The influence of active distances on the distribution of bursts

Anastasiia Smeshko
Institute for Information Transmission Problems
Russian Academy of Sciences,
National Research University
Higher School of Economics,
Moscow Institute of Physics and Technology
Moscow, Russia
smeshko.aa@phystech.edu

Fedor Ivanov
National Research University
Higher School of Economics,
Institute for Information Transmission Problems
Russian Academy of Sciences
Moscow, Russia
fii@iitp.ru

Victor Zyablov
Institute for Information Transmission Problems
Russian Academy of Sciences
Moscow, Russia
zyablov@iitp.ru

***Abstract*—In this paper, we consider the dependence between active distances for convolutional code and the distribution of bursts at the output after Viterbi decoding. We suggest an estimate of the probability of bursts of a certain length based on the active row distances of the code and present a formula for our estimation.**

We take into consideration two recursive systematic convolutional codes with the same free distance, same memory, but with different active distances. Simulation results show that active distances affect the distribution of error bursts.

I. Introduction

Convolutional codes are being studied for a long time. They are widely used for data transmission in different systems. That's why the issues of the constructing and studying capabilities of convolutional codes have great practical importance [1], [2]. The scheme of recursive systematic codes was proposed and described in works devoted to turbo-codes [3]- [4] and it has many applications nowadays. Many works have been devoted to the decoding of convolutional codes; algorithms of maximum likelihood, in particular, Viterbi, have gained the greatest popularity [5]- [7]. The representation of the path trace Viterbi decoder is described in [8]. The bursts distribution at the output of Viterbi decoder has been considered and results are showing that the distribution of bursts with great length can be estimated with geometrical distribution [9], [10]. Nevertheless, convolutional codes have unique local correcting and detecting features [11]- [13], and distance properties. The active distances of the convolutional code that can be generalized from the "extended" distances introduced in [14] are described in detail in [15]. In this work, we propose to consider the relationship between the bursts distribution and active distances of the convolutional code.

In this paper, we consider two systematic recursive convolutional codes $(13, 17)$ and $(13, 15)$ that have the same code memory, same free distance and different distance properties, in particular, different active row distances. We study the influence of active distances of convolutional code on bursts distribution and suggest a formula for estimation of the probability of error bursts depending on the length. We also present simulation results for bursts distribution for these two codes to show a difference between them and the dependence of probabilities on distance properties of convolutional code.

II. Active Distances for Convolutional Code

In this chapter, we consider convolutional codes with the rate $R = \frac{1}{2}$, give a description of trellis representation and define active distances for convolutional codes.

A. Convolutional codes

Let us briefly describe a binary, rate $R = \frac{1}{2}$ convolutional code of memory m. It is necessary to define the field F_2 consisting of elements $\{0, 1\}$. We will use term polynomial $f(D)$ over F_2:

$$f(D) = \sum_{i=0}^{l} f_i D^i,\ \ f_i \in F_2,\ \ f_l = 1,$$

where $l = \deg(f(D))$ is a degree of $f(D)$.

We consider convolutional codes in terms of a delay operator D or D-transformation. Thus, we can present an information sequence as:

$$\mathbf{u}(D) = \mathbf{u}_0 + \mathbf{u}_1 D + \mathbf{u}_2 D^2 + ...$$

and a code sequence as:

$$\mathbf{v}(D) = \mathbf{v}_0 + \mathbf{v}_1 D + \mathbf{v}_2 D^2 + ... \ ,$$

where $\mathbf{u}_i \in F_2$, $\mathbf{v}_i \in \{0, 1\}^2$, $0 \leq i \leq m$.

