Symbol-Level Synchronization and LDPC Code Design for Insertion/Deletion Channels

Feng Wang, Dario Fertonani, and Tolga M. Duman

***Abstract*—We investigate a promising coding scheme over channels impaired by insertion, deletion, and substitution errors, i.e., interleaved concatenation of an outer low-density parity-check (LDPC) code with error-correction capabilities and an inner marker code for synchronization purposes. To limit the decoding latency, we start with a single-pass decoding algorithm, that is, marker code-based synchronization is performed only once per received packet and iterative decoding with information exchange between the inner decoder and outer decoder is not allowed. Through numerical evaluations, we first find the marker code structures which offer the ultimate achievable rate when standard bit-level synchronization is performed. Then, to exploit the correlations in the likelihoods corresponding to different transmitted bits, we introduce a novel symbol-level synchronization algorithm that works on groups of consecutive bits, and show how it improves the achievable rate along with the error rate performance by capturing part of the rate loss due to interleaving. When decoding latency is not an issue and multiple-pass decoding is performed, we utilize extrinsic information transfer (EXIT) charts to analyze the convergence behavior of the receiver, which leads to design of outer LDPC codes with good degree distributions. Finally, design examples are provided along with simulation results which confirm the advantage of the newly designed codes over the ones optimized for the standard additive white Gaussian noise (AWGN) channels, especially for channels with severe synchronization problems.**

***Index Terms*—Insertion/deletion channel, marker codes, synchronization, LDPC code design.**

I. Introduction

CHANNELS with synchronization errors due to a mismatch of the transmitter and receiver clocks are of great interest in practical systems, e.g., in bit-patterned media recoding systems [1]. Despite their importance, channels with such impairments are far from being fully understood and proved to be difficult to analyze mainly because of the unknown positions of random insertions and deletions in the received data sequence. For insertion/deletion channels, the information stability and the Shannon's capacity theorem were proved to hold in [2]. However, a finite-letter expression of the channel capacity does not exist and only upper/lower bounds on the channel capacity are available in the current literature, which are not tight for almost any range of insertion/deletion probabilities [3]–[5].

The difficulty in the study of insertion/deletion channels is also confirmed by the lack of channel codes able to provide reliable communications at rates close to the capacity lower bounds [3]. An early approach for identifying synchronization errors is the use of markers referring to the insertion of known bits to prespecified positions into the transmitted stream [6], so that synchronization can be re-gained by locating the positions of the markers in the received sequence. Apart from these, there are three basic approaches toward designing codes for channels with synchronization errors: algebraic/number theoretic code designs, convolutional (trellis-based) codes, and channel codes based on code concatenation. A comprehensive survey of works on single-deletion-correcting codes is provided in [7] primarily focusing on the algebraic/number theoretic approaches. It becomes evident that even the restriction on the number of possible insertions/deletions to be only one over a codeword does not offer simple solutions. As another example, more recent work [8] starts from single insertion/deletion correcting codes and constructs codes for segmented insertion or deletion channels which allow for a simple left-to-right decoding algorithm. Codes with multiple insertion and/or deletion correcting capabilities are described in [9]. Other works on algebraic codes include the use of Reed-Muller $(1, m)$ code over channels with synchronization and substitution errors; however, the channel model only permits either repetition or deletion of a single bit. Non-binary codes over deletion channels, e.g., Reed-Solomon (RS) codes, are studied in [10], where for $l \leq 36$, RS codes capable of correcting up to $l-3$ deletions are listed. Also, algebraic designs employing cyclic codes are reported in [11], [12]. As an example of the second approach, i.e., convolutional coding over insertion/deletion channels, we can cite pruned convolutional codes in [13]. [14] considers convolutional codes with a long buffer and bit reversal before transmission. In [15], new states in the trellis of a convolutional code are added and a special code construction algorithm is introduced which maximizes the minimum Levenshtein distance [16] between different codewords. Idea of adopting parallel Viterbi decoders to correct insertion, deletion and flipping errors is investigated in [17], where the key idea is to ensure that the decoding procedure starts from the correctly synchronized decoder.

To date, coding schemes with the most promising performance reported over insertion/deletion channels are based on code concatenation. The key idea is to concatenate, through an interleaver, an outer code with good error-correction capabilities with an inner code whose aim is to help the receiver detect the synchronization errors due to the presence of insertions/deletions. For example, [18] considers concatenation of Reed-Solomon codes as outer codes with an inner code designed using a brute force approach, and shows that these are asymptotically good for channels with insertions and

Paper approved by O. Milenkovic, the Editor for Coding Theory and Applications of the IEEE Communications Society. Manuscript received September 7, 2010; no revision.

The authors are with the School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85287-5706, USA (e-mail: Feng.Wang83@asu.edu, dario.fertonani@gmail.com, duman@asu.edu).

This work is funded by the National Science Foundation, under the contract NSF-TF 0830611. Part of the work was presented at the IEEE International Symposium on Information Theory, June 2010.

Digital Object Identifier 10.1109/TCOMM.2011.030411.100546

deletions in the sense that the code rate remains positive. Concatenation of outer low-density parity check (LDPC) codes and inner watermark codes are investigated in [19], while concatenation of LDPC codes and marker codes are investigated in [20], [21]. Optimal maximum-a-posteriori (MAP) marker/watermark code-based synchronization can be performed by means of the forward-backward algorithm presented in [19], [22], while a sub-optimal decoding strategy for concatenation of an LDPC code, a Varshamov-Tenengolts (VT) code and a marker code is considered in [23].

In this paper, we consider binary channels impaired by independent and identically distributed (IID) insertion, deletion, and substitution errors, whose positions are unknown to either the transmitter or the receiver. Particularly, we focus on the channel model adopted in [24], which effectively describes many communication processes impaired by synchronization errors. According to this model, each transmitted bit may be deleted, replaced by two random bits, substituted by its complement or correctly received independently of each other. As in [21], we consider the interleaved serial concatenation of an outer error-correcting code with an inner marker code. To limit the decoding latency, we assume that marker code-based synchronization is performed only once per received packet, i.e., iterations with the outer decoder are not allowed. Our first contribution consists of the evaluation of ultimate outer code rates at which reliable communications (in the Shannon sense [25]) is possible, for a given channel and a given marker code. An approximate solution of this problem was proposed in [21], where the authors characterize the capacity of a proper time-varying binary symmetric channel (BSC) and conjecture that it gives an accurate approximation of the actual achievable rate. In this paper, we consider the exact solution to the problem, based on mutual information arguments, numerically evaluate the information rates, and show how to exploit these achievable rate analyses to optimize the marker code. As an algorithm for marker code-based synchronization, we first consider the standard MAP detector working at the bit level [19], [22]. Then, we introduce the MAP detection at the symbol level, defining a symbol as a group of m consecutive bits, and show how this approach can improve the achievable rate without changing the transmitter structure.

Besides the achievable rate analysis with a single pass decoder, we also investigate the outer LDPC code design when multiple-pass decoding is used, i.e., synchronization is performed multiple times for each received packet by iterating between the decoder for the outer code and the synchronization module. The goal of our design is to find "good" outer LDPC codes concatenated with marker codes for transmission over the insertion/deletion channels which offer better decoding performance than the ones optimized for AWGN channels by varying the variable/check node degree distributions. The motivation is explained as follows: since the detection extrinsic information transfer (EXIT) chart [26] is not flat for the channel model under consideration, LDPC codes designed for AWGN channels are no longer optimal when iterative decoding is performed [27]. Many optimization schemes for LDPC codes over different types of channels have been reported in the literature, e.g., optimized LDPC code for partial response channels by means of density evolution analysis [28], EXIT chart-based code design for multiple-input multiple-output (MIMO) fading channels [26] and ISI channels [29]. In this paper, we choose to use the EXIT charts to analyze the impact of insertions and deletions on the convergence behavior of the receiver and find good variable/check node degree distributions.

