)}[c', r'].\end{aligned}$$

We characterize (c) and (d) in the above equation respectively. Recall that for $t \geq 1$

$$\begin{aligned}p_{t,s} &\triangleq \hbar_s \sum_{d=s+1}^{D} \Psi_d \frac{d}{K} \mathrm{hyge}(d-s-1; K-1, d-1, t-1), \\ \rho_t &\triangleq \frac{\sum_s p_{t,s}}{1 - \sum_{\tau=0}^{t-1} \sum_s p_{\tau,s}}.\end{aligned}$$

Lemma 5. *For $r' > 0$ and $c' \geq c$,*

$$\Pr\left\{C^{(t)} = c | C^{(t-1)} = c', R^{(t-1)} = r', E_{t-1}\right\} = \mathrm{Bi}(c; c', 1-\rho_t).$$

Proof: Under the condition of $R^{(t-1)} = r' > 0$ and E_{t-1}, the BP decoding does not stop at time $t-1$. Note that if $c' = 0$, i.e., all the batches are decodable at time $t-1$, then $C^{(t)} = 0$ with probability one. We henceforth assume $c' > 0$ in this proof. Since $\Theta^{(0)} \supset \Theta^{(1)} \supset \cdots \supset \Theta^{(t-1)}$, we have $C^{(\tau)} > 0$ for $\tau = 0, 1, \ldots, t-1$. We consider a special instance of the condition $C^{(t-1)} = c'$, $R^{(t-1)} = r'$ and E_{t-1} such that the input symbol decoded from time $\tau - 1$ to τ has index $\tau - 1$ for $1 \leq \tau \leq t$, and study the probability of $j \in \bar{\Theta}_s^{(\tau)} \cap \Theta^{(\tau-1)}$ under this instance. Since the probability to be obtain does not depend on the instance, the probability is equal to the probability of the lemma. To simplify the notation, the condition $C^{(t-1)} = c'$, $R^{(t-1)} = r'$ and E_{t-1} is omitted in the remainder of the proof.

For $\tau = 1, \ldots, t$, we first study $\Pr\{j \in \bar{\Theta}_s^{(\tau)} \cap \Theta^{(\tau-1)}\}$ for an arbitrary batch j. There are totally τ input symbols decoded at time τ, where $\tau - 1$ is the index of the input symbol decoded at the step from $\tau - 1$ to τ. Given the initial degree of batch j being d, $j \in \bar{\Theta}_s^{(\tau)} \cap \Theta^{(\tau-1)}$ is equivalent to

1) $\tau - 1 \in A_j$,
2) $|A_j^{(\tau)}| = s$, and
3) $\mathrm{rk}(\mathbf{G}_j^{(\tau-1)}\mathbf{H}_j) = \mathrm{rk}(\mathbf{G}_j^{(\tau)}\mathbf{H}_j) = s$.

Since all batches are formed independently, we know that 1) holds with probability d/K; given 1) the probability that 2) holds is the hypergeometric distribution $\mathrm{hyge}(d-s-1; K-1, \tau-1, d-1)$; given both 1) and 2) the probability that 3) holds is $\hbar_s$ (see (1)). Therefore, the probability for 1), 2) and 3) to hold given $|A_j| = d$ is

$$\frac{d}{K}\hbar_s \mathrm{hyge}(d-s-1; K-1, \tau-1, d-1).$$

Hence, after considering the distribution of the degree,

$$\Pr\left\{j \in \bar{\Theta}_s^{(\tau)} \cap \Theta^{(\tau-1)}\right\} = p_{\tau,s}. \tag{32}$$

Now we study $\Pr\{j \in \Theta^{(\tau)}\}$. Since $\Theta^{(\tau)}$, $\bar{\Theta}_s^{(\tau)} \cap \Theta^{(\tau-1)}$, $s = 0, 1, \ldots, M$ forms a partition of $\Theta^{(\tau-1)}$,

$$\begin{aligned}\Pr\left\{j \in \Theta^{(\tau-1)}\right\} &= \Pr\left\{j \in \Theta^{(\tau)}\right\} + \sum_{s=0}^{M} \Pr\left\{j \in \bar{\Theta}_s^{(\tau)} \cap \Theta^{(\tau-1)}\right\} \\ &= \Pr\left\{j \in \Theta^{(\tau)}\right\} + \sum_{s=0}^{M} p_{\tau,s}.\end{aligned}$$

Using $\Pr\{j \in \Theta^{(0)}\} = 1 - \sum_{s=0}^{M} p_{0,s}$ (see (30)), we obtain that

$$\Pr\left\{j \in \Theta^{(\tau)}\right\} = 1 - \sum_{\tau'=0}^{\tau}\sum_{s=0}^{M} p_{\tau',s}.$$

Hence we have

$$\Pr\left\{j \in \bar{\Theta}^{(t)} | j \in \Theta^{(t-1)}\right\} = \frac{\Pr\{j \in \bar{\Theta}^{(t)} \cap \Theta^{(t-1)}\}}{\Pr\{j \in \Theta^{(t-1)}\}} = \rho_t. \tag{33}$$

In other words, for a batch in $\Theta^{(t-1)}$, it would stay in $\Theta^{(t)}$ with probability $1-\rho_t$. Since batches in $\Theta^{(t-1)}$ stay in $\Theta^{(t)}$ independently, for $B \subset \{1,\dots,n\}$ with $|B| = c'$,

$$\begin{aligned}\Pr\left\{C^{(t)} = c | \Theta^{(t-1)} = B, R^{(t-1)} = r', E_{t-1}\right\} &= \Pr\left\{|\Theta^{(t)}| = c | \Theta^{(t-1)} = B, R^{(t-1)} = r', E_{t-1}\right\} \\ &= \mathrm{Bi}(c; c', 1-\rho_t).\end{aligned}$$

Since the above distribution depends on B only through its cardinality, we have

$$\begin{aligned}&\Pr\left\{C^{(t)} = c | C^{(t-1)} = c', R^{(t-1)} = r', E_{t-1}\right\} \\ &= \sum_{B\subset\{1,\dots,n\}:|B|=c'} \Pr\{C^{(t)} = c|\Theta^{(t-1)} = B, R^{(t-1)} = r', E_{t-1}\} \Pr\{\Theta^{(t-1)} = B | C^{(t-1)} = c', R^{(t-1)} = r', E_{t-1}\} \\ &= \mathrm{Bi}(c; c', 1-\rho_t).\end{aligned}$$

The proof of the lemma is completed. ■

Assume $\sum_{s=0}^{M} p_{t,s} > 0$, which holds when BP decoding can start (see Lemma 1).

Lemma 6. *For $r' > 0$ and $c' \geq c$,*

$$\Pr\left\{R^{(t)} = r | C^{(t)} = c, C^{(t-1)} = c', R^{(t-1)} = r', E_{t-1}\right\} = (\mathbf{Q}_t^{c'-c})[r'-1, r].$$

Proof: First, if $c = c'$, then $\mathbf{Q}_t^{c'-c}$ is the identity matrix, and no batches become decodable for the first time at time t. Therefore, $R^{(t)} = R^{(t-1)} - 1$, which proves the lemma with $c = c'$. Henceforth, we assume $c' > c$. Consider an instance of $\{C^{(t)} = c, C^{(t-1)} = c', R^{(t-1)} = r', E_{t-1}\}$ with $\Theta^{(t-1)} \setminus \Theta^{(t)} = \{1,\dots,c'-c\}$. We will compute the distribution of $R^{(t)}$ by assuming this instance. Since the distribution we will obtain only depends on the instance through c, c' and r', the distribution of $R^{(t)}$ under the condition $\{C^{(t)} = c, C^{(t-1)} = c', R^{(t-1)} = r', E_{t-1}\}$ is the same.

