SYMBOL-WISE PROCESSING IMPLEMENTATION OF SEMI-ITERATIVE TURBO PRINCIPLE IN MULTI-HOP RELAY NETWORKS

Pavel Prochazka and Jan Sykora

Czech Technical University in Prague
Technicka 2 16627, Prague, The Czech Republic
phone: + 420 224 353 921,e-mail: prochp10@fel.cvut.cz, jan.sykora@fel.cvut.cz
web: http://radio.feld.cvut.cz/dirac/

ABSTRACT

The latency presents one of the major problem in communication systems. The seriousness increases in more complex physical scenarios, where the latency is accumulated in sequent nodes. We propose a novel concept of a *latency reduced* distributed iterative decoder design in a multi-node wireless network. The concept is based on the traditional turbo decoder, where we exploit the topology of the network to decompose the iterative decoder among more physically separated nodes. In this concept, the relay-part of the decoder computes symbol-wisely a soft metric, which is then transmitted through a wireless link into the destination-part of the decoder. The iterative process in the destination-part of the decoder has therefore only *limited capabilities for iterations* via the relay-part of the decoder. We formulate an optimization task for the proposed concept, design an initial (add-hoc) decoder decomposition and evaluate its properties. Since fundamental limits of the proposed optimization task are not available so far, we compare the proposed decomposition with the conventional turbo decoder.

1. INTRODUCTION

The turbo decoding introduced in [1] is a conventional way to achieve a near-capacity performance in a standard point-to-point communication channel. It is also known that the codeword achieving the near-capacity performance should be as long as possible [2]. The processing of the turbo decoder works iteratively with a whole sequence. It means that the decoder waits until the whole sequence is received and then the iterative processing (turbo-decoding) is started. This causes, however, two *sources of latency*: (1) time required for the receiving of the whole message and (2) time consumed by the iterative process.

Several works focused on the latency reduction in the iterative turbo-decoding (classical approach) by an improving (parallelization) of the iterative process to consume less time (e.g. [9]). In [4], the authors proposed an algorithm starting the iterative decoding (excluding an initialization) after obtaining one half the codeword sequence and thus they reduced the second source of the latency.

In contrast with the latency reducing approaches in the standard point-to-point communication channels, an another degree of freedom arises in more complex wireless network topologies, where auxiliary relays are employed to improve the communication. The relays are not receivers of the data. Their purpose is purely an auxiliary one. Several options of the processing in relays are available. A family of popular methods is based on a full decoding of the data in relays (Decode and Forward (DF)). The latency caused by the full decoding is then *accumulated in each relay*.

One can find several known latency-free strategies enabling the symbol-wise processing in relay (e.g. Amplify and Forward (AF) strategy). All of them are based on a metric, which is (symbol-wise) evaluated in relay and transmitted into the destination (e.g. amplified symbol for the AF). Nevertheless, these metrics suffer from a certain performance degradation against DF (except for same special cases).

A latency reducing concept usable in the relay channels based on the decoder decomposition was proposed in [5]. The authors introduced a Surviving Pattern (SP) as a metric given by a set of the surviving paths in one forward step of the forward Viterbi Algorithm (VA) . This metric is computable *symbol-wisely* in relay and it is sufficient for the destination-part of the decoder - backward VA (with knowledge of the final state). The method as it is presented in [5] is, however, restricted to convolutional codes.

In this paper, we formulate an intuitive optimization task for the proposed *latency-reduced* decoder decomposition when the conventional turbo coder is assumed to be employed as the channel coder. Our initial add-hoc approach is to extend the SP-based latency-free relay processing also to the turbo codes.

Although the idea beyond this paper is presented mainly on a single source multi-hop relays channel with one relay (see Fig. 1), it is designed mainly for a *multi-source* scenario with sharing resources of wireless communication systems (wireless network coding), where the multi-hop relay channel is used to model a virtual hierarchical data stream [7].