The paper is organized as follows. In the next section, the system model is described. In Section III, we review the standard bit-level MAP detection algorithm and numerically evaluate the ultimate rate achievable by interleaved concatenated coding schemes. In Section IV, we introduce the symbol-level MAP detection algorithm and compare the relevant achievable rates with those characterizing the standard bit-level approach. Error-rate results for a practical LDPC-coded scheme are also reported for both bit-level and symbol-level detection. The EXIT chart-based LDPC code design process for the insertion and deletion channels is provided in Section V along with example designs. Finally, concluding remarks are given in Section VI.

II. System Description

We consider transmission over binary channels impaired by insertion, deletion, and substitution errors, according to the model proposed in [24]. Let $\mathbf{x}_1^T = \{x_k\}_{k=1}^T$ and $\mathbf{y}_1^R = \{y_n\}_{n=1}^R$ be the sequences of bits at the channel input and channel output, respectively, where the number of transmitted bits T is a constant system parameter while the number of received bits R is a random variable depending on the realization of the insertion/deletion process. We can think of the channel as the cascade of two subchannels where, in the first subchannel, each input bit gets deleted (with probability P_d), or experiences an insertion error (with probability P_i), or is correctly transmitted (with probability $P_t = 1 - P_d - P_i$), while the second subchannel is a BSC with substitution probability P_s.[1] As proposed in [24], an input bit experiencing an insertion error gets replaced by two uniformly distributed random bits — we point out that different models for the insertion process exist (for example, see [19], [30]). We assume that insertion, deletion, and substitution errors are all IID, and that the transmitter and the receiver have no information on the positions at which the errors occur.

We adopt the coding scheme depicted in Fig. 1, which consists of the interleaved serial concatenation of an outer error-correcting code with an inner marker code. Specifically, the information bits are first encoded by means of a powerful channel code (e.g., a turbo or LDPC code), then the transmitted sequence is formed by inserting pilot bits, which are often referred to as markers, to the interleaved sequence of coded bits. The marker bits and their positions in the transmitted sequence are known to the receiver, which exploits this information in the MAP detector to recover the synchronization errors due to insertions/deletions, as explained later. For simplicity, we only focus on the case of regular

[1]The analysis and the results presented in this paper can be easily extended to the case when the second subchannel is an arbitrary memoryless channel (e.g., an additive white Gaussian noise channel).

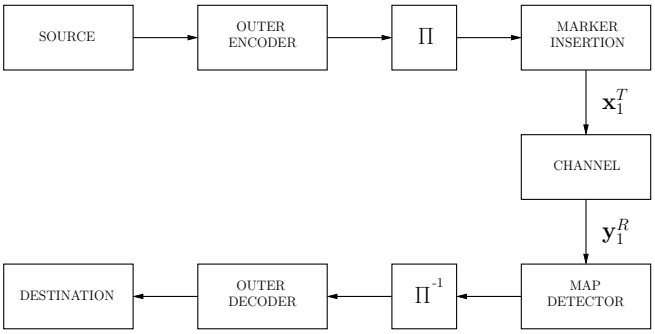

Fig. 1. Block diagram of the considered concatenated coding scheme. Interleaving and deinterleaving blocks are denoted by Π and Π^{-1}, respectively.

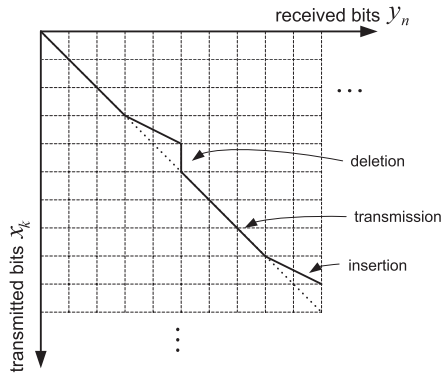

Fig. 2. Synchronization represented by a path on a two dimensional grid.

marker codes with rate

$$r_M = \frac{N_C}{N_C + N_M} ,$$

i.e., the case when the same marker consisting of N_M consecutive bits is inserted every N_C bits at the output of the outer encoder. Hence, if the outer code rate is denoted by r_C, the overall code rate is $r = r_C r_M$. We notice that the same coding scheme was considered in [21], while a similar scheme adopting watermark codes instead of marker codes was considered in [19].

At the receiver side, given the *a priori* log-likelihood ratios (L-values) $\log \frac{P(x_k=0)}{P(x_k=1)}$, the MAP detection is first executed to generate the conditional probability $\xi_k(x_k) = P(\mathbf{y}_1^R|x_k)$ for $k \in \{1, 2, \ldots, T\}$ and $x_k \in \{0, 1\}$ by exploiting the perfect *a priori* information from the marker code. Then the extrinsic information [31] on the transmitted bits can be easily obtained as $\log \frac{P(\mathbf{y}_1^R|x_k=0)}{P(\mathbf{y}_1^R|x_k=1)} = \log \frac{\xi_k(0)}{\xi_k(1)}$. After being deinterleaved, the *a posteriori* information, i.e., the sum of *a priori* and extrinsic L-values, feeds the outer decoder, which finally generates an estimate of the information bits. We point out that decoding performance can be improved by adopting iterative schemes based on the exchange of extrinsic information between the MAP detector and the outer decoder. But, since the MAP detector is typically the bottleneck of the receiver in terms of latency, we assume that the MAP detection is executed only once in Sections III and IV. Iterative detection/decoding is considered in Section V where specific outer code designs are pursued.

III. Bit-Level Synchronization

Let us first review the bit-level MAP detection algorithm for the considered channel model. The algorithm, which already appeared in [19], [22] with some differences in the channel model, is similar to the general forward backward algorithm (FBA) [32], but it cannot be derived by means of the standard approach discussed in [32] because the channel model is not a finite-state Markov chain [19], [22]. According to the turbo principle [32], the code constraints induced by the outer code are neglected in the derivation of the algorithm, and the bits $\mathbf{x}_1^T$ are considered to be statistically independent, namely the *a priori* probability $P(\mathbf{x}_1^T)$ is factorized as $\prod_{k=1}^{T} P(x_k)$, where $P(x_k)$ is $1/2$ if x_k is a code bit, while it is 0 (or 1) if x_k is a pilot bit.

A. Bit Level MAP Detection

Let us define the binary event $D_{k,n}$, with $k \in \{1, 2, \ldots, T\}$, and $n \in \{0, 1, \ldots, R\}$, which denotes whether, of the first k transmitted bits, exactly n bits are received, possibly after being corrupted by the channel or not. We are interested in the exact "frame synchronization" scenario, in which $D_{0,0}$ and $D_{T,R}$ are true with probability one, the values of T and R being known to the receiver. This assumption is not critical since frame synchronization can be obtained with great accuracy in practice [19]. For a better illustration of the resynchronization process, a two-dimensional grid is created to represent the synchronization errors. As shown in Fig. 2, the rows and columns on the grid correspond to the transmitted and received bits $x_k,\ k \in \{1, \ldots, T\}$ and $y_n,\ n \in \{1, \ldots, R\}$, respectively. The solid line refers to a particular channel realization and the dotted line indicates the channel without any insertion or deletion errors. There are only three possible moves to reach a certain state. A diagonal move from the top left corner to the bottom right corner on the grid indicates a successful transmission, i.e., no insertion or deletion, but the bit may not be correctly received. An insertion event is represented by a diagonal move in two adjacent blocks and a vertical move denotes a deletion event. Let us also define the function

$$F(x_k, y_n) = \begin{cases} 1 - P_s & \text{if } y_n = x_k \\ P_s & \text{if } y_n \neq x_k \end{cases} , \tag{1}$$