The relation between an informational sequence and a codeword can be established with a rational generator matrix $\mathbf{G}$:

$$\mathbf{v}(D) = \mathbf{u}(D)\mathbf{G}(D)$$

with a generator matrix

$$\mathbf{G}(D) = \begin{pmatrix} \mathbf{g}^{(1)} & \mathbf{g}^{(2)} \end{pmatrix} \ ,$$

where generator polynomial $\mathbf{g}^{(l)}(D) = g_0^{(l)} + g_1^{(l)} D + g_2^{(l)} D^2 + ... + g_m^{(l)} D^m$, $g_i^{(l)} \in F_2$, for $l = 1, 2$.

In this paper, we will consider only non-systematic convolutional codes in a systematic way with the rate $R = \frac{1}{2}$. In this case, the generator matrix $\mathbf{G}(D)$ looks as follows:

$$\mathbf{G}(D) = \begin{pmatrix} 1 & \frac{\mathbf{g}^{(2)}}{\mathbf{g}^{(1)}} \end{pmatrix} \ . \tag{1}$$

Example 1: Let $\mathbf{g}^{(1)} = 1 + D + D^3$, $\mathbf{g}^{(2)} = 1 + D + D^2 + D^3$, that can be also written in the octal form $\mathbf{g}^{(1)} = 13$, $\mathbf{g}^{(2)} =$

978-1-7281-1944-1/19/$31.00 ©2019 IEEE

17. Then the generator matrix of this code has the following structure:

$$\mathbf{G}(D) = \begin{pmatrix} 1 & \frac{1+D+D^2+D^3}{1+D+D^3} \end{pmatrix} = \begin{pmatrix} 1 & \frac{17}{13} \end{pmatrix} .$$

We also present an encoder for this code at Fig. 1. Hereinafter we can use the notation "code $\mathbf{G}(D) = (\mathbf{g}^{(1)}, \mathbf{g}^{(2)})$" implying that the generator matrix has form (1). It should be noted that we consider codes with generator polynomials that satisfy the condition that the greatest common divisor of generator polynomials is $\gcd(\mathbf{g}^{(1)}, \mathbf{g}^{(2)}) = 1$.

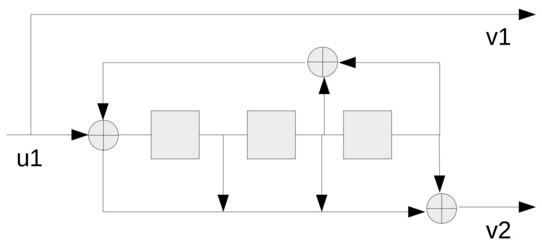

Figure 1. Encoder for systematic code $\mathbf{G}(D) = (13, 17)$

At Fig. 1 there are binary operators $\oplus$, registers and connections between them. The division by polynomial is implemented with the feedback structure. Codes with a recursive encoder are also referred to as the recursive systematic convolutional codes.

The important characteristic of the convolutional code is a memory m.

Definition 1: The maximum degree of generator polynomials of the convolutional code is called code memory:

$$m = \max_{l} \deg \left(\mathbf{g}^{(l)}\right) .$$

It is also a maximum register length, that means encoder output depends on m informational bits. For code $\mathbf{G}(D) = (13, 17)$ memory is $m = 3$. Code memory defines a number of bits which define the current register state $\mathbf{s}_t$ at moment t and it is represented with a sequence of m elements from F_2, then the number of possible states at moment $t > m - 1$ is 2^m. As code memory increases, code performance improves but decoding complexity grows exponentially.

B. Trellis Representation and Active Distances for convolutional codes

Convolutional codes can be described with trellis. Each input symbol $u \in F_2$ at each moment of time t corresponds to a transition to a specific state $\mathbf{s}_t = \left(s_t^{(1)}, s_t^{(2)}, ...s_t^{(m)}\right)$. There is a one-to-one correspondence between a sequence of input symbols and a sequence of register states. It is convenient to make a trellis in which the nodes correspond to register states and the edges are marked with output sequences of the encoder. For codes that we consider here output sequence consists of $n = 2$ bits and the first one is an informational symbol. It is agreed that the initial state of the encoder registers are considered zeros $\mathbf{s}_0 = (0, 0, ...0)$.