Contribution of the Paper

1. We propose a novel concept of the distributed turbo-like *latency-reduced* iterative decoder design, where the outer SISO block (connected to the observation) is split into two parts, which are not supposed to be collocated. Consequently a limited possibility of the iterative process is available, because the relay-part of the decoder is partly or absolutely excluded from the iterative process in destination.
2. We propose an initial decomposition based on an add-hoc selected relay-part SISO - forward VA (due to the latency-free request) and we investigate capabilities of some possibly resulting symbol-wise computable metrics.

EUSIPCO 2011, R2 (30.5.2011)
This work was supported by the FP7-ICT SAPHYRE project, by the Grant Agency of the Czech Republic, grant 102/09/1624, and by the Grant Agency of the Czech Technical University in Prague, grant No. SGS10/287/OHK3/3T/13.

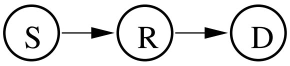


Figure 1: Multi-hop relay channel.

2. CONCEPT OF THE DECODER DECOMPOSITION

2.1 Motivation

An obvious goal of the design relay wireless networks is to have a decoding strategy achieving the capacity ideally without latency. Let us suppose a latency caused by the receiving of the sequence only in the final destination. The iterative decoder required for the near-capacity achievement can be situated partly in the destination and partly in relays[1]. The iterative process is, however, limited then, because the part of the decoder situated in the relay can interact with the rest of the decoder only through a wireless link (see Fig. 2(a)).

If we want to avoid the iterative loops through the wireless links, the relay-part of the decoder (block $\mathscr{F}_2^{-1}$ in Fig. 2) cannot interact with the iterative process in the destination (only a metric γ is once sent through a communication channel). It is thus reasonable to cast minimization requests to the entropy of γ in order to minimize the required energy in relay. Another reasonable requests to the relay-part of the decoder are to prepare the best possible convergence behaviour of the destination-part decoder. It means that the metric γ should contain as much extrinsic information $\mathrm{I}_{\mathrm{E}}^{\mathscr{F}_2^{-1}}$ as possible. Let us describe the mentioned optimization task more formally.

2.2 Optimization Targets of the Iterative Decoder Decomposition

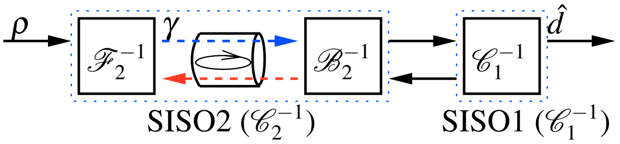
Our ultimate goal is to "appropriately" split the SISO2 block ($\mathscr{C}_2$) in the conventional turbo decoder (see Fig. 2) to two parts ($\mathscr{B}_2^{-1}$) and ($\mathscr{F}_2^{-1}$) jointly with design of the metric γ with respect to the following optimization criteria:

1. latency caused by the evaluation of γ minimization,
2. entropy $\mathrm{H}[\gamma]$ minimization,
3. extrinsic information $\mathrm{I}_{\mathrm{E}}^{\mathscr{F}_2^{-1}}$ (carried by γ in Fig. 2) maximization,
4. convergence process of the concatenated pair ($\mathscr{C}_1^{-1}$, $\mathscr{B}_2^{-1}$) optimization. It means a minimization of the area between the transfer functions and achieving [1,1] point in the EXIT chart [8].

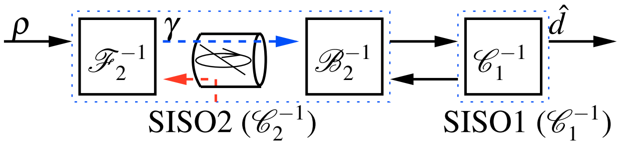
Note that within this paper, we assume an empty link $\mathscr{B}_2^{-1} \rightarrow \mathscr{F}_2^{-1}$ in order to avoid the iterative loops leading through the channel as discussed in Sec. 2.1 (see Fig. 2(b)).

3. SYSTEM MODEL

We define a system model to enable an investigation of the optimization tasks declared in Sec. 2.2 as separately as possible. Also a reference model is defined to make possible a clarification of the qualities of the proposed SISO decoder decomposition.



(a) Scheme of the decomposed decoder with *enabled* loops through the wireless channel.