and the coefficients

$$\alpha_k(n) = P(\mathbf{y}_1^n, D_{k,n}) , \tag{2}$$

$$\beta_k(n) = P(\mathbf{y}_{n+1}^R | D_{k,n}) . \tag{3}$$

These coefficients can be computed by means of the following forward recursion (where the differences with respect to [19], [22] are due to the adopted channel model being different):

$$\alpha_k(n) = \frac{P_i}{4}\,\alpha_{k-1}(n-2) + P_d\,\alpha_{k-1}(n) + P_t\,\alpha_{k-1}(n-1)\sum_{x_k} P(x_k) F(x_k, y_n) , \tag{4}$$

and the following backward recursion [19], [22]:

$$\beta_k(n) = \frac{P_i}{4}\,\beta_{k+1}(n+2) + P_d\,\beta_{k+1}(n) + P_t\,\beta_{k+1}(n+1)\sum_{x_{k+1}} P(x_{k+1}) F(x_{k+1}, y_{n+1}) , \tag{5}$$

which are both initialized by exploiting the "frame synchronization" assumption. Finally, the target conditional probability can be computed as

$$p(\mathbf{y}_1^R|x_k) = \frac{P_i}{4}\sum_{n=0}^{\min(2k,R)}\alpha_{k-1}(n-2)\beta_k(n) + P_d\sum_{n=0}^{\min(2k,R)}\alpha_{k-1}(n)\beta_k(n) + P_t\sum_{n=0}^{\min(2k,R)}\alpha_{k-1}(n-1)\beta_k(n)F(x_k,y_n)\,. \quad (6)$$

B. Achievable Rates by a Specific Marker Code

An interesting information-theoretic problem that arises is the following: what is the ultimate rate at which we can reliably (in the Shannon sense [25]) transmit information through the considered concatenated coding scheme? An approximate solution to this problem can be found in [21], where the authors investigate the BCJR-once bound [33] and characterize the capacity of a BSC with a time-varying substitution probability and conjecture that it provides an accurate characterization of the information rate. Here, we pursue a more precise solution to the problem. First, we notice that the ultimate rate r_C for the outer code that can be achieved through the considered concatenated coding system is given by the mutual information between the independent and uniformly distributed bits at the input of the interleaver at the transmitter side and the soft information at the output of the deinterleaver at the receiver side (see Fig. 1). Because of the complicated MAP detector, this mutual information cannot be computed in closed form, but it can be easily evaluated through Monte Carlo simulations with a large number of channel realizations by obtaining the histogram of the distribution of the extrinsic information (L-values). The reason we choose histograms instead of the Arnold-Loeliger algorithm [34] is that the latter only gives the no-interleaving mutual information while our focus is mainly on interleaved systems using a soft damapper, as discussed in Section IV. Interestingly, this numerical method is equivalent to the evaluation of the EXIT chart for the MAP detector, particularly of its left-most point [26]. In fact, the left-most point of a detection EXIT chart gives the ultimate rate achievable by the outer code when it is concatenated with the inner detector through an interleaver and iterative detection/decoding is not allowed [26]. Hence, for a given marker code with rate r_M, we can evaluate the ultimate value of r_C by means of this numerical method, and then compute the ultimate overall rate as $r = r_C r_M$. We will exploit this result in the next subsection to find optimal marker codes for channels with insertions and deletions.

C. Marker Code Optimization

In this section, we study the problem of selecting a good marker code. We first notice that a lower marker code rate or smaller N_C leads to better synchronization capabilities since the positions of the insertions and deletions can be located more precisely; however, this is obtained with an increased overhead. This argument suggests that an optimal marker code rate r_M exists for different marker codes used over an insertion/deletion channel.

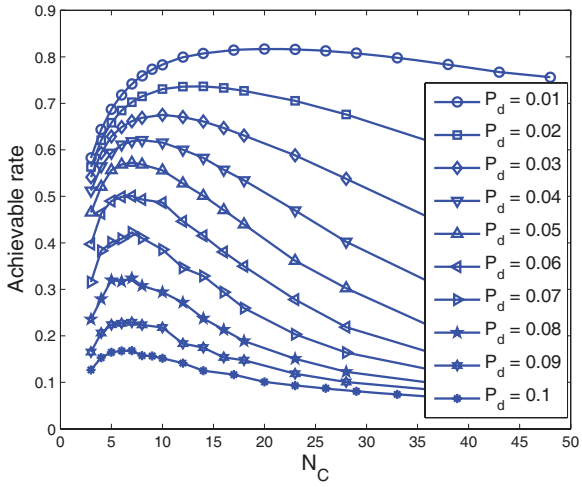

Fig. 3. Achievable rates for different deletion channels for the marker "01" inserted every N_c bits.

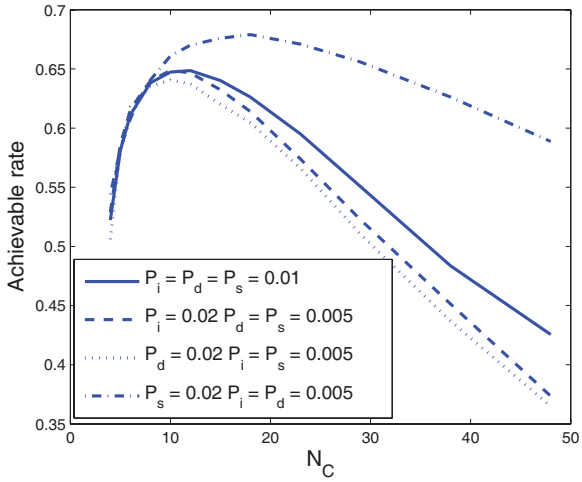

Fig. 4. Achievable rates for different insertion and deletion channels for the marker "01" inserted every N_c bits.

Some results obtained by means of the proposed information rate evaluation method of the previous subsection are shown in Figs. 3-5. Particularly, in Fig. 3, it is shown how the overall rate varies, for different deletion channels ($P_i = P_s = 0$), as a function of N_C, when the two-bit marker "01" is inserted every N_C information-carrying bits. For each value of the deletion probability P_d, a clear maximum is obtained, which determines the marker code rate that is information-theoretically optimal. Not surprisingly, as deletions become more frequent, the achievable rate decreases, so does the rate of the optimal marker code, since an effective synchronization process requires more pilot bits. As another example, Fig. 4 compares the impacts of insertion, deletion and substitution errors on the achievable rates with the constraint that $P_i + P_d + P_s = 0.03$. It is clear that for this particular example, the deletion errors cause more severe damage than the insertion errors to the performance while the substitution errors degrade the capacity much less than the synchronization errors. Note that these achievable rates

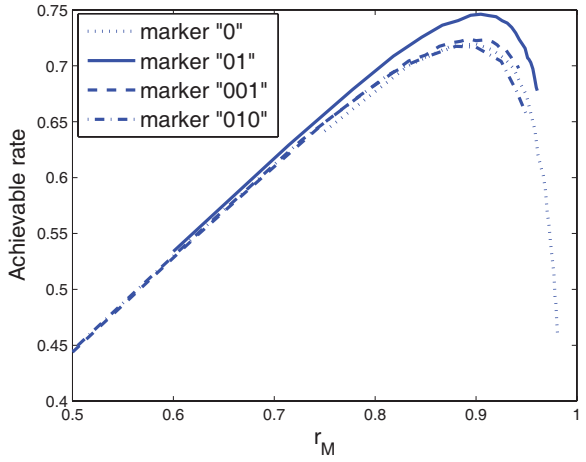

Fig. 5. Achievable rates for different markers as a function of the marker code rate when $P_d = 0.01$, $P_s = 0.01$.

are only valid if a single synchronization stage is employed (single-pass decoding) and they are violated when an iterative decoding/synchronization scheme is adopted. We also note that the gap to the existing Shannon capacity lower bounds is also large. For instance, a lower bound for the capacity of an i.i.d. deletion channel with $P_d = 0.05$ is 0.728 [35] while the maximum rate found in Fig. 3 is less than 0.6.