Example 2: For code $\mathbf{G}(D) = (1 + D + D^3, 1 + D + D^2 + D^3) = (13, 17)$ we present a trellis at Fig. 2 where the moments of time are plotted on the horizontal axis. The maximum number of states is $2^m = 8$, the first state is 000. The edges are marked with output sequences, where the first symbol is input one. You can see that for each input symbol there is only one transition with one output sequence. Thus, for an informational sequence there is only one trellis path. It should be noted that the up arrow means zero is saved in the first register, and the down arrow means one is saved.

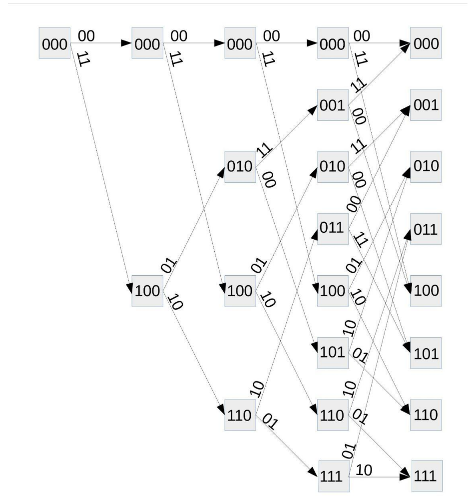

Figure 2. Trellis for systematic code $\mathbf{G}(D) = (13, 17)$

The informational sequence is generally semi-infinite, but in reality we are dealing with sequences of finite length. Finite sequences corresponds to finite trellis path. Hereinafter we consider codewords of finite length that corresponds to trellis path of finite length.

After describing trellis representation let us describe the distance properties of convolutional code.

Definition 2: Let the initial state at the moment of time $t > 0$ be zero: $\mathbf{s}_t = \mathbf{0}$. Then if there is a trellis path of length j that corresponds to some sequence $\mathbf{v} = (v_1, v_2, ...v_j)$, where v_i is tuple of length $n = 2$, $1 \leq i \leq j$, and does not have two consecutive zero states in between and if at moment $t + j$ the register state is $\mathbf{s}_{t+j} = \mathbf{0}$, then the active row distance of convolutional code is defined as follows:

$$a_j^r = \min_{\mathbf{v} \in C_j} \sum_{i=1}^{j} w_H(v_i) ,$$

where $C_j = \{\mathbf{v} | \mathbf{s}_t = \mathbf{s}_{t+j} = \mathbf{0}, \nexists k \in \{t, ...t + j - 1\} : \mathbf{s}_k = \mathbf{0} \cap \mathbf{s}_{k+1} = \mathbf{0}\}$, $w_H(v_i)$ is a Hamming weight of v_i .

Definition 3: Free distance of the convolutional code is a minimum of active row distances over all possible j:

$$d_{free} = \min_{j} a_j^r .$$

At Fig. 3 and Fig. 4 we present numeric results for active distances depending on the number of 2-tuples j (length of trellis path) for convolutional codes with same memory $m = 3$ but different active distances: code $\mathbf{G}(D) = (13, 17)$ and code $\mathbf{G}(D) = (13, 15)$ respectively.

III. Decoding and Analysis of output

In our simulation, we use a Viterbi algorithm for decoding. This algorithm provides maximum likelihood performance, the decoder receives a codeword as an input and gives the most probable trellis path corresponded to some informational sequence to an output. We have finite informational sequences