(b) Proposed latency-reducing turbo-like decoder with *disabled* loops through the wireless channel.

Figure 2: Turbo decoder decomposition - basic concept. Block SISO2 is split into two parts ($\mathscr{F}_2^{-1}$) and ($\mathscr{B}_2^{-1}$). There is assumed a restriction (wireless link) on the links $\mathscr{B}_2^{-1} \rightarrow \mathscr{F}_2^{-1}$ and $\mathscr{F}_2^{-1} \rightarrow \mathscr{B}_2^{-1}$ (red and blue emphasis - Fig. 2(a)) . In an extreme case, the link $\mathscr{B}_2^{-1} \rightarrow \mathscr{F}_2^{-1}$ is omitted (this case is assumed within this paper - Fig. 2(b)). Note that if there is no restriction to the links between $\mathscr{F}_2^{-1}$ and $\mathscr{B}_2^{-1}$, the decoder stands for a conventional turbo decoder.

3.1 Investigated System Model

We assume a simplified multi-hop relay channel shown in Fig. 1, where the link from the relay to destination is assumed to be a perfect one. The whole model is shown in Fig. 3. Concrete parts of the model are described below.

3.1.1 *Source*

The source consists from two serially concatenated convolutional codes with interleaver and a signal space mapper. We assume an i.i.d. data vector $\mathbf{d}$ inputing the encoder $\mathscr{C}_1$ such as the codewords $\mathbf{c}_1 = \mathscr{C}_1(\mathbf{d})$. The codewords $\mathbf{c}_1$ input into an interleaver and the interleaved codewords are encoded by an encoder $\mathscr{C}_2$ such as $\mathbf{c}_2 = \mathscr{C}_2(\Pi(\mathbf{c}_1)) = \mathscr{C}(\mathbf{d})$, where we have denoted the whole encoder by $\mathscr{C}$ and Π denotes the interleaver. Finally the codeword vector $\mathbf{c}_2$ is modulated by a signal space mapper $\mathscr{M}$, i.e., $\mathbf{s} = \mathscr{M}(\mathbf{c}_2)$.

3.1.2 *Source-Relay Channel*

We assume a standard AWGN channel. The received vector is given by $\mathbf{x} = \mathbf{w} + \mathbf{s}$, where $\mathbf{x}$ is the received signal space vector into the relay and $\mathbf{w}$ is a complex zero-mean Gaussian vector with $\mathrm{var}(\mathbf{w}) = \mathbf{I}\sigma$. We have denoted the unit matrix by $\mathbf{I}$.

As it was said, the relay-destination channel is assumed to be the perfect one, i.e., $\mathbf{x} = \mathbf{s}_R$. This simplification enables us to separately investigate points 3 and 4 in Sec. 2.2.

3.1.3 *Relay Processing*

First, a standard Soft Output Demodulator (SODEM) is applied $\rho = \mathscr{D}(\mathbf{x}_R)$. Then the relay-part of the SISO block $\mathscr{F}_2^{-1}$ is applied to obtain the metric $\gamma = \mathscr{F}^{-1}(\rho)$.

3.1.4 *Destination Processing*

The decoder is implemented by the Sum Product Algorithm on Factor Graphs (FG-SPA) [3]. Unusual parts of the FG-

---

[1] Note that the conventional AF strategy has situated the iterative decoder only in the final destination.

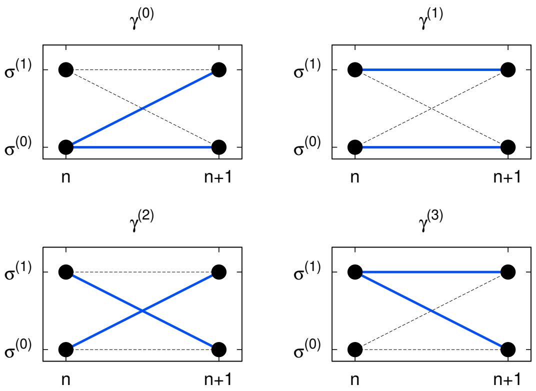


Figure 4: Set of all possible SP for a FSM with two states and two input data symbols.