The proposed approach can be used not only to find the optimal rate for a given marker code, but also to compare different marker codes. As an example, the marker code "00" is clearly not a good choice compared to "01" since there is no transition between the two bits and the receiver cannot determine as precisely whether an insertion or deletion error happens prior to the specific marker. On the other hand, for "01", there is a transition in the marker sequence and a single deletion or insertion can be easily identified. In Fig. 5, we compare four regular marker codes obtained by inserting the markers "0", "01", "001", and "010" every N_C information bits, for the case of a deletion/substitution channel with $P_d = P_s = 10^{-2}$. The results, which are given in terms of the overall rate r as a function of the marker code rate r_M, show that for this particular channel the best choice of marker code among the three candidates is to insert the pilot bits "01" every 18 information bits, which provides an overall rate of about 0.75. It is also not surprising to see that the marker "001" outperforms "010" for higher marker code rates. This is attributed to the following: it is more likely that more than one bit get deleted between two adjacent markers and hence the marker "010" may not be able to detect these synchronization errors while the marker "001" still can.

We conclude this section by noting that for all the studied scenarios, the guidelines for marker code design that we obtain through our analysis are in good agreement with the approximate analysis proposed in [21].

IV. Symbol-Level Synchronization

One key observation is that, since insertion/deletion channels have memory, the soft information at the output of the MAP detector corresponding to two bits with different time indices is correlated. Hence, information is lost when such correlations are neglected, which is exactly what is done in our concatenated system because of the presence of the interleaver/deinterleaver [26].[2] On the other hand, interleaving is fundamental because it allows us to split the decoding process into two serial steps, namely the inner detection and the outer decoding; the other option being joint detection/decoding, which would be computationally infeasible [19]–[21]. In the following, we propose a solution that allows us to recover part of the information loss while preserving the interleaving process, hence also its advantage of splitting the decoding process into an inner detection and an outer decoding.

A. Symbol Level MAP Detection

We introduce MAP detection at the symbol level, defining a symbol as a group of m consecutive bits. Consequently, the T transmitted bits are partitioned into T_S symbols, $S_k = \mathbf{x}_{m(k-1)+1}^{mk}$, $k \in \{1, 2, \ldots, T_S\}$, taking values on $\{0, 1\}^m$. The last symbol, however, may consists of less than m bits, but we assume that $T/m = T_S$ is an integer for simplicity. In this case, synchronization can be carried out by means of a symbol-level FBA which is obtained by extending the bit-level derivation given in [19], [22] to the symbol-level case. In the following, we provide the details of the algorithm.

Let us re-define the binary event $D_{k,n}$, with $k \in \{1, 2, \ldots, T_S\}$ and $n \in \{1, 2, \ldots, R\}$, which denotes whether, of the first k transmitted symbols (i.e., km bits), exactly n bits are received or not, possibly after being corrupted by the channel. With this redefinition of the event $D_{k,n}$, the definitions in (2) and (3) still hold. As in the bit-level case, the coefficients can be computed by means of the forward/backward recursions. For simplicity, we give here the formulations for the case $m = 2$, i.e., bits $\{x_{2k-1}, x_{2k}\}$ are grouped as one symbol, noting that the extension to the case of $m > 2$ is straightforward. In this case, there are 9 possible ways to reach a certain state on the trellis, and the resulting recursions are given as shown in (7) and (8), respectively, and are both initialized by exploiting again the exact "frame synchronization" assumption. Finally, the target extrinsic information can be computed as shown in (9).

B. Achievable Rate Improvement with Symbol Level Synchronization

As an example use of the proposed algorithm, Fig. 6 compares the mutual information between the symbols at the input of the interleaver at the transmitter side and the soft information at the output of the deinterleaver at the receiver side, for the case of one-bit symbols and two-bit symbols. Specifically, it is shown how the overall achievable rate varies, for an IID deletion channel ($P_i = P_s = 0$, $P_d = 0.01$), as a function of N_C, when the two-bit marker "01" is inserted every N_C information-carrying bits. For comparison, the mutual information computed in the absence of interleaving, i.e., by evaluating the expectations $E[\log P(\mathbf{y}_1^R)]$ and $E[\log P(\mathbf{x}_1^T, \mathbf{y}_1^R)]$ using Monte Carlo techniques with a large number of channel simulations, and obtaining $I(\mathbf{x}_1^T; \mathbf{y}_1^R)$ as $T - E[\log P(\mathbf{y}_1^R)] + E[\log P(\mathbf{x}_1^T, \mathbf{y}_1^R)]$ [36], is also shown.

[2] In the context of outer LDPC codes, the interleaving is implicit.

$$\begin{aligned}\alpha_k(n) \;=\; & P_d^2\,\alpha_{k-1}(n) + P_dP_t\,\alpha_{k-1}(n-1)\sum_{i=0}^{1}\sum_{x_{2k-i}} P(x_{2k-i})F(x_{2k-i},y_n)\\ & + P_t^2\,\alpha_{k-1}(n-2)\sum_{x_{2k-1}} P(x_{2k-1})F(x_{2k-1},y_{n-1})\cdot\sum_{x_{2k}} P(x_{2k})F(x_{2k},y_n)\\ & + \frac{P_i}{4}P_d\,\alpha_{k-1}(n-2)\cdot 2 + \frac{P_i^2}{16}\,\alpha_{k-1}(n-4)\\ & + \frac{P_i}{4}P_t\,\alpha_{k-1}(n-3)\sum_{i=0}^{1}\sum_{x_{2k-i}} P(x_{2k-i})F(x_{2k-i},y_{n-2i}) \end{aligned} \tag{7}$$

$$\begin{aligned}\beta_k(n) \;=\; & P_d^2\,\beta_{k+1}(n) + P_dP_t\,\beta_{k+1}(n+1)\sum_{i=1}^{2}\sum_{x_{2k+i}} P(x_{2k+i})F(x_{2k+i},y_{n+1})\\ & + P_t^2\,\beta_{k+1}(n+2)\sum_{x_{2k+1}} P(x_{2k+1})F(x_{2k+1},y_{n+1})\cdot\sum_{x_{2k+2}} P(x_{2k+2})F(x_{2k+2},y_{n+2})\\ & + \frac{P_i}{4}P_d\,\beta_{k+1}(n+2)\cdot 2 + \frac{P_i^2}{16}\,\beta_{k+1}(n+4)\\ & + \frac{P_i}{4}P_t\,\beta_{k+1}(n+3)\sum_{i=1}^{2}\sum_{x_{2k+i}} P(x_{2k+i})F(x_{2k+i},y_{n+2i-1})\;, \end{aligned} \tag{8}$$

$$\begin{aligned} p(\mathbf{y}_1^R|x_{2k-1},x_{2k}) \;=\; & P_d^2\sum_{n=0}^{\min(4k,R)}\alpha_{k-1}(n)\beta_k(n) + P_dP_t\sum_{n=0}^{\min(4k,R)}\sum_{i=0}^{1}\alpha_{k-1}(n-1)\beta_k(n)F(x_{2k-i},y_n)\\ & + P_t^2\sum_{n=0}^{\min(4k,R)}\alpha_{k-1}(n-2)\beta_k(n)F(x_{2k-1},y_{n-1})F(x_{2k},y_n) + \frac{P_i}{4}P_d\sum_{n=0}^{\min(4k,R)}\alpha_{k-1}(n-2)\beta_k(n)\cdot 2\\ & + \frac{P_i}{4}P_t\sum_{n=0}^{\min(4k,R)}\sum_{i=0}^{1}\alpha_{k-1}(n-3)\beta_k(n)F(x_{2k-i},y_{n-2i}) + \frac{P_i^2}{16}\sum_{n=0}^{\min(4k,R)}\alpha_{k-1}(n-4)\beta_k(n)\;. \end{aligned} \tag{9}$$