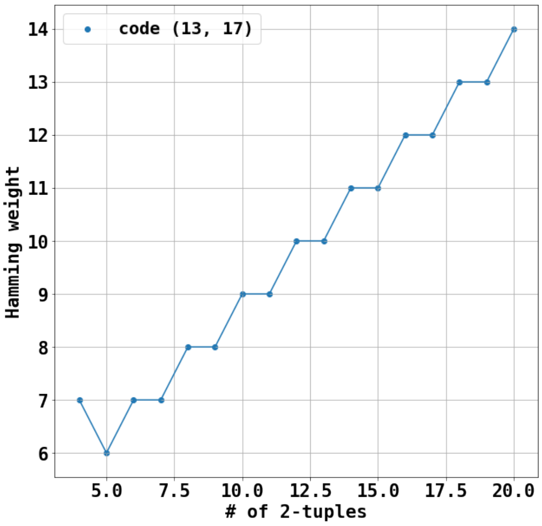

Figure 3. Active distances for systematic code $\mathbf{G}(D) = (13, 17)$

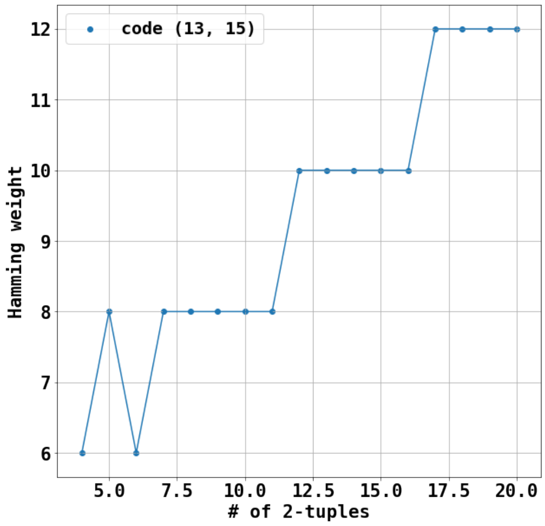

Figure 4. Active distances for systematic code $\mathbf{G}(D) = (13, 15)$

at the input and use zero-termination. That means we suppose input sequence of length N corresponds to some trellis path from the initial zero state. From any state registers can return into zero state after m input symbols. Thus, we consider sequences of register states of length $N+m+1$ or trellis path of length $N+m$ that begins and also ends at zero register state. If decoding is correct the sequences of states at the decoded path and at the path corresponded to the transmitted vector are the same.

We consider an output by Viterbi decoder that is a trellis path of length $N+m$. Then there is a sequence of bits of length $N+m$ saved to the first register while transferring from state to state. We suppose that the initial state is zero state that mean before the sequence there are m zero bits.

Definition 4: Let the code memory be m. In this sequence of bits the first incorrect bit after m correct is a beginning of the burst. If after an incorrect bit in the burst there is a sequence of correct bits of length at least m, then this incorrect bit is the last bit in this burst:

$$\ldots \underbrace{c \ldots c}_{m} \quad \underbrace{e \ldots e}_{\text{length of burst}} \quad \underbrace{c \ldots c}_{m} \ldots ,$$

where c - correct bit, e - burst bit. There can be sequences of correct bits in bursts but their length is less than m. After a sequence of m correct bits saved to the first register the register state is guaranteed to become correct.

The bursts define error trellis path that begins after the transition from the correct path (at least two consecutive correct states) to the first error state that means the beginning of burst. The last error state before the correct path ends error trellis path. The number of transition on error path is number of error states plus one. It defines length of error trellis path and can be written as number of 2-tuples.

In this paper, we consider the distribution of error trellis paths depending on their length after Viterbi decoding.

IV. Dependence distribution of error bursts of active distances

A. Derivation of the theoretical formula

Let us suppose that we have a recursive systematic convolutional code with rate $R = \frac{1}{2}$ and we have the value of active row distances a_j^r for different trellis path length j.

The sequence of input bits of length j corresponds to the trellis path (sequence of j states) and to the sequence of output tuples. Thus, it corresponds to the sequence of $nj = 2j$ bits. If we have correctly decoded sequence then after adding sent sequence to decoded one we will have a zero path(sequence of $j+1$ zero states). If we have the error trellis path then after adding sent sequence to the decoded one we will have a path with start and end at zero state corresponded to a sequence of output tuples of non-zero weight. The ones in the resulting sequence will be in the places of errors in the decoded sequence. According to the definitions of active row distance and error bursts, the minimum possible Hamming weight of obtained after adding sequence of output tuples will be a_j^r. In this way, since the code is linear, we can consider the transmission of a zero codeword without loss of generality. Error path in this codeword will start after zero state and end in a zero state.