SPA $\mathscr{B}_2^{-1}$ will be described later. We assume the conventional forward-backward scheduling algorithm between blocks $\mathscr{B}_2^{-1}$ and $\mathscr{C}^{-1}$.

3.2 Reference System Model

In order to measure the properties of the proposed method, we need a reference model for a comparison. This reference model is equal with the model described in Sec. 3.1, where we do not cast any restriction to the iterative process, so that it is a conventional turbo decoder.

4. INITIAL SOLUTION OF THE DECOMPOSITION TASK AND ITS PROPERTIES

Due to the complexity of the optimization problem mentioned in Sec. 2.2, we set up an initial decoder decomposition and investigate its properties.

4.1 Investigated System Model Population - Concrete Blocks Assignment for the Decoder Decomposition

Based on the results of [5], we *add-hoc* decide to assign the forward VA to the block $\mathscr{F}_2^{-1}$. We further assume a *symbol-wise* processing in relay. It means that the metric γ must be computable symbol-wisely (due to the latency-free request to the design).

The forward VA offers two kinds of metrics, which are computable in a symbol-wise manner. First the mentioned discrete valued SP and second the optimization metric underlaying a decision about the surviving paths and consequently about the SP. Since the SP is a sufficient information for the backward VA, we set (again add-hoc) the metric γ to be formed by the SP.

The decomposition of the decoder according to Sec. 3.1 can be summarized as follows in our approach:

1. the symbol-wise processing of the block $\mathscr{F}_2^{-1}$ (assumption),
2. the block $\mathscr{F}_2^{-1}$ given by the forward part of the VA (add-hoc choice),
3. metric γ given by the SP (add-hoc choice).

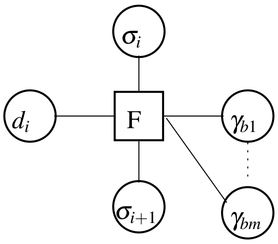


Figure 5: The factor node of the destination-part SISO $\mathscr{B}_2^{-1}$.

4.2 FG-SPA split SISO Block Design and the Decoder Implementation

We have already prepared and described the whole model excepting the decoder. Now, we describe the structure of the decoder and its implementation using FG-SPA. We start with the description of the SP (see Fig. 4 for an illustration). Although this description is referred to the decoder $\mathscr{C}_2^{-1}$ in our system model (Sec. 3.1), we utilize the SP implementation for a general Finite State Machine (FSM).

4.2.1 *Surviving Pattern Description*

We denote the state equation of the described FSM by $\sigma_{n+1} = \Sigma(\sigma_n, d_n)$, where we denoted the matrix Σ defining the state equation. One surviving path in the forward VA is described unambiguously by a tuple (σ_n, d_n). The surviving pattern is given by all used surviving paths in one forward step of the VA (see Fig. 4) and it is described by several tuples (σ_n, d_n), which can be interpreted as a concrete set of positions in Σ matrix.

We index the set of all possible tuples (σ_n, d_n) resulting into the state $\Sigma(\sigma_n, d_n) = \sigma_{n+1} = i$ by γ_{bi}. Since one surviving path leads into each state σ_{n+1}, the SP can be described unambiguously by a vector $\gamma_b = [\gamma_{b1}, \ldots, \gamma_{bn}]$. For example, if the matrix Σ and mapping of γ_b are given by

$$\Sigma = \left[\begin{array}{cccc} 0^{\gamma_{b1}=0} & 0^{\gamma_{b1}=1} & 1^{\gamma_{b2}=0} & 1^{\gamma_{b2}=1} \\ 2^{\gamma_{b3}=0} & 2^{\gamma_{b3}=1} & 3^{\gamma_{b4}=0} & 3^{\gamma_{b4}=1} \end{array} \right],$$

the vector $\gamma_b = [0\,0\,1\,1]$ refers unambiguously to the pattern described by the tuples (σ_n, d_n) - $(0,0), (2,0), (1,1)$ and $(3,1)$.