Let $\mathcal{A}$ be the set of indices of decodable input symbols at time $t-1$, excluding the input symbol decoded from time $t-1$ to t. We have $|\mathcal{A}| = r'-1$, which is valid since $r' > 0$. Since batches with index in $B' \setminus B$ become decodable only starting at time t, we have $R^{(t)} = |\mathcal{A} \cup (\cup_{i=1}^{\delta} A_i^{(t)})|$. We use a similar method as in Lemma 4 to compute the distribution of $R^{(t)}$. Define $Z_0 \equiv |\mathcal{A}|$ as a constant random variable on $\{0,1,\dots,K-t\}$, and for $r = 1,\dots,c'-c$ define $Z_r = |\mathcal{A} \cup_{m=1}^{r} A_m|$. Note that $Z_{c'-c} = R^{(t)}$. Since the contributors of each batch are independently chosen, $Z_0,\dots,Z_{c'-c}$ forms a Markov chain. Specifically, for $j < i$, $\Pr\{Z_r = j | Z_{r-1} = i\} = 0$ and for $j \geq i$,

$$\begin{aligned}\Pr\{Z_r = j|Z_{r-1} = i\} &= \Pr\left\{|\mathcal{A} \cup (\cup_{m=1}^{r} A_m^{(t)})| = j \big| |\mathcal{A} \cup (\cup_{m=1}^{r-1} A_m^{(t)})| = i\right\} \\ &= \sum_{s=j-i}^{j} \underbrace{\Pr\left\{|A_r^{(t)}| = s \big| |\mathcal{A} \cup (\cup_{m=1}^{r-1} A_m^{(t)})| = i\right\}}_{(e)} \\ &\quad \times \underbrace{\Pr\left\{|\mathcal{A} \cup (\cup_{m=1}^{r} A_m^{(t)})| = j \big| |\mathcal{A} \cup (\cup_{m=1}^{r-1} A_m^{(t)})| = i, |A_r^{(t)}| = s\right\}}_{(f)}.\end{aligned}$$

Term (e) is equal to $\Pr\{r \in \bar{\Theta}_s^{(t)} | r \in \Theta^{(t-1)} \cap \bar{\Theta}^{(t)}\} = \frac{p_{t,s}}{\sum_s p_{t,s}}$ (see (32)) for $s \leq M$. Term (f) is a hypergeometric distribution $\mathrm{hyge}(s-j+i; K-t, i, s)$. Overall, we have $\Pr\{Z_r = j | Z_{r-1} = i\} = \mathbf{Q}_t[i,j]$, independent of r. Therefore, $Z_0, \ldots, Z_{c'-c}$ forms a homogeneous Markov chain with transition matrix $\mathbf{Q}_t$. The proof is completed by considering the transition matrix from Z_0 to $Z_{c'-c}$. ■

Now we are ready to complete the proof of Theorem 1. With the above two lemmas, we can write

$$\begin{aligned}
\boldsymbol{\Lambda}_n^{(t)}[c,:] &= \sum_{c',r'>0} (\mathbf{Q}_t^{c'-c})[r'-1,:]\mathrm{Bi}(c;c',1-\rho_t)\boldsymbol{\Lambda}_n^{(t-1)}[c',r'] \\
&= \sum_{c'\geq c} \mathrm{Bi}(c;c',1-\rho_t)\boldsymbol{\Lambda}_n^{(t-1)}[c',1{:}]\mathbf{Q}_t^{c'-c}.
\end{aligned}$$

This completes the proof of Theorem 1.

APPENDIX II
PROOFS OF SEVERAL PROPERTIES

Proof of Lemma 2: The first claim can be proved by induction over t. First $1-\rho_0 = 1-p_0$ by definition. Suppose that 1) holds for certain $t \geq 0$. We have $\prod_{\tau=0}^{t+1}(1-\rho_\tau) = (1-\rho_{t+1})(1-\sum_{\tau=0}^{t} p_\tau) = 1-\sum_{\tau=0}^{t+1} p_\tau$, where the first equality follows by the induction hypothesis and the second equality follows the definition of ρ_t. To prove the second claim, we have $\rho_t \prod_{\tau=0}^{t-1}(1-\rho_\tau) = \rho_t(1-\sum_{\tau=0}^{t-1} p_\tau) = p_t$, where the first equality follows by 1) and the second equality follows the definition of ρ_t ■

Proof of Lemma 3: We first prove the formula of $\mathbf{U}_t^{-1}$. Let $\mathbf{U}_t'$ be an upper-triangular matrix with $\mathbf{U}_t'[i,j] = (-1)^{j-i}\binom{K-t-i}{j-i}$ for $i \leq j$. We check that $\mathbf{U}_t\mathbf{U}_t' = \mathbf{I}$. We write

$$(\mathbf{U}_t\mathbf{U}_t')[i,j] = \sum_{k=i}^{j} \mathbf{U}_t[i,k]\mathbf{U}_t'[k,j]. \tag{34}$$

When $i=j$, it is clear that $(\mathbf{U}_t\mathbf{U}_t')[i,i] = 1$. Since $\mathbf{U}_t\mathbf{U}_t'$ is upper triangular, we verify that $(\mathbf{U}_t\mathbf{U}_t')[i,j] = 0$ for $j > i$. Expanding the RHS of (34), we get

$$\begin{aligned}
(\mathbf{U}_t\mathbf{U}_t')[i,j] &= \sum_{k=i}^{j} \binom{K-t-i}{k-i}(-1)^{j-k}\binom{K-t-k}{j-k} \\
&= \binom{K-t-i}{j-i}\sum_{k=i}^{j}(-1)^{j-k}\binom{j-i}{k-i} \\
&= \binom{K-t-i}{j-i}\sum_{k=0}^{j-i}(-1)^{j-i-k}\binom{j-i}{k} \\
&= 0.
\end{aligned}$$

Therefore, $\mathbf{U}_t^{-1} = \mathbf{U}_t'$.

To complete the proof, we need to verify the equality $\mathbf{Q}_t = \mathbf{U}_t\mathbf{D}_t\mathbf{U}_t^{-1}$. Write

$$\begin{aligned}
(\mathbf{U}_t\mathbf{D}_t\mathbf{U}_t^{-1})[i,j] &= \sum_{k=i}^{j} \binom{K-t-i}{k-i}\mathbf{Q}_t[k,k](-1)^{j-k}\binom{K-t-k}{j-k} \\
&= \binom{K-t-i}{j-i}\sum_{k=i}^{j}(-1)^{j-k}\mathbf{Q}_t[k,k]\binom{j-i}{k-i}.
\end{aligned}$$

When $i = j$, it is clear that $(\mathbf{U}_t\mathbf{D}_t\mathbf{U}_t^{-1})[i,i] = \mathbf{Q}_t[i,i]$. Since $\mathbf{U}_t\mathbf{D}_t\mathbf{U}_t^{-1}$ is upper triangular, we consider $j > i$ henceforth. By the definition of $\mathbf{Q}_t[k,k]$, we have

$$\begin{aligned}\sum_{k=i}^{j}(-1)^{j-k}\mathbf{Q}_t[k,k]\binom{j-i}{k-i} &= \sum_{k=i}^{j}(-1)^{j-k}\binom{j-i}{k-i}\sum_{s=0}^{k\wedge M}\frac{p_{t,s}}{p_t}\frac{\binom{k}{s}}{\binom{K-t}{s}} \\ &= \sum_{s=0}^{j\wedge M}\frac{p_{t,s}}{p_t\binom{K-t}{s}}\sum_{k=i\vee s}^{j}(-1)^{j-k}\binom{j-i}{k-i}\binom{k}{s}.\end{aligned}$$

In the following, we show that

$$\sum_{k=i\vee s}^{j}(-1)^{j-k}\binom{j-i}{k-i}\binom{k}{s} = \begin{cases}\binom{i}{s-j+i} & j-i \le s \le j, \\ 0 & s < j-i,\end{cases} \tag{35}$$

which completes the proof that $(\mathbf{U}_t\mathbf{D}_t\mathbf{U}_t^{-1})[i,j] = \mathbf{Q}_t[i,j]$.

The proof of (35) using binomial coefficients with negative integers. We write

$$\begin{aligned}\sum_{k=i\vee s}^{j}(-1)^{j-k}\binom{j-i}{k-i}\binom{k}{s} &= \sum_{k=i\vee s}^{j}(-1)^{j-k}\binom{j-i}{j-k}\binom{k}{k-s} \\ &= \sum_{k=i\vee s}^{j}(-1)^{j-k}\binom{j-i}{j-k}(-1)^{k-s}\binom{-s-1}{k-s} \\ &= (-1)^{j-s}\sum_{k=i\vee s}^{j}\binom{j-i}{j-k}\binom{-s-1}{k-s} \\ &= (-1)^{j-s}\binom{j-i-s-1}{j-s},\end{aligned}$$

where the last equality is obtained by Vandermonde's identity by considering the two cases $i < s$ and $i \geq s$. Note that when $s < j-i$, $\binom{j-i-s-1}{j-s} = 0$. Otherwise,

$$(-1)^{j-s}\binom{j-i-s-1}{j-s} = \binom{i}{j-s}.$$

The proof of the lemma is completed. ■

Appendix III

Proofs about Stopping Time Distribution

Proof of Theorem 2: We will show that for $1 \le c \le n$ and $t \ge 0$,

$$\mathbf{\Lambda}_n^{(t)}[c,:] = \frac{n}{c}\prod_{i=0}^{t}(1-\rho_i)\mathbf{\Lambda}_{n-1}^{(t)}[c-1,:]. \tag{36}$$

By expanding the above recursive formula, we have for $c \ge 0$ and $t \ge 0$,

$$\mathbf{\Lambda}_n^{(t)}[c,:] = \binom{n}{c}\prod_{i=0}^{t}(1-\rho_i)^c\mathbf{\Lambda}_{n-c}^{(t)}[0,:]. \tag{37}$$

Substituting (37) into (4) and by Lemma 2, we get

$$P_{\text{stop}}(t|n) = \sum_{c=0}^{n}\binom{n}{c}\prod_{i=0}^{t}(1-\rho_i)^c\mathbf{\Lambda}_{n-c}^{(t)}[0,0] = \sum_{c=0}^{n}\binom{n}{c}\left(1-\sum_{\tau=0}^{t}p_\tau\right)^c\mathbf{\Lambda}_{n-c}^{(t)}[0,0],$$

proving (8). Further, (9) is obtained by (5) for $c = 0$. To prove (10), we have

$$\begin{aligned}\mathbf{\Lambda}_n^{(t)}[0,:] &= \sum_{c=0}^{n} \rho_t^c \mathbf{\Lambda}_n^{(t-1)}[c, 1{:}]\mathbf{Q}_t^c \\ &= \sum_{c=0}^{n} \binom{n}{c} \rho_t^c \prod_{i=0}^{t-1} (1-\rho_i)^c \mathbf{\Lambda}_{n-c}^{(t-1)}[0, 1{:}]\mathbf{Q}_t^c \\ &= \sum_{c=0}^{n} \binom{n}{c} p_t^c \mathbf{\Lambda}_{n-c}^{(t-1)}[0, 1{:}]\mathbf{Q}_t^c\end{aligned}$$

where the first equality follows from (6) with $c = 0$, the second equality is obtained by substituting (37), and the last step is obtained by applying Lemma 2.