We will provide you with the estimation of the probability of an error trellis path of length j. As we have mentioned earlier the minimum non-zero number of errors in bits in decoded sequence is a_j^r. There can be more errors in bits, but it is logical to assume that the most often number of errors that leads to the path of this length is the active distance a_j^r. Thus, we are going to give a lower bound on probability of error trellis path of certain length. We consider a sequence of $2j$ bits of Hamming weight a_j^r corresponded to the trellis path of length j with the start and the end at zero state. The probability of error in a bit while transmitting is p in our consideration.

The number of errors in the sequence of $2j$ bits i can be greater than a_j^r and up to $2j$. The number of errors in places of ones i_1 in the sequence greater than $\frac{a_j^r}{2}$ leads to a decoding into a sequence which is under consideration and there can be any number of errors in places of zeros in the sequence. The probability of i_1 errors between a_j^r ones in the sequence has the binomial distribution: $\binom{a_j^r}{i_1} p^{i_1}(1-p)^{a_j^r - i_1}$, and the probability of $i - i_1$ errors between $2j - a_j^r$ zeros has the same one: $\binom{a_j^r}{i_1} p^{i_1}(1-p)^{a_j^r - i_1}$. To sum up, in the sequence of length $2j$ the total number of errors i is greater than $\frac{a_j^r}{2}$ and can be up to $2j$ and the number of errors in ones i_1 is greater than $\frac{a_j^r}{2}$ and can be up to a_j^r but not greater than the total number of errors. We can write the estimation on probability of error trellis path of length j:

$$P(s=j) = \sum_{i>\frac{a_j^r}{2}}^{2j} \sum_{i_1>\frac{a_j^r}{2}}^{min(a_j^r,i)} \binom{a_j^r}{i_1} p^{i_1}(1-p)^{a_j^r-i_1} \times \\ \times \binom{2j-a_j^r}{i-i_1} p^{i-i_1}(1-p)^{2j-a_j^r-i+i_1} ,$$

where s is a random variable which defined length of an error trellis path.

The equation can be simplified and rewritten. This reasoning leads us to the following lemma.

Lemma 1. *The lower estimation on probability of the occurrence of error trellis path of length j is:*

$$P(s=j) = \qquad (2)$$

$$= \sum_{i>\frac{a_j^r}{2}}^{2j} p^i(1-p)^{2j-i} \sum_{i_1>\frac{a_j^r}{2}}^{min(a_j^r,i)} \binom{a_j^r}{i_1}\binom{2j-a_j^r}{i-i_1} , \qquad (3)$$

where s is a random variable which defined length of an error trellis path.

B. Simulation results

Here we present the simulation results of decoding the output of a binary symmetric channel with a transition probability p. In our simulation we transmitted and decoded $5 \cdot 10^6$ informational words of length 500. We considered two recursive systematic convolutional codes with $\mathbf{G}(D) = (13, 17)$ and $\mathbf{G}(D) = (13, 15)$ of memory $m = 3$ and free distance $d_{free} = 6$. For theoretical estimation we used the formula (3). The results are presented at Fig. (5) and Fig. (6) where we also give the estimation by the geometrical distribution proposed in [9]- [10]. These figures show us that theoretical estimation based on active distances much better approximate the probabilities of error bursts with small length and give us more information of distribution features. At Fig. (7) there are simulation results for error bursts distribution of these two codes and at Fig. (8) there are theoretical results. Thus, our theoretical formula describes well the behavior of bursts distribution of the convolutional code and can be used for comparing performances of the different codes.