4.2.2 *Update Rules of the Factor Node Working with the Surviving Pattern*

Each factor node in FG-SPA is described by a conditional probability function. We describe the conditional probability mass function (pmf) of the factor node referring to the SP (see Fig. 5) by

$$p(\gamma_b, \sigma_{n+1} | d_n, \sigma_n) = F(d_n, \sigma_n, \sigma_{n+1}, \gamma_b), \tag{1}$$

where $F(d_n, \sigma_n, \sigma_{n+1}, \gamma_b)$ indicates if both the tuple (σ_n, d_n) refers to the index γ_{bi} and $\Sigma(\sigma_n, d_n) = \sigma_{n+1} = i$. This update rule can be further improved by knowledge of the distribution $p(\gamma_b)$ resulting from the forward VA.

4.2.3 *Surviving Pattern Based SISO Functional Block*

Based on the conditional pmf (1), we are able to design the FG-SPA operating with the SP. The FG-SPA serves as the

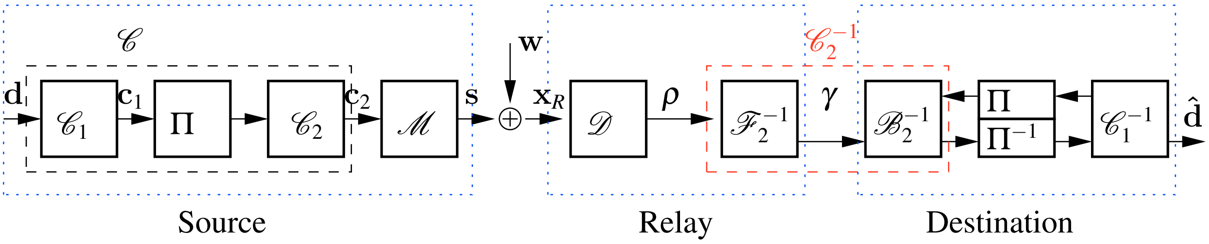


Figure 3: System model for measurement of the information-loss caused by the use of the proposed method (Sec. 3.1). We study an impact of the corruption caused by the decomposition of the SISO block $\mathscr{C}_2$ to $\mathscr{F}_2$ and $\mathscr{B}_2$ with enabled iterative processing only between blocks $\mathscr{B}_2^{-1}$ and $\mathscr{C}_1^{-1}$.

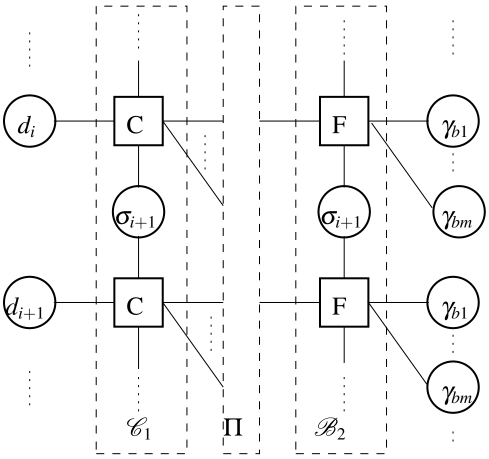


Figure 6: The Factor graph of the decoder used in the proposed method. The SISO block $\mathscr{C}_1$ refers to the convolutional encoder $\mathscr{C}_1$, the SISO block $\mathscr{B}_2$ refers to the destination part of the split decoder $\mathscr{C}_2$ (described in Sec. 4.2.3) and Π is the interleaver.

SISO block $\mathscr{B}_2^{-1}$ implementation. The designed $\mathscr{B}_2^{-1}$ SISO block replaces the standard SISO block $\mathscr{C}_2$ used in the conventional turbo decoder and the iterative detection is thus enabled.

4.2.4 *FG-SPA Description of the Decoder*

The factor graph of the decoder can be seen in Fig. 6. It is in fact a conventional FG-SPA implementation of the turbo decoder [3], where the SISO block $\mathscr{C}_2$ is replaced by $\mathscr{B}_2^{-1}$ described in Sec. 4.2.3.

4.3 Performance Analysis

We have implemented the model according to Fig. 3. We have used the following concrete parameters.