Now we prove (36) by induction. When $t = 0$, we have by Theorem 1 that

$$\begin{aligned}\mathbf{\Lambda}_n^{(0)}[c,:] &= \mathrm{Bi}(c; n, 1-\rho_0)\mathbf{Q}_0^{n-c}[0,:] \\ &= \frac{n}{c}(1-\rho_0)\mathrm{Bi}(c-1; n-1, 1-\rho_0)\mathbf{Q}_0^{(n-1)-(c-1)}[0,:] \\ &= \frac{n}{c}(1-\rho_0)\mathbf{\Lambda}_{n-1}^{(0)}[c-1,:]. \end{aligned} \tag{38}$$

Suppose that (36) holds for $t \geq 0$. Applying the recursive formula of Theorem 1, we can show that

$$\begin{aligned}\mathbf{\Lambda}_{n+1}^{(t)}[c,:] &= \sum_{c'=c}^{n+1} \mathrm{Bi}(c; c', 1-\rho_t)\mathbf{\Lambda}_{n+1}^{(t-1)}[c', 1{:}]\mathbf{Q}_t^{c'-c} \\ &= \sum_{c'=c}^{n+1} \mathrm{Bi}(c; c', 1-\rho_t)\frac{n+1}{c'}\prod_{i=0}^{t-1}(1-\rho_i)\mathbf{\Lambda}_n^{(t-1)}[c'-1, 1{:}]\mathbf{Q}_t^{c'-c} \\ &= \frac{n+1}{c}\prod_{i=0}^{t}(1-\rho_i)\sum_{c'=c}^{n+1} \mathrm{Bi}(c-1; c'-1, 1-\rho_t)\mathbf{\Lambda}_n^{(t-1)}[c'-1, 1{:}]\mathbf{Q}_t^{c'-c} \\ &= \frac{n+1}{c}\prod_{i=0}^{t}(1-\rho_i)\sum_{c''=c-1}^{n} \mathrm{Bi}(c-1; c'', 1-\rho_t)\mathbf{\Lambda}_n^{(t-1)}[c'', 1{:}]\mathbf{Q}_t^{c''-(c-1)} \\ &= \frac{n+1}{c}\prod_{i=0}^{t}(1-\rho_i)\mathbf{\Lambda}_n^{(t)}[c-1,:].\end{aligned}$$

The proof is completed. ■

Proof of Theorem 3: We first show

$$\mathbf{\Lambda}_n^{(t)}[0,:] = \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}\mathbf{\Delta}_{t,i}^n \mathbf{U}_t^{-1} \tag{39}$$

by induction in t. The claim for $t = 0$ can be shown by replacing $p_0\mathbf{Q}_0$ in (9) with the decomposition in Lemma 3. Suppose that the claim of the theorem holds for certain $t \geq 0$. Substituting this form of $\mathbf{\Lambda}_n^t$ into (10) with $t+1$ in place of t, we obtain

$$\begin{aligned}\mathbf{\Lambda}_n^{(t+1)} &= \sum_{c=0}^{n}\binom{n}{c}\sum_{i=0}^{2^t-1}\mathbf{V}_{t,i}\mathbf{\Delta}_{t,i}^n\mathbf{U}_t^{-1}[:, 1{:}](p_{t+1}\mathbf{Q}_{t+1})^c \\ &= \sum_{i=0}^{2^t-1}\mathbf{V}_{t,i}\sum_{c=0}^{n}\binom{n}{c}\mathbf{\Delta}_{t,i}^{n-c}\mathbf{U}_t^{-1}[:, 1{:}]\mathbf{U}_{t+1}(p_{t+1}\mathbf{D}_{t+1})^c\mathbf{U}_{t+1}^{-1}.\end{aligned}$$

Using the same technique as proving (35), we can verify that

$$\mathbf{U}_t^{-1}[:,1:]\mathbf{U}_{t+1} = \begin{bmatrix} -\mathbf{U}_t[0,1:] \\ \mathbf{I} \end{bmatrix} = \begin{bmatrix} -\mathbf{U}_t[0,1:] \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix}. \tag{40}$$

Substituting the above equation into (40), we get

$$\begin{aligned}
\mathbf{\Lambda}_n^{(t+1)} &= \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i} \sum_{c=0}^{n} \binom{n}{c} \mathbf{\Delta}_{t,i}^{n-c} \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} (p_{t+1}\mathbf{D}_{t+1})^c \mathbf{U}_{t+1}^{-1} \\
&\quad + \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i} \sum_{c=0}^{n} \binom{n}{c} \mathbf{\Delta}_{t,i}^{n-c} \begin{bmatrix} -\mathbf{U}_t[0,1:] \\ \mathbf{0} \end{bmatrix} (p_{t+1}\mathbf{D}_{t+1})^c \mathbf{U}_{t+1}^{-1} \\
&= \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[1:] \sum_{c=0}^{n} \binom{n}{c} (\mathbf{\Delta}_{t,i}[1:,1:])^{n-c} (p_{t+1}\mathbf{D}_{t+1})^c \mathbf{U}_{t+1}^{-1} \\
&\quad - \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0]\mathbf{U}_t[0,1:] \sum_{c=0}^{n} \binom{n}{c} (\mathbf{\Delta}_{t,i}[0,0])^{n-c} (p_{t+1}\mathbf{D}_{t+1})^c \mathbf{U}_{t+1}^{-1} && (41) \\
&= \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[1:] \left(\mathbf{\Delta}_{t,i}[1:,1:] + p_{t+1}\mathbf{D}_{t+1}\right)^n \mathbf{U}_{t+1}^{-1} \\
&\quad - \sum_{i=0}^{2^t-1} \mathbf{U}_t[0,1:]\mathbf{V}_{t,i}[0] \left(\mathbf{\Delta}_{t,i}[0,0]\mathbf{I} + p_{t+1}\mathbf{D}_{t+1}\right)^n \mathbf{U}_{t+1}^{-1}, && (42)
\end{aligned}$$

where (41) is obtained by noting $\mathbf{\Delta}_{t,i}$ is diagonal and (42) is obtained by combining the binomial terms. The proof of (39) is completed by checking the definition of $\mathbf{V}_{t+1,i}$ and $\mathbf{\Delta}_{t+1,i}$.

Substituting the formula of $\mathbf{\Lambda}_n^{(t)}[0,:]$ in (39) into (8), we get

$$\begin{aligned}
P_{\text{stop}}(t|n) &= \sum_{c=0}^{n} \binom{n}{c} \left(1 - \sum_{\tau=0}^{t} p_\tau\right)^c \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i} \mathbf{\Delta}_{t,i}^{n-c} \mathbf{U}_t^{-1}[:,0] \\
&= \sum_{c=0}^{n} \binom{n}{c} \left(1 - \sum_{\tau=0}^{t} p_\tau\right)^c \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \mathbf{\Delta}_{t,i}^{n-c}[0,0] \\
&= \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \sum_{c=0}^{n} \binom{n}{c} \left(1 - \sum_{\tau=0}^{t} p_\tau\right)^c (\mathbf{\Delta}_{t,i}[0,0])^{n-c} \\
&= \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \left(1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0]\right)^n,
\end{aligned}$$

where the second equality is obtained using the facts that i) $\mathbf{\Delta}_{t,i}$ is diagonal, ii) $\mathbf{U}_t^{-1}$ is upper-triangular and iii) $\mathbf{U}_t^{-1}[0,0] = 1$. ■

Proof of Theorem 4: For $0 \le t \le K$ and $0 \le i \le K - t$, let

$$\lambda_{t,i} = p_t \mathbf{Q}_t[i,i] = p_t \mathbf{D}_t[i,i] = \sum_{s=0}^{i \wedge M} p_{t,s} \frac{\binom{i}{s}}{\binom{K-t}{s}}, \tag{43}$$

with which we can rewrite

$$q_t = 1 - \sum_{\tau=0}^{t} p_\tau + \sum_{\tau=0}^{t} \lambda_{\tau,t-\tau}.$$