The results of our experiments and calculations for these two codes are presented at tables (I) and (II). We use next abbreviations: k - length of error trellis path, a - active distance, d - minimum value of number of errors in 1, Th.Pr. - theoretical probability(from formula (3)), Sim.Pr. - simulation probability, Num. - number of error bursts. Simulation probability we consider as $Num/500/5000000$, cause length of information sequence was 500, number of sequences was 5000000.

Table I
SIMULATION AND THEORETICAL RESULTS FOR RECURSIVE SYSTEMATIC CODE $\mathbf{G}(D) = (13, 17)$

k	a	d	Th. Pr.	Sim. Pr.	Num.
4	7	4	2.64e-5	2.08e-5	51512
5	6	4	1.16e-5	1.12e-5	27639
6	7	4	2.64e-5	2.02e-5	50007
7	7	4	2.64e-5	2.17e-5	53414
8	8	5	1.26e-6	5.43e-6	13364
9	8	5	1.26e-6	3.27e-6	8037
10	9	5	2.77e-6	4.55e-6	11137
11	9	5	2.77e-6	4.96e-6	12116
12	10	6	1.38e-7	1.45e-6	3530
13	10	6	1.38e-7	1.06e-6	2593
14	11	6	2.96e-7	1.22e-6	2953
15	11	6	2.96e-7	1.08e-6	2616
16	12	7	1.52e-8	3.86e-7	934
17	12	7	1.52e-8	3.48e-7	841
18	13	7	3.2e-8	2.93e-7	707
19	13	7	3.2e-8	2.23e-7	537
20	14	8	1.68e-9	9.96e-8	239

Table II
SIMULATION AND THEORETICAL RESULTS FOR RECURSIVE SYSTEMATIC CODE $\mathbf{G}(D) = (13, 15)$

k	a	d	Th. Pr.	Sim. Pr.	Num.
4	6	4	1.16e-5	1.12e-5	27719
5	8	5	1.26e-6	8.48e-7	2098
6	6	4	1.16e-5	1.18e-5	29082
7	8	5	1.26e-6	1.56e-6	3855
8	8	5	1.26e-6	1.83e-6	4453
9	8	5	1.26e-6	1.99e-6	4889
10	8	5	1.26e-6	1.28e-6	3129
11	8	5	1.26e-6	1.39e-6	3393
12	10	6	1.38e-7	5.67e-7	1384
13	10	6	1.38e-7	4.77e-7	1162
14	10	6	1.38e-7	4.50e-7	1093
15	10	6	1.38e-7	2.98e-7	722
16	10	6	1.38e-7	2.27e-7	545
17	12	7	1.52e-8	1.37e-7	332
18	12	7	1.52e-8	1.03e-7	248
19	12	7	1.52e-8	8.23e-8	198
20	12	7	1.52e-8	6.75e-8	162

V. CONCLUSION

In this paper we propose a formula for the lower estimation of error bursts distribution at the output of Viterbi decoder based on knowing active row distances of the convolutional code. We present the results which show that our formula (3) gives accurate values for short error trellis path. Taking into account spectrum of code we can improve our lower bound. As error bursts of small length have greater probability it is important to accurately assess the likelihood of their occurrence. This theoretical estimation describes the probabilities of error bursts of small length much better than the geometrical distribution proposed in [9]- [10] and give us more information about bursts distribution pattern as our formula is based on the distance properties of code. It is also can be used for comparing the performance of different convolutional codes without simulation.

In the nearest future we are going to consider different probabilities of error in a bit in a channel and study the spectrum of code and the average weights of error bursts in order to improve our estimation.