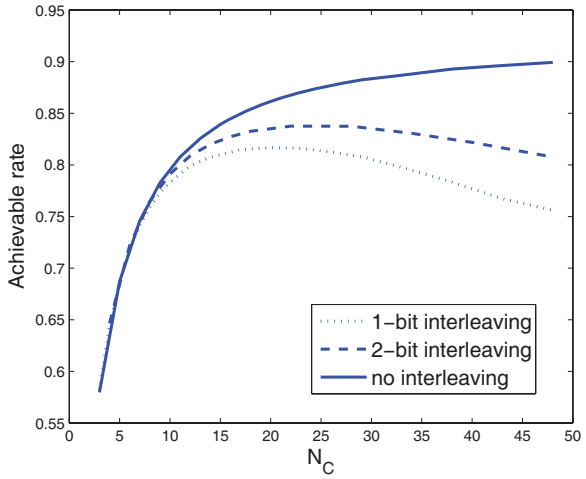

Fig. 6. Achievable rate improvement through symbol-level decoding for the marker "01" inserted every N_c bits.

This curve quantifies the transmission rate loss due to interleaving. Since the complexity of the algorithm grows exponentially in the group size m which makes it infeasible for large values of m, only the achievable rate for the case of 2-bit interleaving is shown. It is clear that adopting symbol-level detection recovers a significant part of the interleaving loss, particularly as the marker code rate increases. For instance, by comparing the two relevant maximum achievable rates, we can conclude that symbol-level detection is about 5% better in capacity for the given example. Although omitted from this paper, other simulation results also show similar gains for different channels, e.g., the insertion-only channel and insertion/deletion channels.

C. Exploiting Correlation via Demapper/Detector

In this section, we consider a practical coding scheme with the aim of confirming the performance gain predicted by our information-theoretic analysis for the symbol-level detection over the bit-level detection. Specifically, we adopt a binary LDPC code of length 16383 and rate $r_C = 0.87$ concatenated with a marker code with rate $r_M = 30/32$, obtained by inserting the marker "01" every 30 LDPC-coded bits. Hence, $r = 0.8156$ is obtained for the overall code rate. We compare the performance obtained by feeding the LDPC decoder with the soft information produced by the bit-level detector and the symbol-level detector (with $m = 2$ and $m = 3$). In the bit-level detection case, the output of the detector directly feeds the LDPC decoder, which performs 100 self iterations and then produces the estimate of the information bits. In the symbol-level detection case, the output of the detector cannot directly feed the LDPC decoder, which

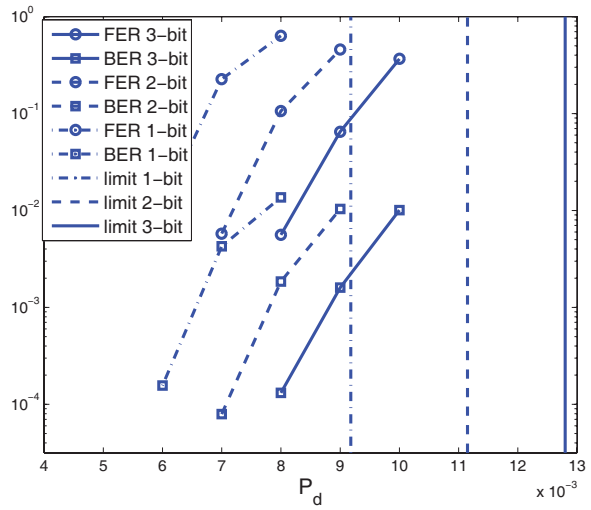

Fig. 7. Decoding improvement through symbol-level decoding for 1-bit interleaving, 2-bit interleaving and 3-bit interleaving.

is binary and cannot manage symbol-level soft information. Hence, to convert the symbol-level information to bit-level information, we adopt the soft demapper module proposed in [37]. Specifically, we use iterative soft demapping (see [37] for detailed formulations): for every 10 self iterations of the LDPC decoder, we perform one iteration of the soft demapper, so that the total number of 100 self iterations of the LDPC decoder is preserved for a fair comparison with the bit-level case.

The resulting frame-error rate (FER) and bit-error rate (BER) curves are compared in Fig. 7, for the case of a deletion only channel. For comparison, the ultimate deletion probability P_d at which a scheme with the considered marker code and an outer code with rate $r_C = 0.87$ can provide reliable communications is also shown — these values are obtained by means of the information-theoretic analysis described in the previous sections. An interesting fact is that a BER lower than 10^{-2} is obtained by means of MAP detection with $m = 3$ at values of the deletion probability at which bit-level detection cannot converge even in the presence of an information-theoretically optimal code as shown in Fig. 7. The improvement provided by the symbol-level detection is evident: for a given BER, using a MAP detector with $m = 2$ allows the receiver to work with a deletion probability increased by about 10^{-3} with respect to the bit-level one, and the MAP detector with $m = 3$ provides an even greater robustness to deletion errors.

V. EXIT Chart-Based Outer LDPC Code Design for Insertion/Deletion Channels

In the previous sections, with the interest of reducing decoding latency, we focused on the case of single-pass decoding for the outer code concatenated with the inner marker code over insertion/deletion channels. We now consider an iterative scheme where extrinsic information is exchanged between the MAP detector (synchronization) block and the outer decoder. This is motivated by the observation that when iterative decoding is allowed, specifically designed LDPC codes for insertion and deletion channels may provide performance gains over the ones optimized for AWGN-only channels. Detailed EXIT chart based analysis offers an insight into this problem.

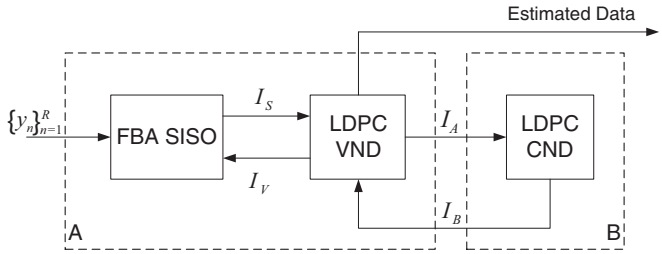

Fig. 8. Detailed decoder/detector block diagram at the receiver side.

In this section, we consider an LDPC code consisting of N variable nodes and $N - K$ check nodes connected by an edge interleaver [38] with rate $r_C = K/N$. For simplicity, as in [26], only check-regular LDPC codes are considered, i.e., every parity-check equation involves a constant number of variable nodes, denoted by d_c. We emphasize that joint design of variable and check nodes may offer a better performance but the check-regular LDPCs already give good results as reported in the previous literature. Suppose I is the total number of different variable node degrees of the LDPC code denoted by $d_{v,i},\ i = 1, \ldots, I$. Let a_i to be the fraction of variable nodes with degree $d_{v,i}$. The goal of code design is to find the set of parameters $\{\lambda_i\}$ that provides the best decoding performance where [26]

$$\sum_{i=1}^{I} \lambda_i = 1, \quad 0 \le \lambda_i \le 1,$$
$$\lambda_i = \frac{d_{v,i}}{(1 - r_C) d_c} \cdot a_i. \tag{10}$$

Because of the first constraint, we need $I \ge 3$ to have any flexibility in our code design.