Using Lemma 1 and the definition of $\lambda_{t,j}$, we have that

$$\lambda_{t,j} = 0, \text{ when } 0 \le t < r_{\text{BP}}, t + j < r_{\text{BP}}; \tag{44}$$

$$\lambda_{t,j} > \lambda_{t,j-1}, \text{ when } 0 \le t < r_{\text{BP}}, t + j \ge r_{\text{BP}}; \tag{45}$$

$$0 < \lambda_{t,0} < \lambda_{t,1} < \ldots < \lambda_{t,K-t}, \text{ when } t \ge r_{\text{BP}}. \tag{46}$$

We further show inductively that for $i = 0, 1, \ldots, 2^t - 1$,

$$\mathbf{\Delta}_{t,i}[j,j] = 0, \text{ when } t + j < r_{\text{BP}}, \tag{47}$$

$$\mathbf{\Delta}_{t,i}[j,j] > \mathbf{\Delta}_{t,i}[j-1,j-1], \text{ when } t + j \ge r_{\text{BP}}. \tag{48}$$

By the definition of $\mathbf{\Delta}_{0,0}$ in Theorem 3, we write $\mathbf{\Delta}_{0,0}[j,j] = p_0 \mathbf{D}_0[j,j] = \lambda_{0,j}$, which, together with (44)-(46) with $t = 0$, implies (47) and (48) for $t = 0$. Suppose that (47) and (48) hold for certain $t \ge 0$. By the recursive formula in Theorem 3, we have for $i = 0, 1, \ldots, 2^t - 1$,

$$\mathbf{\Delta}_{t+1,i}[j,j] = \mathbf{\Delta}_{t,i}[j+1,j+1] + p_{t+1}\mathbf{D}_{t+1}[j,j] = \mathbf{\Delta}_{t,i}[j+1,j+1] + \lambda_{t+1,j},$$

$$\mathbf{\Delta}_{t+1,2^t+i}[j,j] = \mathbf{\Delta}_{t,i}[0,0] + p_{t+1}\mathbf{D}_{t+1}[j,j] = \mathbf{\Delta}_{t,i}[0,0] + \lambda_{t+1,j}.$$

When $t+1+j < r_{\text{BP}}$, by the induction hypothesis, we have $\mathbf{\Delta}_{t,i}[j+1,j+1] = 0$ and $\mathbf{\Delta}_{t,i}[0,0] = 0$, and by (44), we have $\lambda_{t+1,j} = 0$. Therefore, $\mathbf{\Delta}_{t+1,i}[j,j] = 0$ and $\mathbf{\Delta}_{t+1,2^t+i}[j,j] = 0$ when $t+1+j < r_{\text{BP}}$, which completes the proof of (47). When $t+1+j \ge r_{\text{BP}}$, by the induction hypothesis, we have $\mathbf{\Delta}_{t,i}[j+1,j+1] > \mathbf{\Delta}_{t,i}[j,j]$, and by (45) or (46), we have $\lambda_{t+1,j} > \lambda_{t+1,j-1}$. Therefore, $\mathbf{\Delta}_{t+1,i}[j,j] > \mathbf{\Delta}_{t+1,i}[j,j]$ and $\mathbf{\Delta}_{t+1,2^t+i}[j,j] > \mathbf{\Delta}_{t+1,2^t+i}[j,j]$ when $t+1+j \ge r_{\text{BP}}$, which completes the proof of (48).

Now we are ready to prove i) and ii) of the theorem. When $t = 0$, by Theorem 3 and $\lambda_{0,0} = 0$, we have $P_{\text{stop}}(0|n) = \mathbf{V}_{0,0}[0]\left(1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{0,0}[0,0]\right)^n = q_0^n$, proving i). When $1 \le t < r_{\text{BP}}$, by Theorem 3 and (47), $P_{\text{stop}}(t|n) = (1 - \sum_{\tau=0}^{t} p_\tau)^n \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0]$. To prove ii), we show that for $t \ge 1$

$$\sum_{i=0}^{2^t-1} \mathbf{V}_{t,i} = \mathbf{0}. \tag{49}$$

When $t = 1$, we have

$$\sum_{i=0}^{1} \mathbf{V}_{1,i} = \mathbf{U}_0[0,1:] - \mathbf{U}_0[0,0]\mathbf{U}_0[0,1:] = \mathbf{0}.$$

Suppose that (49) holds for certain $t \ge 1$. We have

$$\sum_{i=0}^{2^{t+1}-1} \mathbf{V}_{t+1,i} = \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[1:] - \mathbf{U}_t[0,1:] \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] = \mathbf{0}.$$

Before proving iii) of the theorem, we show by induction that for $i = 1, \ldots, 2^t - 1$,

$$\mathbf{\Delta}_{t,0}[j,j] > \mathbf{\Delta}_{t,i}[j,j], \text{ when } t + j \ge r_{\text{BP}}. \tag{50}$$

The above inequality holds trivially for $t=0$. Suppose that (50) holds for certain $t \geq 0$. When $t+1+j \geq r_{\text{BP}}$, we have for $i = 0, 1, \dots, 2^t - 1$,

$$\begin{aligned}
\mathbf{\Delta}_{t+1,0}[j,j] &= \mathbf{\Delta}_{t,0}[j+1,j+1] + p_{t+1}\mathbf{D}_{t+1}[j,j] \\
&\geq \mathbf{\Delta}_{t,i}[j+1,j+1] + p_{t+1}\mathbf{D}_{t+1}[j,j] = \mathbf{\Delta}_{t+1,i}[j,j] \\
&> \mathbf{\Delta}_{t,i}[0,0] + p_{t+1}\mathbf{D}_{t+1}[j,j] = \mathbf{\Delta}_{t+1,2^t+i}[j,j],
\end{aligned}$$

where the first inequality follows by the induction hypothesis with equality only when $i=0$, and the second inequality follows from (47) and (48).

Now, we prove iii) for $t \geq r_{\text{BP}} \geq 1$. By (50), we know that for $i = 1, \dots, 2^t - 1$,

$$\mathbf{\Delta}_{t,0}[0,0] > \mathbf{\Delta}_{t,i}[0,0],$$

and hence

$$\frac{1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0]}{q_t} = \frac{1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0]}{1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,0}[0,0]} < 1. \tag{51}$$

By Theorem 3 and noting that $\mathbf{V}_{t,0}[0] = \mathbf{U}_0[0,t] = \binom{K}{t} > 0$, we write

$$\begin{aligned}
\lim_{n\to\infty} \frac{-\log P_{\text{stop}}(t|n)}{n} &= \lim_{n\to\infty} \frac{-\log q_t^n \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0](1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0])^n / q_t^n}{n} \\
&= -\log q_t + \lim_{n\to\infty} \frac{-\log\left(\mathbf{V}_{t,0}[0] + \sum_{i=1}^{2^t-1} \mathbf{V}_{t,i}[0](1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0])^n / q_t^n\right)}{n} \\
&= -\log q_t.
\end{aligned}$$

The proof is completed. ■

Appendix IV
Proofs about Poisson Number of Batches

Proof of Theorem 7: Let $\bar{\mathbf{Q}}_t$ be a $(K+1)\times(K+1)$ matrix such that $\bar{\mathbf{Q}}_t[t:, t:] = \mathbf{Q}_t$, and all the other components of $\bar{\mathbf{Q}}_t$ are zero. For integers $n \geq 0$ and $t \geq 0$ define $(n+1)\times(K+1)$ matrix $\bar{\mathbf{\Lambda}}_n^{(t)}$ recursively as follows: i) $\bar{\mathbf{\Lambda}}_n^{(0)} = \mathbf{\Lambda}_n^{(0)}$, and ii) for $t > 0$,

$$\bar{\mathbf{\Lambda}}_n^{(t)}[c,:] = \sum_{c'=c}^{n} \text{Bi}(c; c', 1-\rho_t)\bar{\mathbf{\Lambda}}_n^{(t-1)}[c',:]\bar{\mathbf{Q}}_t^{c'-c}. \tag{52}$$

Note that compared with the iterative formula in Theorem 1, $\bar{\mathbf{\Lambda}}_n^{(t-1)}$ in the above formula is not shortened.