4.3.1 *Simulation Setup*

The data sequence contains $N = 256$ symbols. The encoder $\mathscr{C}_1$ is selected to be 1/2-rate $[75]_o$ convolutional code, where the o-subscript denotes the octal notation. The interleaver is constructed as the s-interleaver with polynomial $15x + 16x^2$, where $x \in \mathbb{Z}_{256}$ according to [6]. The encoder $\mathscr{C}_2$ is

1. the simplest possible 1/2-rate systematic recursive convolutional encoder with generator $G = [1; D/(1+D)]$ (the encoder is referred as C_A).

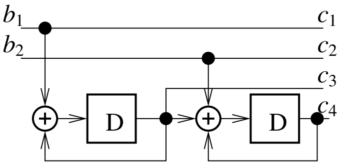


Figure 7: The recursive convolutional code composed from two grouped C_A codes.

2. a ”super-encoder” composed from two encoders[2] C_A. The encoder is shown in Fig. 7 and it is denoted by C_B.

The mapper $\mathscr{M}$ is the well known BPSK mapping. The soft output demodulator is based on the minimal Euclidean distance.

4.3.2 *Results*

The error-rate performance of the proposed *latency-free* method based on the SP as the metric γ with C_A encoder as $\mathscr{C}_2$ is shown in Fig. 8. The performance is compared with the performance of the conventional turbo decoder without any restrictions on the iterative process. One can see that the performance loss caused by the restrictions on iterations is about 5 dB for the SP metric and the forward VA as the relay-part decoder.

One can offer an idea that a more complex encoder $\mathscr{F}_2$ can improve the performance. Observing Fig. 9, we can state that the error-performance gain caused by the use of the more complex encoder C_B instead of the simple encoder C_A is about 1 dB. Nevertheless, this gain is paid by a considerable increased entropy rate as one can see in Fig. 10.

It seems that the use of the add-hoc chosen pure SP as γ metric corrupts the performance (by 4-5 dB in our models). These initial results point us to some further improvements how to rectify the metric γ (some possibilities are discussed later). proposed metric offers a latency-free strategy.

5. CONCLUSIONS AND DISCUSSIONS

We have proposed a novel concept of a *reduced latency* distributive iterative decoding in a wireless network based on a decomposition of one SISO block, where the iterative process can generally work through wireless links. We have also formulated some optimization tasks to the proposed iterative

---

[2] Note that we slightly violate the latency-free assumption casted in Sec. 4.1, because this encoder in fact works with two data symbols in relay.

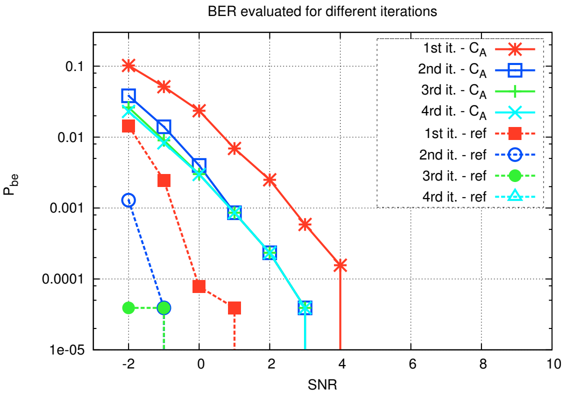


Figure 8: The relation of the BER behaviour to iteration for the proposed *latency-free* method based on the SP as the metric γ with recursive encoder C_A compared to the standard turbo decoder.

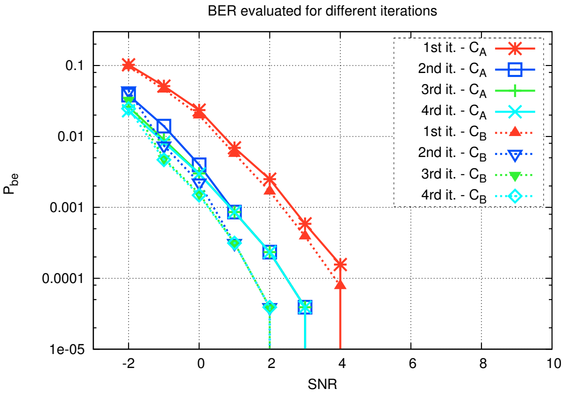


Figure 9: The relation of the BER behaviour to iteration for the proposed method based on the SP as the metric γ with recursive encoder C_B compared to the encoder C_A. There is approx. 1 dB improvement of the performance with the encoder C_B against C_A.