A. EXIT Chart Based Analysis of the Decoding Performance

Since the outer LDPC decoder can be partitioned into LDPC variable node detector (VND) and LDPC check node detector (CND) [26], for multiple-pass decoding, the information exchage between the inner MAP detector and outer LDPC decoder is further illustrated in Fig. 8, where Block A consists of two sub-blocks which are referred to as FBA SISO and LDPC VND. Mutual information between the LDPC-coded bits and the corresponding L-values, $\{I_A, I_B, I_S, I_V\} \in [0, 1]$, are exchanged between these blocks during the iterative decoding process. It is worth mentioning that only the extrinsic information, i.e., the difference between the *a posteriori* and the *a priori* L-values, is exchanged [26].

As stated in Section III, in the sub-block FBA SISO, MAP detection is applied on the received sequence $\{y_k\}$ with soft input *a priori* information given by I_V and extrinsic L-values of the transmitted bits are generated. I_S measures the reliability of these L-values. It is difficult to describe the relationship between I_V and I_S in closed form, instead, Monte Carlo simulations are performed to generate the so-called detection EXIT chart. A detection EXIT chart example for insertion and deletion channels is shown in Fig. 9 using bit-level synchronization and the marker code "01". Marker code

TABLE I
LDPC CODE PARAMETERS FOR INSERTION AND DELETION CHANNELS

	r_M	r_C	d_c	d_v	a
$P_i = P_d = 0.001$	0.96	0.9841	189	{2 3 225}	{0.8183 0.178 0.0037}
$P_i = P_d = 0.003$	0.92	0.96	75	{2 3 104}	{0.1782 0.82 0.0018}
$P_i = P_d = 0.005$	0.9	0.9412	51	{2 3 75}	{0.2762 0.72 0.0038}
$P_i = P_d = 0.007$	0.8824	0.9231	39	{2 3 57}	{0.2769 0.718 0.0051}
$P_i = P_d = 0.009$	0.8571	0.9091	33	{2 3 44}	{0.244 0.75 0.006}
$P_i = P_d = 0.01$	0.8571	0.9	30	{2 3 45}	{0.3233 0.669 0.0077}
$P_i = P_d = 0.02$	0.8333	0.8125	16	{2 3 52}	{0.4175 0.574 0.0085}
$P_i = P_d = 0.03$	0.8333	0.7273	11	{2 3 56}	{0.5751 0.414 0.0109}
$P_i = P_d = 0.04$	0.8333	0.625	8	{2 3 97}	{0.4641 0.531 0.0049}
$P_i = P_d = 0.05$	0.8333	0.5	6	{2 3 97}	{0.2286 0.769 0.0024}
$P_d = 0.002$	0.96	0.9836	183	{2 3 227}	{0.3723 0.626 0.0017}
$P_d = 0.006$	0.9355	0.9605	76	{2 3 161}	{0.1371 0.862 0.0009}
$P_d = 0.01$	0.9091	0.9464	56	{2 3 90}	{0.1928 0.805 0.0022}
$P_d = 0.02$	0.875	0.9091	33	{2 3 65}	{0.2401 0.756 0.0039}
$P_d = 0.04$	0.8	0.85	20	{2 3 33}	{0.2148 0.778 0.0072}
$P_d = 0.06$	0.7778	0.7857	14	{2 3 31}	{0.2868 0.703 0.0102}
$P_d = 0.08$	0.7778	0.7	10	{2 3 50}	{0.1978 0.798 0.0042}
$P_d = 0.1$	0.7778	0.625	8	{2 3 69}	{0.2679 0.728 0.0041}
$P_i = 0.002$	0.96	0.9839	186	{2 3 256}	{0.4801 0.518 0.0019}
$P_i = 0.006$	0.9355	0.9605	76	{2 3 170}	{0.1402 0.859 0.0008}
$P_i = 0.01$	0.9091	0.9464	56	{2 3 87}	{0.2293 0.768 0.0027}
$P_i = 0.02$	0.875	0.9091	33	{2 3 53}	{0.2775 0.717 0.0055}
$P_i = 0.04$	0.8	0.85	20	{2 3 25}	{0.2554 0.733 0.0116}
$P_i = 0.06$	0.7778	0.7857	14	{2 3 30}	{0.2507 0.74 0.0093}
$P_i = 0.08$	0.7778	0.7273	11	{2 3 29}	{0.4314 0.552 0.0166}
$P_i = 0.1$	0.7778	0.6667	9	{2 3 28}	{0.5125 0.467 0.0205}

rates are chosen based on the scheme proposed in Section III-C.

The variable nodes take I_S as the *a priori* information and perform the standard sum-product algorithm (SPA) with information received from the LDPC CND. The EXIT curve of the combined FBA SISO and LDPC VND is described by the relationship between I_A and I_B, given by [26]

$$I_A(I_B, d_v) = J\left(\sqrt{(d_v - 1)[J^{-1}(I_B)]^2 + [J^{-1}(I_S)]^2}\right), \tag{11}$$

where the function $J(\sigma)$ is defined as

$$J(\sigma) = 1 - \int_{-\infty}^{\infty} \frac{e^{-(\xi - \sigma^2/2)^2/2\sigma^2}}{\sqrt{2\pi\sigma^2}} \cdot \log_2[1 + e^{-\xi}] d\xi. \tag{12}$$

In this case, I_S can be numerically evaluated from the detection EXIT chart using a polynomial approximation with input $I_V(I_B, d_v) = J\left(\sqrt{d_v} \cdot J^{-1}(I_B)\right)$. For instance, when $P_i = P_d = 0.01$, we can write

$$I_S = 0.41491 \cdot I_V^5 - 1.1518 \cdot I_V^4 + 1.2405 \cdot I_V^3 - 0.71968 \cdot I_V^2 + 0.33549 \cdot I_V + 0.83146.$$

For a certain variable node degree distribution, the effective VND transfer curve is thus

$$I_A(I_B) = \sum_{i=1}^{I} \lambda_i \cdot I_A(I_B, d_{v,i}) = \sum_{i=1}^{I} \lambda_i \cdot J\left(\sqrt{(d_{v,i} - 1)[J^{-1}(I_B)]^2 + [J^{-1}(I_S)]^2}\right). \tag{13}$$

At the CND, "box plus" operation [31] is performed to generate I_B from I_A which can be approximately written as (it is useful to express it as the inverse function) [26]

$$I_B^{-1}(I_A, d_c) = I_A(I_B, d_c) \approx 1 - J\left(\frac{J^{-1}(1 - I_B)}{\sqrt{d_c - 1}}\right). \tag{14}$$

The EXIT curves $I_A(I)$ and $I_B^{-1}(I)$ form the EXIT chart for the entire receiver which predicts the decoding performance. At the initialization step of the decoding process, FBA SISO computes the output extrinsic L-values with no *a priori* information i.e., $I_V = 0$. VND utilizes the mutual information I_S to start the SPA during which the mutual information $I_A(0)$ and $I_B\big(I_A(0)\big)$ are exchanged between VND and CND. After one iteration of SPA, VND generates the output extrinsic information which serves as the *a priori* information for FBA SISO and starts to iterate from the first step. The difference between the current and previous iteration is that VND produces more reliable information $I_A\Big(I_B\big(I_A(0)\big)\Big)$ if $I_A(I) > I_B^{-1}(I)\ \forall I \in [0, 1)$. Iterative decoding stops when a valid LDPC codeword is obtained or the maximum number of iterations is reached. At the end of the process, the overall L-values are produced for the estimation of the transmitted bit sequence. Note that when the condition $I_A(I) > I_B^{-1}(I)\ \forall I \in [0, 1)$ is satisfied, i.e., the "tunnel" created by the two curves $I_A(I)$ and $I_B^{-1}(I)$ is open, the mutual information of VND output will converge to 1 after several iterations which leads to a decoding performance with an error rate approaching zero. Thus, the goal of LDPC code design is to find a set of $\{\lambda_i\}$ that keeps the tunnel open for the highest deletion/insertion rate.