We show that

$$\bar{\mathbf{\Lambda}}_n^{(t)}[:,i] = \mathbf{\Lambda}_n^{(i)}[:,0], \quad i = 0, \dots, t, \tag{53}$$

$$\bar{\mathbf{\Lambda}}_n^{(t)}[:,t+1:] = \mathbf{\Lambda}_n^{(t)}[:,1:], \tag{54}$$

by induction in t. The claim holds for $t=0$ by definition. Suppose that (53) and (54) hold for certain $t \geq 0$. We have by the definition that for $0 \leq c \leq n$,

$$\bar{\mathbf{\Lambda}}_n^{(t+1)}[c,:] = \bar{\mathbf{\Lambda}}_n^{(t)}[c,:] + \sum_{c'=c+1}^{n} \text{Bi}(c; c', 1-\rho_{t+1})\bar{\mathbf{\Lambda}}_n^{(t)}[c',:]\bar{\mathbf{Q}}_{t+1}^{c'-c}.$$

Since the first $t+1$ columns of $\bar{\mathbf{Q}}_{t+1}$ are all zero, we have for $i = 0, \ldots, t$, $\bar{\mathbf{\Lambda}}_n^{(t+1)}[c, i] = \bar{\mathbf{\Lambda}}_n^{(t)}[c, i] = \mathbf{\Lambda}_n^i[c, 0]$. Since the first $t+1$ rows of $\bar{\mathbf{Q}}_{t+1}$ are all zero, we can write

$$\begin{aligned}
\bar{\mathbf{\Lambda}}_n^{(t+1)}[c, t+1{:}] &= \bar{\mathbf{\Lambda}}_n^{(t)}[c, t+1{:}] + \sum_{c'=c+1}^{n} \mathrm{Bi}(c; c', 1-\rho_{t+1}) \bar{\mathbf{\Lambda}}_n^{(t)}[c', t+1{:}] \mathbf{Q}_{t+1}^{c'-c} \\
&= \sum_{c'=c}^{n} \mathrm{Bi}(c; c', 1-\rho_{t+1}) \mathbf{\Lambda}_n^{(t)}[c', 1{:}] \mathbf{Q}_{t+1}^{c'-c} \\
&= \mathbf{\Lambda}_n^{(t+1)}[c, :],
\end{aligned}$$

where the second equality follows from the induction hypothesis and the last equality follows by Theorem 1.

Expending the recursive formula (52), we have

$$\begin{aligned}
\bar{\mathbf{\Lambda}}_n^{(t)}[c, :] &= \mathbf{e}_0 \sum \mathrm{Bi}(c_0; n, 1-\rho_0) \bar{\mathbf{Q}}_0^{n-c_0} \times \\
&\quad \times \mathrm{Bi}(c_1; c_0, 1-\rho_1) \bar{\mathbf{Q}}_1^{c_0-c_1} \times \cdots \times \mathrm{Bi}(c; c_{t-1}, 1-\rho_t) \bar{\mathbf{Q}}_t^{c_{t-1}-c} \\
&= \mathbf{e}_0 \sum \binom{n}{c_0} (1-\rho_0)^{c_0} (\rho_0 \bar{\mathbf{Q}}_0)^{n-c_0} \times \qquad (55) \\
&\quad \times \binom{c_0}{c_1} (1-\rho_1)^{c_1} (\rho_1 \bar{\mathbf{Q}}_1)^{c_0-c_1} \times \cdots \times \binom{c_{t-1}}{c} (1-\rho_t)^{c} (\rho_t \bar{\mathbf{Q}}_t)^{c_{t-1}-c}
\end{aligned}$$

where the summation is over all $(c_0, \ldots, c_{t-1})$ such that $n \geq c_0 \geq c_1 \geq \cdots \geq c_{t-1} \geq c$. Reorganizing (55) using Lemma 2, we obtain

$$\bar{\mathbf{\Lambda}}_n^{(t)}[c, :] = \mathbf{e}_0 \sum \binom{n}{c_0} \binom{c_0}{c_1} \cdots \binom{c_{t-1}}{c} \left(\frac{p_{t+1}}{\rho_{t+1}}\right)^c \left(p_0 \bar{\mathbf{Q}}_0\right)^{n-c_0} \left(p_1 \bar{\mathbf{Q}}_1\right)^{c_0-c_1} \cdots \left(p_t \bar{\mathbf{Q}}_t\right)^{c_{t-1}-c}.$$

Define

$$\check{\mathbf{\Lambda}}_{\bar{n}}^{(t)} = \sum_n \frac{\bar{n}^n}{n!} e^{-\bar{n}} \sum_c \bar{\mathbf{\Lambda}}_n^{(t)}[c, :].$$

By (17), (53) and (54), we have

$$\check{\mathbf{\Lambda}}_{\bar{n}}^{(t)}[t{:}] = \tilde{\mathbf{\Lambda}}_{\bar{n}}^{(t)}. \qquad (56)$$

Substituting the expression of $\bar{\mathbf{\Lambda}}_n^{(t)}[c, :]$ and using the fact that

$$\binom{n}{c_0} \binom{c_0}{c_1} \cdots \binom{c_{t-1}}{c} = \frac{n!}{(n-c_0)!(c_0-c_1)! \cdots (c_{t-1}-c)! c!},$$

we have

$$\check{\mathbf{\Lambda}}_{\bar{n}}^{(t)} = \mathbf{e}_0 \sum e^{-\bar{n}} \frac{\left(\bar{n} \frac{p_{t+1}}{\rho_{t+1}}\right)^c}{c!} \frac{(\bar{n} p_0 \bar{\mathbf{Q}}_0)^{n-c_0}}{(n-c_0)!} \frac{(\bar{n} p_1 \bar{\mathbf{Q}}_1)^{c_0-c_1}}{(c_0-c_1)!} \cdots \frac{(\bar{n} p_t \bar{\mathbf{Q}}_t)^{c_{t-1}-c}}{(c_{t-1}-c)!},$$

where the summation is over all $(n, c_0, \ldots, c_{t-1}, c)$ such that $n \geq c_0 \geq c_1 \geq \cdots \geq c_{t-1} \geq c$.

Let $x_{t+1} = c$, $x_0 = n - c_0$, $x_t = c_{t-1} - c$ and $x_\tau = c_{\tau-1} - c_\tau$ for $1 \leq \tau \leq t-1$. We can rewrite the above expression as

$$
\begin{aligned}
\check{\mathbf{\Lambda}}_{\bar{n}}^{(t)} &= \mathbf{e}_0 \sum_{x_\tau:\tau=0,\ldots,t+1} e^{-\bar{n}} \frac{\left(\bar{n}\frac{p_{t+1}}{\rho_{t+1}}\right)^{x_{t+1}}}{x_{t+1}!} \frac{(\bar{n}p_0\bar{\mathbf{Q}}_0)^{x_0}}{x_0!} \frac{(\bar{n}p_1\bar{\mathbf{Q}}_1)^{x_1}}{x_1!} \cdots \frac{(\bar{n}p_t\bar{\mathbf{Q}}_t)^{x_t}}{x_t!} \\
&= \mathbf{e}_0 e^{-\bar{n}} \sum_{x_{t+1}} \frac{\left(\bar{n}\frac{p_{t+1}}{\rho_{t+1}}\right)^{x_{t+1}}}{x_{t+1}!} \sum_{x_0} \frac{(\bar{n}p_0\bar{\mathbf{Q}}_0)^{x_0}}{x_0!} \sum_{x_1} \frac{(\bar{n}p_1\bar{\mathbf{Q}}_1)^{x_1}}{x_1!} \cdots \sum_{x_t} \frac{(\bar{n}p_t\bar{\mathbf{Q}}_t)^{x_t}}{x_t!} \\
&= \mathbf{e}_0 e^{-\bar{n}} \exp\left(\bar{n}\frac{p_{t+1}}{\rho_{t+1}}\right) \exp\left(\bar{n}p_0\bar{\mathbf{Q}}_0\right) \exp\left(\bar{n}p_1\bar{\mathbf{Q}}_1\right) \cdots \exp\left(\bar{n}p_t\bar{\mathbf{Q}}_t\right) \qquad (57) \\
&= \mathbf{e}_0 \exp\left(-\bar{n}\left(1-\frac{p_{t+1}}{\rho_{t+1}}\right)\right) \exp\left(\bar{n}p_0\bar{\mathbf{Q}}_0\right) \exp\left(\bar{n}p_1\bar{\mathbf{Q}}_1\right) \cdots \exp\left(\bar{n}p_t\bar{\mathbf{Q}}_t\right) \\
&= \mathbf{e}_0 \exp\left(-\bar{n}\left(\sum_{\tau=0}^{t} p_\tau\right)\right) \exp\left(\bar{n}p_0\bar{\mathbf{Q}}_0\right) \exp\left(\bar{n}p_1\bar{\mathbf{Q}}_1\right) \cdots \exp\left(\bar{n}p_t\bar{\mathbf{Q}}_t\right), \qquad (58)
\end{aligned}
$$

where (57) is obtained using the definition of matrix exponential, and (58) follows from the definition of ρ_t. Thus, we have

$$
\check{\mathbf{\Lambda}}_{\bar{n}}^{(t)} = \exp\left(-\bar{n}p_t\right) \check{\mathbf{\Lambda}}_{\bar{n}}^{(t-1)} \exp\left(\bar{n}p_t\bar{\mathbf{Q}}_t\right)
$$

with $\check{\mathbf{\Lambda}}_{\bar{n}}^{0} = \tilde{\mathbf{\Lambda}}_{\bar{n}}^{0}$ given in (20). The proof is complete by noting (56) and $\exp(\bar{n}p_t\bar{\mathbf{Q}}_t) = \begin{bmatrix} \mathbf{I} & \\ & \exp(\bar{n}p_t\mathbf{Q}_t) \end{bmatrix}$. ∎