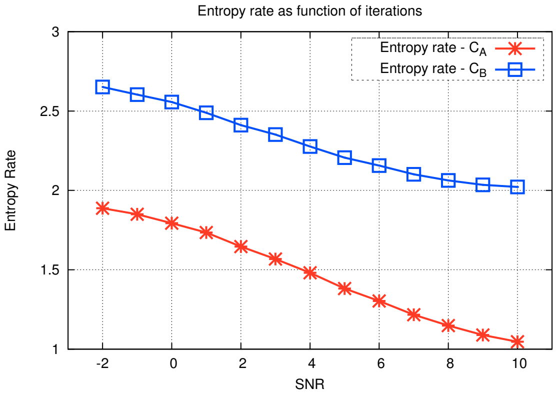


Figure 10: The entropy rate of the SP referred to the recursive encoders C_A and C_B recursive encoder.

decoding (Sec. 2.2) resulting from intuitive requests mentioned in Sec. 2.1.

Within this paper, we have studied only the simplest possible (but practical) case, where we have disabled the iterative loops through the wireless channel (Fig. 2(b)). It should be noted that the analysis suffers from an absence of the fundamental limits related to Sec. 2.2 and thus it is difficult to classify the results.

We have (add-hoc) provided an initial decoder decomposition and investigated its performance. The obtained error-rate performance is shown in Figs. 8 and 9 and related entropy of the metric carried through the channel is shown in Fig. 10. One can see that the proposed *latency-free* semi-iterative turbo-like decoder converges (the iterative process in destination evolves), but it suffers from a performance-degradation, which is caused by the restriction on the information exchange within the split SISO block.

Finally, observing Fig. 9 we have slightly improved the error-rate performance by grouping of two simple FSM to a more complex one, nevertheless the improvement of the error-rate performance was at the expanse of the increased SP-entropy (Fig. 10).

Future Work

- Determine the fundamental limits related to Sec. 2.2.
- Systematically compose γ to optimize requests declared in Sec. 2.2.

REFERENCES

[1] Claude Berrou and Alain Glavieux. Near optimum error correcting coding and decoding: Turbo-codes. *IEEE Trans. Commun.*, 44(10):1261–1271, October 1996.

[2] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 1991.

[3] F. Kschischang, B. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. Inf. Theory*, 47(2):498–519, 2001.

[4] Ya Cheng Lu, Tso Cho Chen, and Erl Huei Lu. Low-latency turbo decoder design by concurrent decoding of component codes. In *Innovative Computing Information and Control, 2008. ICICIC '08. 3rd International Conference on*, page 533, June 2008.

[5] Pavel Prochazka and Jan Sykora. Per-symbol representation of sufficient statistics for FSM decoder metric used in BC phase of 2WRC with HDF strategy. In *COST 2100 MCM*, pages 1–8, Athens, Greece, February 2010. TD-10-10087.

[6] Jing Sun and Oscar Y. Takeshita. Interleavers for turbo codes using permutation polynomials over integer rings. *IEEE Trans. Inf. Theory*, 51(1):101–119, 2005.

[7] Jan Sykora and Alister Burr. Layered design of hierarchical exclusive codebook and its capacity regions for HDF strategy in parametric wireless 2-WRC. *Accepted in IEEE Trans. Veh. Technol.*, 2011.

[8] Stephan ten Brink. Convergence begavior of iteratively decoded parallel concatenated codes. *IEEE Trans. Commun.*, 49(10):1727–1737, October 2001.

[9] Seokhyun Yoon and Y. Bar-Ness. A parallel map algorithm for low latency turbo decoding. *Communications Letters, IEEE*, 6(7):288 –290, July 2002.