TABLE II
PERFORMANCE IMPROVEMENT AT A BER LEVEL OF 10^{-3} WITH SPECIFIC LDPC CODE DESIGN OVER INSERTION/DELETION CHANNELS

$P_i = P_d$	0.001	0.003	0.005	0.007	0.009	0.01	0.02	0.03	0.04	0.05
Ratio of BERs	1.04	1.256	3.061	5.065	7.018	8.477	14.83	22.07	29.31	35.68

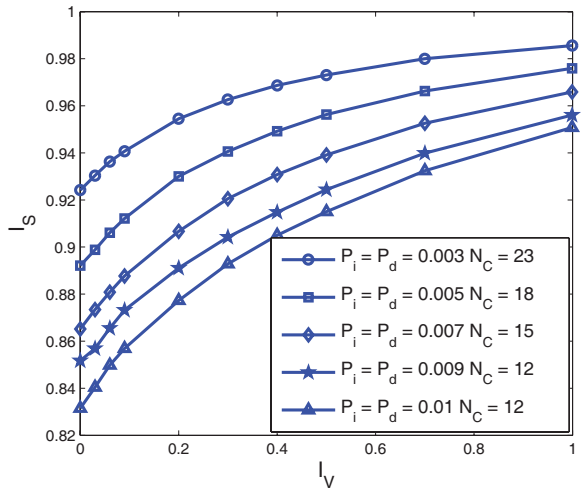

Fig. 9. Detection EXIT chart for several insertion and deletion channels for the marker "01" inserted every N_c bits.

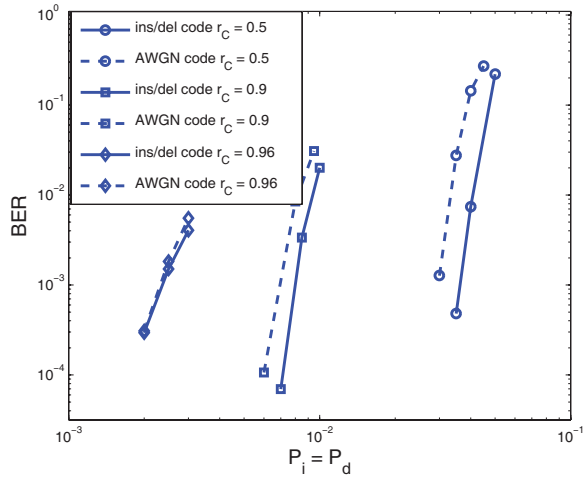

Fig. 10. BER performance of different LDPC codes over an insertion/deletion channel with $P_i = P_d$.

B. LDPC Code Design Example for Insertion/Deletion Channels

Design examples are given in Table I using the bit-level synchronization algorithm for several insertion/deletion channels, deletion only channels and insertion-only channels, respectively. We choose $I = 3$ and fix the average variable node degree $\bar{d}_v$ to be 3. Listed LDPC code degree distributions guarantee convergence with the highest code rate r_C for different deletion/insertion rates. Therefore, the overall code rate r, product of r_M and r_C, denotes the highest achievable rate when iterative decoding is performed. For deletion probabilities of 0.01 and 0.1, the overall rates are obtained as 0.860 and 0.486, respectively, where the capacity lower bound is 0.919 for $P_d = 0.01$ and 0.531 for $P_d = 0.1$. The corresponding gaps are 0.059 and 0.045 bits/channel use for the two cases, which are clearly smaller than the one demonstrated in Section III-C. We also expect that the gap to the capacity bound can be further narrowed by allowing $I > 3$ and not fixing $\bar{d}_v$ to be 3.

The advantages of the designed codes are also confirmed by the error rate simulation results shown in Fig. 10 and Table II. In the figure, we pick three codes for insertion/deletion channels with rates $r_C = 0.96$, $r_C = 0.9$ and $r_C = 0.5$ from Table I and compare them with the codes optimized for AWGN channels. The length of the LDPC codeword is set to be $N = 5000$ and the selected marker code rate is determined to maximize the transmission rate. In Table II, we calculate the ratio of the BERs of the two codes (optimized one versus the AWGN-only code) when the codes optimized for insertion and deletion channels attain a BER of 10^{-3}. The higher the ratio, the greater the improvement. Clearly, all of the codes outperform the ones designed for AWGN channels. However, the gap becomes less obvious as the insertion/deletion rate decreases. This is not a surprising result because for low insertion/deletion probabilities, the detection EXIT charts tend to be flat as illustrated in Fig. 9, which is similar to the one for a memoryless AWGN channel. In this case, specific design of an LDPC code for insertion/deletion channel may not be required since the gain is negligible. Similar conclusions are drawn for ISI channels with short channel impulse responses in [29]. Also, when the symbol-level detection is performed, the left-most point in the detection EXIT chart is much better than the bit-level case as explained in Section IV-B. The right-most point in the detection EXIT chart is identical for both cases since MAP detector achieves ideal synchronization in this case. Therefore, the detection EXIT chart for symbol-level detection is flatter than the one for the bit-level case. This observation suggests that for channels with low insertion/deletion rates, it is more likely that symbol-level detection itself already yields a good performance and iterative decoding and LDPC code design may not be needed, which is also an obvious fact, since when $m = T$, optimal detection (i.e., for synchronization purposes) is achieved and there is no gain with iterative decoding/demapping. Clearly, this is not feasible in practice since the detection complexity in m is exponential and T is typically large.

VI. CONCLUSIONS

We have studied performance of an outer LDPC code concatenated with an inner marker code for data transmission over insertion/deletion channels. Two decoding strategies are considered: single-pass decoding and multi-pass decoding with information exchange between the inner detector and the outer decoder. For the first case, through numerical mutual information analyses, we have developed a technique that allows us to optimize the marker code based on the ultimate rate achievable by the concatenated scheme. Moreover, we

have presented a new symbol-level detection algorithm, which has been proved to outperform the standard bit-level one in terms of achievable rates. An iterative detector/demapper is also designed which is able to exploit the results of the symbol level synchronizer. Finally, when iterative decoding is allowed, we have shown that by choosing good variable and check node degree distributions, LDPC codes designed for insertion/deletion channels offer better error correcting capabilities than those optimal for the AWGN-only channels. Simulation results related to practical LDPC codes showing clear performance gains have been provided for both cases under consideration. Although we only focus on the marker codes (as the inner synchronization code), similar analyses and design procedure can also be applied to other concatenated coding schemes, e.g., an LDPC code concatenated with an inner watermark code [19].

REFERENCES

[1] R. L. White, R. M. H. New, and R. F. W. Pease, "Patterned media: a viable route to 50 Gbit/in^2 and up for magnetic recording?" *IEEE Trans. Magnetics*, vol. 33, no. 1, pp. 990–995, Jan. 1997.

[2] R. L. Dobrushin, "Shannon's theorems for channels with synchronization errors," *Problems of Inf. Transmiss.*, vol. 3, no. 4, pp. 11–26, 1967.

[3] M. Mitzenmacher, "A survey of results for deletion channels and related synchronization channels," *Probability Surveys*, pp. 1–33, June 2009.

[4] D. Fertonani and T. M. Duman, "Novel bounds on the capacity of the binary deletion channel," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2753–2765, June 2010.

[5] D. Fertonani, T. M. Duman, and M. F. Erden, "Bounds on the capacity of channels with insertions, deletions and substitutions," *IEEE Trans. Commun.*, vol. 59, no. 1, pp. 2–6, Jan. 2011.

[6] F. F. Sellers Jr., "Bit loss and gain correction code," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 35–38, Jan. 1962.