Proof of Theorem 8: We prove the theorem using

$$
\tilde{P}_{\text{stop}}(t|\bar{n}) = \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \exp\left(-\bar{n}\left(\sum_{\tau=0}^{t} p_\tau - \mathbf{\Delta}_{t,i}[0,0]\right)\right).
$$

When $t=0$, we have $\tilde{P}_{\text{stop}}(0|\bar{n}) = \mathbf{V}_{0,0}[0]\exp\left(-\bar{n}\left(p_0 - \mathbf{\Delta}_{0,0}[0,0]\right)\right) = \exp\left(-\bar{n}\left(p_0 - \lambda_{0,0}\right)\right) = \exp(-\bar{n}p_0)$, where the last equality follows from $\lambda_{0,0} = 0$ (see (44)). Hence i) is proved by noting $q_0 = 1 - p_0$. When $1 \le t < r_{\text{BP}}$, since $\mathbf{\Delta}_{t,i}[0,0] = 0$ (see (47)), we have $\tilde{P}_{\text{stop}}(t|\bar{n}) = \exp\left(-\bar{n}\sum_{\tau=0}^{t} p_\tau\right) \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] = 0$, where the last equality follows from (49), proving ii). To prove iii), by (51) and $\mathbf{V}_{t,0}[0] = \mathbf{U}_0[0,t] = \binom{K}{t} > 0$, we write

$$
\begin{aligned}
\lim_{\bar{n}\to\infty} \frac{-\log \tilde{P}_{\text{stop}}(t|\bar{n})}{\bar{n}} &= \lim_{\bar{n}\to\infty} \frac{-\log \exp(-\bar{n}(1-q_t)) \sum_{i=0}^{2^t-1} \mathbf{V}_{t,i}[0] \exp\left(-\bar{n}\left(\sum_{\tau=0}^{t} p_\tau - \mathbf{\Delta}_{t,i}[0,0] - 1 + q_t\right)\right)}{\bar{n}} \\
&= 1 - q_t.
\end{aligned}
$$

The proof is completed. ∎

Appendix V

Proofs about Inactivation

Proof of Theorem 11: First, we have $\mathbf{\Lambda}_n^{(0)} = \mathbf{\Gamma}_n^{(0)}$ by their definitions, proving the formula for $t=0$. For $t>0$, define matrices $\mathbf{\Gamma}_n^{t(1)}$ and $\mathbf{\Gamma}_n^{t(2)}$ as

$$
\begin{aligned}
\mathbf{\Gamma}_n^{(t1)}[c,r] &= \Pr\left\{\hat{C}^{(t)} = c, \hat{R}^{(t)} = r, \hat{R}^{(t-1)} > 0\right\} \\
\mathbf{\Gamma}_n^{(t2)}[c,r] &= \Pr\left\{\hat{C}^{(t)} = c, \hat{R}^{(t)} = r, \hat{R}^{(t-1)} = 0\right\}.
\end{aligned}
$$

Since

$$\mathbf{\Gamma}_n^{(t)} = \mathbf{\Gamma}_n^{(t1)} + \mathbf{\Gamma}_n^{(t2)},$$

we characterize the two terms on the RHS.

Write

$$\mathbf{\Gamma}_n^{(t1)}[c,r] = \sum_{c'}\sum_{r'>0} \underbrace{\Pr\{\hat{R}^{(t)} = r|\hat{C}^{(t)} = c, \hat{C}^{(t-1)} = c', \hat{R}^{(t-1)} = r'\}}_{(a)} \times$$
$$\times \underbrace{\Pr\{\hat{C}^{(t)} = c|\hat{C}^{(t-1)} = c', \hat{R}^{(t-1)} = r'\}}_{(b)} \mathbf{\Gamma}_n^{(t-1)}[c', r'],$$

where term (a) and (b) can be obtained using Lemma 6 and Lemma 5, respectively, since only normal BP decoding is applied from time $t-1$ to t when $\hat{R}^{(t-1)} > 0$. Similar to the procedure for obtaining (34), we have

$$\mathbf{\Gamma}_n^{(t1)}[c,:] = \sum_{c' \geq c} \mathrm{Bi}(c; c', 1-\rho_t)\mathbf{\Gamma}_n^{(t-1)}[c', 1{:}]\mathbf{Q}_t^{c'-c}. \tag{59}$$

The components in $\mathbf{\Gamma}_n^{t(2)}$ corresponds to the case that inactivation occurs during from time $t-1$ to time t, where an undecoded input symbol is marked as inactive and is treated as decoded. We write

$$\mathbf{\Gamma}_n^{(t2)}[c,r] = \sum_{c'} \underbrace{\Pr\{\hat{R}^{(t)} = r|\hat{C}^{(t)} = c, \hat{C}^{(t-1)} = c', \hat{R}^{(t-1)} = 0\}}_{(c)} \times$$
$$\times \underbrace{\Pr\{\hat{C}^{(t)} = c|\hat{C}^{(t-1)} = c', \hat{R}^{(t-1)} = 0\}}_{(d)} \mathbf{\Gamma}_n^{(t-1)}[c', 0].$$

Since the inactive symbol in the decoding step from time $t-1$ to t can be regarded as the only decodable input symbol in time $t-1$, we can obtain (c) and (d) using Lemma 6 with $r' = 1$ and Lemma 5 with $r' = 1$, respectively. Thus, we have

$$\mathbf{\Gamma}_n^{t(2)}[c,:] = \sum_{c' \geq c} \mathrm{Bi}(c; c', 1-\rho_t)\mathbf{\Gamma}_n^{(t-1)}[c', 0]\mathbf{e}_0\mathbf{Q}_t^{c'-c}. \tag{60}$$

Combining (59) and (60), the recursive formula of Theorem 11 is proved. ■

Proof of Theorem 12: We first show by induction that for $1 \leq c \leq n$ and $t \geq 0$,

$$\mathbf{\Gamma}_n^{(t)}[c,:] = \frac{n}{c}\prod_{i=0}^{t}(1-\rho_i)\mathbf{\Gamma}_{n-1}^{(t)}[c-1,:]. \tag{61}$$

Since $\mathbf{\Gamma}_n^{(0)} = \mathbf{\Lambda}_n^{(0)}$, we have by (38) that (61) holds with Suppose that (61) holds for certain $t \geq 0$. Applying the recursive formula of Theorem 11, we can show that

$$\begin{aligned}\mathbf{\Gamma}_{n+1}^{(t)}[c,:] &= \sum_{c'=c}^{n+1} \mathrm{Bi}(c;c',1-\rho_t)\mathbf{\Gamma}_{n+1}^{(t-1)}[c',:]\mathbf{N}_t\mathbf{Q}_t^{c'-c} \\ &= \sum_{c'=c}^{n+1} \mathrm{Bi}(c;c',1-\rho_t)\frac{n+1}{c'}\prod_{i=0}^{t-1}(1-\rho_i)\mathbf{\Gamma}_{n}^{(t-1)}[c',:]\mathbf{N}_t\mathbf{Q}_t^{c'-c} \\ &= \frac{n+1}{c}\prod_{i=0}^{t}(1-\rho_i)\sum_{c'=c}^{n+1} \mathrm{Bi}(c-1;c'-1,1-\rho_t)\mathbf{\Gamma}_{n}^{(t-1)}[c'-1,:]\mathbf{N}_t\mathbf{Q}_t^{c'-c} \\ &= \frac{n+1}{c}\prod_{i=0}^{t}(1-\rho_i)\sum_{c''=c-1}^{n} \mathrm{Bi}(c-1;c'',1-\rho_t)\mathbf{\Gamma}_{n}^{(t-1)}[c'',:]\mathbf{N}_t\mathbf{Q}_t^{c''-(c-1)} \\ &= \frac{n+1}{c}\prod_{i=0}^{t}(1-\rho_i)\mathbf{\Gamma}_{n}^{(t)}[c-1,:].\end{aligned}$$

By expanding (61) recursively, we have for $c \geq 0$ and $t \geq 0$,

$$\mathbf{\Gamma}_{n}^{(t)}[c,:] = \binom{n}{c}\prod_{i=0}^{t}(1-\rho_i)^c\mathbf{\Gamma}_{n-c}^{(t)}[0,:]. \tag{62}$$

Substituting (62) into (21) and by Lemma 2, we get

$$P_{\text{inac}}(t|n) = \sum_{c=0}^{n}\binom{n}{c}\prod_{i=0}^{t}(1-\rho_i)^c\mathbf{\Gamma}_{n-c}^{(t)}[0,0] = \sum_{c=0}^{n}\binom{n}{c}\left(1-\sum_{\tau=0}^{t}p_\tau\right)^c\mathbf{\Gamma}_{n-c}^{(t)}[0,0],$$

proving the formula of $P_{\text{inac}}(t|n)$. Further, (25) is obtained by (22) for $c = 0$. To prove (26), we have

$$\begin{aligned}\mathbf{\Gamma}_{n}^{(t)}[0,:] &= \sum_{c=0}^{n}\rho_t^c\mathbf{\Gamma}_{n}^{(t-1)}[c,:]\mathbf{N}_t\mathbf{Q}_t^c \\ &= \sum_{c=0}^{n}\binom{n}{c}\rho_t^c\prod_{i=0}^{t-1}(1-\rho_i)^c\mathbf{\Gamma}_{n-c}^{(t-1)}[0,:]\mathbf{N}_t\mathbf{Q}_t^c \\ &= \sum_{c=0}^{n}\binom{n}{c}p_t^c\mathbf{\Gamma}_{n-c}^{(t-1)}[0,:]\mathbf{N}_t\mathbf{Q}_t^c\end{aligned}$$

where the first equality follows from (23) with $c = 0$, the second equality is obtained by substituting (62), and the last step is obtained by applying Lemma 2. ■