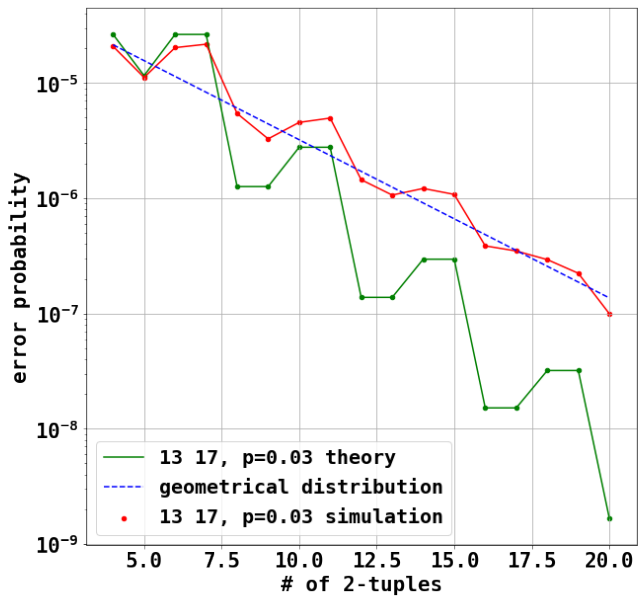

Figure 5. Distribution of error bursts in dependence of error trellis path length with $p = 0.03$ for systematic code $\mathbf{G}(D) = (13, 17)$

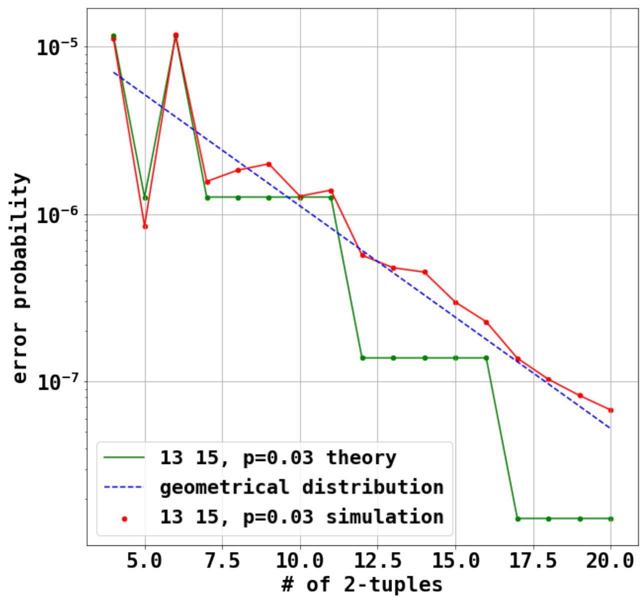

Figure 6. Distribution of error bursts in dependence of error trellis path length with $p = 0.03$ for systematic code $\mathbf{G}(D) = (13, 15)$

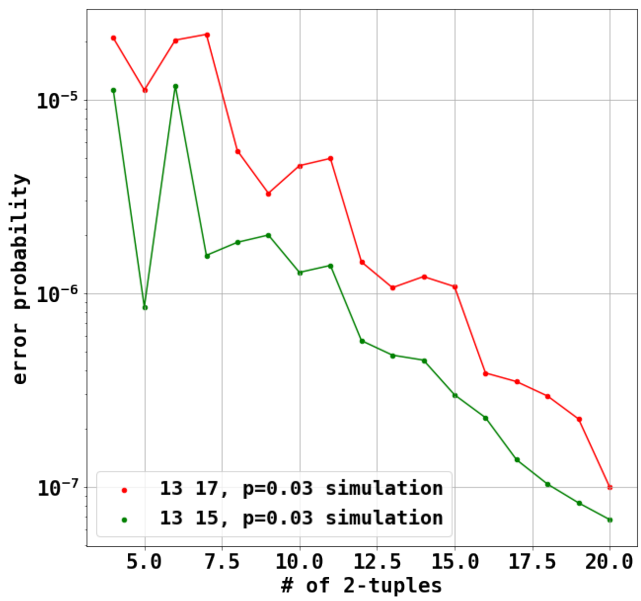

Figure 7. Distribution of error bursts in dependence of error trellis path length with $p = 0.03$ for two codes (simulation results)