[7] N. J. A. Sloane, "On single-deletion-correcting codes," in *Codes and Designs: Proc. Conf. Honoring Professor Dijen K. Ray-Chaudhuri on the Occasion of his 65th Birthday*, May 2000, pp. 273–291.

[8] Z. Liu and M. Mitzenmacher, "Codes for deletions and insertion channels with segmented errors," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 224–232, Jan. 2010.

[9] A. S. Helberg and H. C. Ferreira, "On multiple insertion/deletion correcting codes," *IEEE Trans. Inf. Theory*, vol. 48, no. 1, pp. 305–308, Jan. 2002.

[10] L. McAven and R. Safavi-Naini, "Classification of the deletion correcting capabilities of Reed-Solomon codes of dimension 2 over prime fields," *IEEE Trans. Inf. Theory*, vol. 53, no. 6, pp. 2280–2294, June 2007.

[11] S. E. Tavares and M. Fukada, "Matrix approach to synchronization recovery for binary cyclic codes," *IEEE Trans. Inf. Theory*, vol. 15, no. 1, pp. 93–101, Jan. 1969.

[12] G. Seguin, "On synchronizable binary cyclic codes," *IEEE Trans. Inf. Theory*, vol. 21, no. 5, pp. 589–592, Sep. 1975.

[13] L. Cheng and H. Ferreira, "Rate-compatible path-pruned convolutional codes and their applications on channels with insertion, deletion and substitution errors," in *Proc. IEEE Information Theory Workshop 2005*, Aug. 2005, pp. 20–25.

[14] T. G. Swart and H. C. Ferreira, "Insertion/deletion correcting coding schemes based on convolution coding," *Electron. Lett.*, vol. 38, no. 16, pp. 871–873, Aug. 2002.

[15] M. F. Mansour and A. H. Tewfik, "Convolutional codes for channels with substitutions, insertions, and deletions," in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, Nov. 2002, pp. 1051–1055.

[16] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Dokl. Akad. Nauk SSSR*, vol. 163, no. 4, pp. 845–848, 1965.

[17] H. C. Ferreira, T. G. Swart, and M. P. dos Santos, "Using parallel-interconnected Viterbi decoders to correct insertion/deletion errors," in *Proc. IEEE 7th AFRICON Conference*, vol. 1, 2004, pp. 341–344.

[18] L. Schulman and D. Zuckerman, "Asymptotically good codes correcting insertions, deletions, and transpositions," *IEEE Trans. Inf. Theory*, vol. 45, no. 7, pp. 2552–2557, Nov. 1999.

[19] M. C. Davey and D. J. Mackay, "Reliable communication over channels with insertions, deletions and substitutions," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 687–698, Feb. 2001.

[20] E. A. Ratzer, "Error-correction on non-standard communication channels," Ph.D. dissertation, University of Cambridge, Sep. 2003.

[21] ——, "Marker codes for channels with insertions and deletions," *Annals of Telecommun.*, vol. 60, pp. 29–44, Jan. 2003.

[22] L. R. Bahl and F. Jelinek, "Decoding for channels with insertions, deletions, and substitutions with applications to speech recognition," *IEEE Trans. Inf. Theory*, vol. 21, pp. 404–411, July 1975.

[23] J. Chen, M. Mitzenmacher, C. Ng, and N. Varnica, "Concatenated codes for deletion channels," in *Proc. IEEE International Symposium on Information Theory*, 2003, p. 218.

[24] R. G. Gallager, "Sequential decoding for binary channels with noise and synchronization errors," MIT Lincoln Lab. Group, Tech. Rep., Oct. 1961.

[25] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.

[26] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 670–678, Apr. 2004.

[27] M. Franceschini, G. Ferrari, and R. Raheli, "Does the performance of LDPC codes depend on the channel?" *IEEE Trans. Commun.*, vol. 54, no. 12, pp. 2129–2132, Dec. 2006.

[28] N. Varnica and A. Kavčić, "Optimized low-density parity-check codes for partial response channels," *IEEE Commun. Lett.*, vol. 7, no. 4, pp. 168–170, Apr. 2003.

[29] M. Franceschini, G. Ferrari, and R. Raheli, "EXIT chart-based design of LDPC codes for inter-symbol interference channels," in *Proc. IST Summit*, June 2005.

[30] M. Mitzenmacher, "Capacity bounds for sticky channels," *IEEE Trans. Inf. Theory*, vol. 54, no. 1, pp. 72–77, Jan. 2008.

[31] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inf. Theory*, vol. 42, pp. 429–445, Mar. 1996.

[32] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, pp. 498–519, Feb. 2001.

[33] A. Kavčić, X. Mao, and M. Mitzenmacher, "Binary intersymbol interference channels: Gallager codes, density evolution, and code performance bounds," *IEEE Trans. Inf. Theory*, vol. 40, no. 7, pp. 1636–1652, July 2003.

[34] D. Arnold, H. Loeliger, P. Vontobel, A. Kavčić, and W. Zeng, "Simulation-based computation of information rates for channels with memory," *IEEE Trans. Inf. Theory*, vol. 52, no. 8, pp. 3498–3508, Aug. 2006.

[35] E. Drinea and A. Kirsch, "Directly lower bounding the information capacity for channels with i.i.d. deletions and duplications," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 86–102, Jan. 2010.

[36] J. Hu, T. M. Duman, M. F. Erden, and A. Kavčić, "Achievable information rates for channels with insertions, deletions and intersymbol interference with i.i.d. inputs," *IEEE Trans. Commun.*, vol. 58, no. 4, pp. 1102–1111, Apr. 2010.

[37] X. Li and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding," *IEEE Commun. Lett.*, vol. 1, no. 6, pp. 169–171, Nov. 1997.

[38] D. J. Mackay, "Good error correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, pp. 399–431, Mar. 1999.

Feng Wang received the B.S. degree from Southeast University, Nanjing, in 2005 and the M.S. degree from Michigan Technological University, Houghton, in 2007, respectively, all in electrical engineering. He is currently working towards the Ph.D. degree in Electrical Engineering at Arizona State University. His research interests lie in the areas of digital communications, particularly coding and detection techniques, with application to digital storage systems and wireless communication systems.

Dario Fertonani was born in Mantua, Italy, in 1980. He received the M.S. degree in telecommunications engineering in 2004 and the Ph.D. degree in information technology in 2008, both from the University of Parma, Parma, Italy. He held a postdoctoral position at Arizona State University, Tempe, AZ, from May 2008 to December 2009. Currently, he works at Qualcomm Inc., San Diego, CA. His research interests include various topics in digital communications and information theory, with a particular focus on detection and coding for channels with memory. He is the coauthor of several publications in international journals and conferences.

Tolga M. Duman (S'95-M'98-SM'03-F'11) is a Professor of Electrical Engineering with the School of ECEE at Arizona State University. He received the B.S. degree from Bilkent University in Turkey in 1993, M.S. and Ph.D. degrees from Northeastern University, Boston, in 1995 and 1998, respectively, all in electrical engineering. Since August 1998, he has been with the Electrical Engineering Department of Arizona State University first as an Assistant Professor (1998-2004), then as an Associate Professor (2004-2008), and starting August 2008 as a Professor. Dr. Duman's current research interests are in systems, with particular focus on communication and signal processing, including wireless and mobile communications, coding/modulation, coding for wireless communications, data storage systems and underwater acoustic communications.

Dr. Duman has served as an editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS (2003-08) and IEEE COMMUNICATIONS SURVEYS AND TUTORIALS (2002-07). He is currently an editor for IEEE TRANSACTIONS ON COMMUNICATIONS in the area of coding and communication theory (2007-present), and for *Elsevier Physical Communications Journal* (2010-present). Dr. Duman is a Fellow of IEEE, a recipient of the National Science Foundation CAREER Award and IEEE Third Millennium medal.