Proof of Theorem 13: We first show

$$\mathbf{\Gamma}_{n}^{(t)}[0,:] = \sum_{i=0}^{2^t-1}\mathbf{V}'_{t,i}\mathbf{\Delta}_{t,i}^{n}\mathbf{U}_t^{-1} \tag{63}$$

by induction in t. The claim for $t = 0$ can be shown by replacing $p_0\mathbf{Q}_0$ in (25) with the decomposition in Lemma 3. Suppose that the claim of the theorem holds for certain $t \geq 0$. Substituting this form of $\mathbf{\Lambda}_n^t$ into (26) with $t+1$ in place of t, we obtain

$$\mathbf{\Gamma}_{n}^{(t+1)}[0,:] = \sum_{i=0}^{2^t-1}\mathbf{V}'_{t,i}\sum_{c=0}^{n}\binom{n}{c}\mathbf{\Delta}_{t,i}^{n-c}\mathbf{U}_t^{-1}\mathbf{N}_{t+1}\mathbf{U}_{t+1}(p_{t+1}\mathbf{D}_{t+1})^c\mathbf{U}_{t+1}^{-1}. \tag{64}$$

We can verify that

$$\begin{aligned}\mathbf{U}_t^{-1}\mathbf{N}_{t+1}\mathbf{U}_{t+1} &= (\mathbf{U}_t^{-1}[:,0]\mathbf{e}_0 + \mathbf{U}_t^{-1}[:,1:])\mathbf{U}_{t+1} \\ &= \begin{bmatrix} \mathbf{U}_{t+1}[0,:] - \mathbf{U}_t[0,1:] \\ \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{U}_{t+1}[0,:] - \mathbf{U}_t[0,1:] \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix},\end{aligned}$$

where the second equality follows from (40). Similar to the steps obtaining (42), substituting the above equation into (64) and combining the binomial terms, we get

$$\begin{aligned}\mathbf{\Lambda}_n^{(t+1)} &= \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[1:] \left(\mathbf{\Delta}_{t,i}[1:,1:] + p_{t+1}\mathbf{D}_{t+1}\right)^n \mathbf{U}_{t+1}^{-1} \\ &+ \sum_{i=0}^{2^t-1} (\mathbf{U}_{t+1}[0,:] - \mathbf{U}_t[0,1:])\mathbf{V}'_{t,i}[0] \left(\mathbf{\Delta}_{t,i}[0,0]\mathbf{I} + p_{t+1}\mathbf{D}_{t+1}\right)^n \mathbf{U}_{t+1}^{-1}.\end{aligned}$$

The proof of (63) is completed by checking the definition of $\mathbf{V}'_{t+1,i}$ and $\mathbf{\Delta}_{t+1,i}$.

Substituting the above formula of $\mathbf{\Gamma}_n^{(t)}$ in (63) into (24), we obtain the following formula of $P_{\text{inac}}(t|n)$ given in this theorem. ■

Proof of Theorem 14: When $t < r_{\text{BP}}$, we know that $\mathbf{\Delta}_{t,i}[0,0] = 0$ (see (47)). So

$$P_{\text{inac}}(t|n) = \left(1 - \sum_{\tau=0}^{t} p_\tau\right)^n \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0] = q_t^n \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0].$$

It can be shown inductively that

$$\sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i} = \mathbf{U}_t[0,:]. \tag{65}$$

First, by definition $\mathbf{V}'_{0,0} = \mathbf{U}_0[0,:]$. Suppose that (65) holds for certain $t > 0$. We write

$$\begin{aligned}\sum_{i=0}^{2^t+1} \mathbf{V}'_{t+1,i} &= \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[1:] + \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0](\mathbf{U}_{t+1}[0,:] - \mathbf{U}_t[0,1:]) \\ &= \mathbf{U}_t[0,1:] + \mathbf{U}_t[0,0](\mathbf{U}_{t+1}[0,:] - \mathbf{U}_t[0,1:]) \\ &= \mathbf{U}_{t+1}[0,:],\end{aligned}$$

where the second equality follows by the induction hypothesis and the last equality follows by $\mathbf{U}_t[0,0] = 1$. By (65), we have $P_{\text{inac}}(t|n) = q_t^n \mathbf{U}_t[0,0] = q_t^n$.

When $t \geq r_{\text{BP}}$, by (51), we know that for $i = 1, \ldots, 2^t - 1$,

$$\frac{1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0]}{q_t} < 1. \tag{66}$$

By Theorem 13 and noting that $\mathbf{V}'_{t,0}[0] = \mathbf{U}_0[0,t] = \binom{K}{t} > 0$, we write

$$\begin{aligned}\lim_{n\to\infty} \frac{-\log P_{\text{inac}}(t|n)}{n} &= \lim_{n\to\infty} \frac{-\log q_t^n \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0](1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0])^n / q_t^n}{n} \\ &= -\log q_t + \lim_{n\to\infty} \frac{-\log\left(\mathbf{V}'_{t,0}[0] + \sum_{i=1}^{2^t-1} \mathbf{V}'_{t,i}[0](1 - \sum_{\tau=0}^{t} p_\tau + \mathbf{\Delta}_{t,i}[0,0])^n / q_t^n\right)}{n} \\ &= -\log q_t.\end{aligned}$$

The proof is completed. ■

Proof of Theorem 16: The recursive formula in Theorem 11 can be rewritten into a form similar to (52) as:

$$\bar{\mathbf{\Gamma}}_n^{(t)}[c,:] = \sum_{c' \geq c} \mathrm{Bi}(c; c', 1-\rho_t) \bar{\mathbf{\Gamma}}_n^{(t-1)}[c',:] \bar{\mathbf{N}}_t \bar{\mathbf{Q}}_t^{c'-c},$$

where $\bar{\mathbf{\Gamma}}_n^{(t)}[c,:] = (\mathbf{0} \;\; \mathbf{\Gamma}_n^{(t)}[c,:])$ and $\bar{\mathbf{N}}_t$ is a $(K+1) \times (K+1)$ matrix such $\bar{\mathbf{N}}_t[t-1:,t:] = \mathbf{N}_t$ and all the other components are zeros. The proof can be completed by following the steps after (52) and using the fact (27). ■

Proof of Theorem 17: When $t < r_{\mathrm{BP}}$, we know that $\mathbf{\Delta}_{t,i}[0,0] = 0$ (see (47)). So

$$\tilde{P}_{\text{inac}}(t|\bar{n}) = \exp\left(-\bar{n} \sum_{\tau=0}^{t} p_\tau\right) \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0],$$

where $\sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0] = \mathbf{U}_t[0,0] = 1$ by (65).

When $t \geq r_{\mathrm{BP}}$, by (66) and $\mathbf{V}'_{t,0}[0] = \mathbf{U}_0[0,t] = \binom{K}{t} > 0$, we write

$$\begin{aligned}\lim_{\bar{n}\to\infty} \frac{-\log \tilde{P}_{\text{inac}}(t|\bar{n})}{\bar{n}} &= \lim_{\bar{n}\to\infty} \frac{-\log \exp(-\bar{n}(1-q_t)) \sum_{i=0}^{2^t-1} \mathbf{V}'_{t,i}[0] \exp\left(-\bar{n}\left(\sum_{\tau=0}^{t} p_\tau - \mathbf{\Delta}_{t,i}[0,0] - 1 + q_t\right)\right)}{\bar{n}} \\ &= 1 - q_t.\end{aligned}$$

The proof is completed. ■

Appendix VI
Tables of Degree Distributions

Several degree distributions used in the numerical evaluations of this paper are listed in Table III.

References

[1] S. Yang and R. W. Yeung, "Coding for a network coded fountain," in *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, Saint Petersburg, Russia, July 31 - Aug. 5 2011, pp. 2647–2651.

[2] S. Yang and R. Yeung, "Batched sparse codes," *Information Theory, IEEE Transactions on*, vol. 60, no. 9, pp. 5322–5346, Sep 2014.

[3] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inform. Theory*, vol. 49, no. 2, pp. 371–381, Feb. 2003.

[4] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE/ACM Trans. Networking*, vol. 11, no. 5, pp. 782–795, Oct. 2003.

[5] T. Ho, B. Leong, M. Medard, R. Koetter, Y. Chang, and M. Effros, "The benefits of coding over routing in a randomized setting," in *Proc. IEEE ISIT '03*, Jun. 2003.