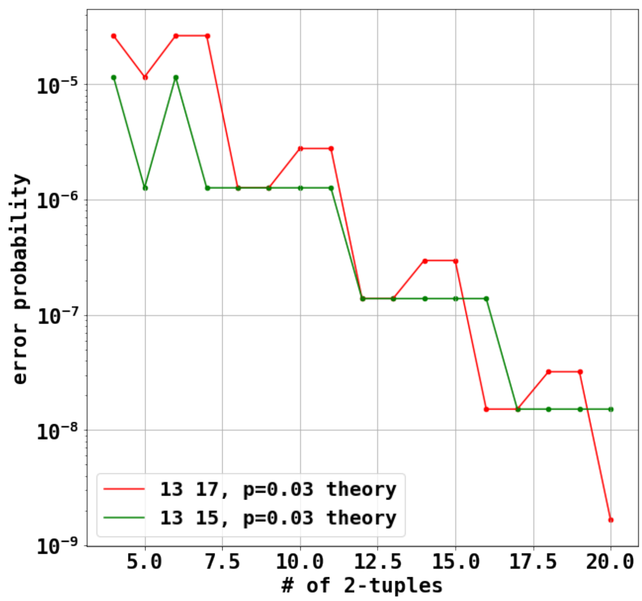

Figure 8. Distribution of error bursts in dependence of error trellis path length with $p = 0.03$ for two codes (theoretical results)

References

[1] D. J. Costello, Jr., “A construction technique for random-error-correcting convolutional codes,” IEEE Trans. Inform. Theory, vol. IT-19, pp. 631–636, 1969.

[2] G. Forney, ”Convolutional codes I: Algebraic structure”, IEEE Transactions on Information Theory (1970).

[3] C. Berrou, A. Glavieux, P. Thitimajshima, ”Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1”, Proceedings of ICC ’93 - IEEE International Conference on Communications (1993).

[4] S. Benedetto, G. Montorsi, ”Role of recursive convolutional codes in turbo codes”, IET, May 1995.

[5] G.D. Forney, ”The Viterbi algorithm”, Proceedings of the IEEE (1973).

[6] A. Viterbi, ”Convolutional Codes and Their Performance in Communication Systems”, IEEE Transactions on Communication Technology (1971).

[7] A. Viterbi, ”Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm,” IEEE Transactions on Information Theory, vol. 13, pp. 260-269, April 1967.

[8] Atsushi Yamashita, Tadayoshi Katoh, Masaru Moriwake, Tadashi Nakamura, ”Path trace Viterbi decoder”, US Patent (1987).

[9] R.L. Miller, L.J. Deutsch, S.A. Butman, ”On the Error Statistics of Viterbi Decoding and the Performance of Concatenated Codes,” NTRS (October 1981).

[10] Justesen, J., Andersen, J. D. (1998), ”Critical Lengths of Error Events in Convolutional Codes”, IEEE Transactions on Information Theory.

[11] Johannesson, R. and Zigangirov, K.Sh., ”Fundamentals of Convolutional Coding”, Piscataway: IEEE Press, 1999.

[12] Zyablov V. V., Johannesson R., Pavlushkov, V. A. (2004), ”Detecting and Correcting Capabilities of Convolutional Codes”, Problems of Information Transmission, 40(3), 187–194.

[13] Ivanov F., Kreshchuk A., Zyablov V., ”On the Local Erasure Correction Capacity of Convolutional Codes”, 2018 International Symposium on Information Theory and Its Applications (ISITA), P. 296-300.

[14] C. Thommesen and J. Justesen, “Bounds on distances and error exponents of unit-memory codes,” IEEE Transactions on Information Theory, vol. IT-29, pp. 637–645, Sept. 1983.

[15] Host S., Johannesson R., Zigangirov, K. S., Zyablov V. V., ”Active distances for convolutional codes,” IEEE Transactions on Information Theory (1999).