[6] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Communication*, vol. 1, no. 1, pp. 3–20, 2008.

[7] Y. Wu, "A trellis connectivity analysis of random linear network coding with buffering," in *Proc. IEEE ISIT '06*, Seattle, USA, Jul. 2006.

[8] A. F. Dana, R. Gowaikar, R. Palanki, B. Hassibi, and M. Effros, "Capacity of wireless erasure networks," *IEEE Trans. Inform. Theory*, vol. 52, no. 3, pp. 789–804, 2006.

[9] B. Tang, S. Yang, Y. Yin, B. Ye, and S. Lu, "Expander graph based overlapped chunked codes," in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, Cambridge, MA, USA, July 1-6 2012, pp. 2451–2455.

[10] K. Mahdaviani, M. Ardakani, H. Bagheri, and C. Tellambura, "Gamma codes: A low-overhead linear-complexity network coding solution," in *Network Coding (NetCod), 2012 International Symposium on*, June 2012, pp. 125–130.

[11] S. Yang and B. Tang, "From LDPC to chunked network codes," in *Information Theory Workshop (ITW), 2014 IEEE*, Nov 2014, pp. 406–410.

TABLE III

DEGREE DISTRIBUTIONS FOR THE RANK DISTRIBUTION IN TABLE I. FOR THE FIRST THREE DEGREE DISTRIBUTIONS, WE GIVE THE VALUES OF THE SAME SET OF PROBABILITY MASSES, THAT INCLUDE ALL THE POSITIVE PROBABILITY MASSES OF THESE DISTRIBUTIONS. FOR THE FORTH DEGREE DISTRIBUTION, ONLY THE POSITIVE PROBABILITY MASSES ARE LISTED.

(a) $\mathbf{\Psi}^{\mathrm{asy}}$: the degree distribution obtained using the asymptotic analysis.

Ψ_{11}^{asy}	Ψ_{12}^{asy}	Ψ_{13}^{asy}	Ψ_{14}^{asy}	Ψ_{15}^{asy}	Ψ_{20}^{asy}	Ψ_{21}^{asy}	Ψ_{26}^{asy}	Ψ_{27}^{asy}
0	0	0	0.0467	0.2502	0.1079	0.0781	0	0.0350
Ψ_{28}^{asy}	Ψ_{37}^{asy}	Ψ_{38}^{asy}	Ψ_{50}^{asy}	Ψ_{51}^{asy}	Ψ_{72}^{asy}	$\Psi_{116}^{\mathrm{asy}}$	$\Psi_{117}^{\mathrm{asy}}$	$\Psi_{256}^{\mathrm{asy}}$
0.0968	0.0728	0.0199	0.0676	0.0087	0.0679	0.0277	0.0312	0.0896

(b) $\mathbf{\Psi}^{\mathrm{BP}}$: the degree distribution obtained by modifying Ψ^{asy} using the approach introduced in Section VI for BP decoding.

Ψ_{11}^{BP}	Ψ_{12}^{BP}	Ψ_{13}^{BP}	Ψ_{14}^{BP}	Ψ_{15}^{BP}	Ψ_{20}^{BP}	Ψ_{21}^{BP}	Ψ_{26}^{BP}	Ψ_{27}^{BP}
0.0826	0.0734	0.0550	0.0429	0.1745	0.0348	0.0809	0	0.0321
Ψ_{28}^{BP}	Ψ_{37}^{BP}	Ψ_{38}^{BP}	Ψ_{50}^{BP}	Ψ_{51}^{BP}	Ψ_{72}^{BP}	Ψ_{116}^{BP}	Ψ_{117}^{BP}	Ψ_{256}^{BP}
0.0888	0.0484	0.0183	0.0620	0.0080	0.0623	0.0254	0.0286	0.0822

(c) $\mathbf{\Psi}^{\mathrm{inac}}$: the degree distribution obtained by modifying $\mathbf{\Psi}^{\mathrm{asy}}$ using the approach introduced in Section VI for inactivation decoding.

$\Psi_{11}^{\mathrm{inac}}$	$\Psi_{12}^{\mathrm{inac}}$	$\Psi_{13}^{\mathrm{inac}}$	$\Psi_{14}^{\mathrm{inac}}$	$\Psi_{15}^{\mathrm{inac}}$	$\Psi_{20}^{\mathrm{inac}}$	$\Psi_{21}^{\mathrm{inac}}$	$\Psi_{26}^{\mathrm{inac}}$	$\Psi_{27}^{\mathrm{inac}}$
0	0.0796	0.0973	0.0414	0.2126	0.0955	0.0692	0.0088	0.0309
$\Psi_{28}^{\mathrm{inac}}$	$\Psi_{37}^{\mathrm{inac}}$	$\Psi_{38}^{\mathrm{inac}}$	$\Psi_{50}^{\mathrm{inac}}$	$\Psi_{51}^{\mathrm{inac}}$	$\Psi_{72}^{\mathrm{inac}}$	$\Psi_{116}^{\mathrm{inac}}$	$\Psi_{117}^{\mathrm{inac}}$	$\Psi_{256}^{\mathrm{inac}}$
0.0857	0.0644	0.0176	0.0598	0.0077	0.0512	0.0245	0.0276	0.0262

(d) $\mathbf{\Psi}^{\mathrm{mee}}$: the degree distribution that maximizes the asymptotic decrease rate of error probability (and the number of inactivations) obtained by solving (12).

Ψ_{10}^{mee}	Ψ_{70}^{mee}	Ψ_{71}^{mee}	$\Psi_{110}^{\mathrm{mee}}$	$\Psi_{111}^{\mathrm{mee}}$	$\Psi_{165}^{\mathrm{mee}}$	$\Psi_{265}^{\mathrm{mee}}$
0.4584	0.0063	0.0706	0.0112	0.0650	0.0751	0.3135

[12] Q. Huang, K. Sun, X. Li, and D. O. Wu, "Just fun: A joint fountain coding and network coding approach to loss-tolerant information spreading," in *Proceedings of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '14. New York, NY, USA: ACM, 2014, pp. 83–92. [Online]. Available: http://doi.acm.org/10.1145/2632951.2632965

[13] S. Yang, R. W. Yeung, H. F. Cheung, and H. H. Yin, "BATS: Network coding in action," in *Communication, Control, and Computing (Allerton), 2014 52nd Annual Allerton Conference on*, Sept 2014, pp. 1204–1211.

[14] X. Xu, P. K. M. Gandhi, Y. L. Guan, and P. H. J. Chong, "Two-phase cooperative broadcasting based on batched network code," *arXiv preprint arXiv:1504.04464*, 2015.

[15] R. Karp, M. Luby, and A. Shokrollahi, "Finite length analysis of LT codes," in *Proc. Int. Symp. Information Theory ISIT 2004*, 2004.

[16] E. Maneva and A. Shokrollahi, "New model for rigorous analysis of LT-codes," in *Proc. IEEE ISIT '06*, Jul. 2006, pp. 2677–2679.

[17] A. Shokrollahi, S. Lassen, and R. Karp, "Systems and processes for decoding chain reaction codes through inactivation," U.S. Patent 6,856,263, Feb. 15, 2005.

[18] A. Shokrollahi and M. Luby, *Raptor Codes*, ser. Foundations and Trends in Communications and Information Theory. now, 2011, vol. 6.

[19] F. L. Blasco, G. Liva, and G. Bauch, "Finite length analysis of lt codes under inactivation decoding," private communication, 2015.

[20] T. C. Ng and S. Yang, “Finite-length analysis of BATS codes,” 2013, arXiv preprint arXiv:1312.4811v1. [Online]. Available: http://arxiv.org/abs/1312.4811v1

[21] C. Moler and C. Van Loan, “Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later,” *SIAM review*, vol. 45, no. 1, pp. 3–49, 2003.

[22] N. J. Higham, “The scaling and squaring method for the matrix exponential revisited,” *SIAM Journal on Matrix Analysis and Applications*, vol. 26, no. 4, pp. 1179–1193, 2005.

[23] A. H. Al-Mohy and N. J. Higham, “Computing the action of the matrix exponential, with an application to exponential integrators,” *SIAM Journal on Scientific Computing*, vol. 33, no. 2, pp. 488–511, 2011.

[24] B. A. LaMacchia and A. M. Odlyzko, “Solving large sparse linear systems over finite fields,” in *Proc. CRYPTO’90*. Springer, 1991, pp. 109–133.

[25] C. Pomerance and J. W. Smith, “Reduction of huge, sparse matrices over finite fields via created catastrophes,” *Experimental Math*, vol. 1, no. 89-94, 1992.

[26] T. Richardson and R. Urbanke, “Efficient encoding of low-density parity-check codes,” *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 638–656, 2001.

[27] A. Shokrollahi, “Raptor codes,” *Information Theory, IEEE Transactions on*, vol. 52, no. 6, pp. 2551–2567, June 2006.

[28] F. L. Blasco, G. Liva, and G. Bauch, “LT code design for inactivation decoding,” in *Information Theory Workshop (ITW), 2014 IEEE*, Nov 2014, pp. 441–445.