On the Application of Factor Graphs and the Sum–Product Algorithm to ISI Channels

Giulio Colavolpe and Gianpietro Germi

Abstract—In this paper, based on the application of the sum–product (SP) algorithm to factor graphs (FGs) representing the joint *a posteriori* probability (APP) of the transmitted symbols, we propose new iterative soft-input soft-output (SISO) detection schemes for intersymbol interference (ISI) channels. We have verified by computer simulations that the SP algorithm converges to a good approximation of the exact marginal APPs of the transmitted symbols if the FG has girth at least 6. For ISI channels whose corresponding FG has girth 4, the application of a *stretching* technique allows us to obtain an equivalent girth-6 graph.

For sparse ISI channels, the proposed algorithms have advantages in terms of complexity over optimal detection schemes based on the Bahl–Cocke–Jelinek–Raviv (BCJR) algorithm. They also allow a parallel implementation of the receiver and the possibility of a more efficient complexity reduction. The application to joint detection and decoding of low-density parity-check (LDPC) codes is also considered and results are shown for some partial-response magnetic channels. Also in these cases, we show that the proposed algorithms have a limited performance loss with respect to that can be obtained when the optimal "serial" BCJR algorithm is used for detection. Therefore, for their parallel implementation, they represent a favorable alternative to the modified "parallel" BCJR algorithm proposed in the literature for the application to magnetic channels.

Index Terms—Factor graphs, intersymbol interference (ISI) channels, iterative detection, low-density parity-check (LDPC) codes, partial-response channels, sum–product (SP) algorithm.

I. INTRODUCTION

IN recent years, the use of factor graphs (FGs) to represent complicated global functions of many variables and the marginalization of these functions by means of the sum–product (SP) algorithm have attracted the attention of many researchers in the communications area [1]. Originally devised to describe the decoding algorithm for low-density parity-check (LDPC) codes [2]–[5], this framework has been extended to other probabilistic models [1], [6], [7]. It can be used not only to reinterpret existing results, but also to derive new detection/decoding strategies [8]–[10], or to devise new applications for existing algorithms [11].

In this paper, we consider the application of this framework to detection over a known intersymbol interference (ISI)

channel. By representing on an FG the joint *a posteriori* probability (APP) mass function of the transmitted symbols and applying to this graph the SP algorithm, we derive a simple soft-input soft-output (SISO) detection algorithm that can be used for turbo equalization [12]. The same approach and the resulting detection algorithm is also considered in [13]. However, from the observation that in general the FG describing the channel has cycles,[1] the authors of [13] conclude that the derived algorithm is suboptimal and therefore they discard this approach. On the contrary, we have verified by extensive computer simulations that when the girth of the graph is at least 6, the performance is practically optimal. In addition, in the case of graphs of girth 4, we apply the SP algorithm to a new FG obtained by transforming the original one. The resulting new algorithm has a negligible performance loss with respect to optimal detection.

The proposed algorithms have a complexity which depends on the number of nonzero channel interferers only. As a consequence, with respect to the Bahl–Cocke–Jelinek–Raviv (BCJR) algorithm [14], whose complexity depends on the channel memory, the proposed algorithms are more suited for *sparse* ISI channels, i.e., channels characterized by a large memory but a small number of interferers, as may occur in high-frequency transmissions due to multipath [15]. For these channels, other algorithms have been proposed based on different approaches, to obtain the same goal [15], [16].

Another advantage over the BCJR algorithm consists of the intrinsic parallel structure of the SP algorithm with *flooding schedule* [17], allowing very-high-speed detection. This aspect is very important when an LDPC code is used and turbo equalization is performed. In fact, if detection is performed by using the proposed algorithms, an *overall* graph taking into account both the code and the channel model can be built, allowing combined detection and decoding in a fully parallel manner without an increase in complexity with respect to separate detection and decoding. On the contrary, when the BCJR algorithm is used for detection, combined detection and decoding is not possible and the serial structure of the BCJR algorithm prevents from the possibility of a parallel implementation of the receiver.

In the technical literature on magnetic recording, for applications with LDPC codes, mainly the BCJR algorithm is considered for detection [18]–[22]. In order to overcome the above mentioned limitations, several schemes have been proposed in which the BCJR is run only once or one time each K iterations of the LDPC decoder [20]–[22], although it has been shown that the best schedule is obtained for $K = 1$ [13]. To allow parallel

G. Colavolpe is with the Dipartimento di Ingegneria dell'Informazione, Università di Parma, I-43100 Parma, Italy (e-mail: giulio@unipr.it).

G. Germi was with the Dipartimento di Ingegneria dell'Informazione, Università di Parma, I-43100 Italy. He is now with Selta S.p.a., Roveleto di Cadeo, Piacenza, Italy.

[1]A *cycle* is a closed path in the graph and its *length* is defined as the corresponding number of path edges. The length of the smallest cycle is the *girth* of the graph.

joint detection and decoding, in [13] a version of the BCJR algorithm with parallel schedule has been proposed. As already mentioned, the proposed detection schemes represent an alternative solution to these problems.

Finally, with respect to the BCJR algorithm, the complexity of these novel detection algorithms can be reduced more efficiently. In fact, we are able to truncate the interfering symbols with lowest weight regardless of their position. On the contrary, when the BCJR algorithm is used, we are constrained by the trellis structure to a truncation of the farthest interfering symbols [23]. Similar considerations hold for other trellis-based reduced-state soft-output schemes [24].

The paper is structured as follows. In the next section, we will describe the proposed algorithms based on FGs and the SP algorithm, discussing their convergence to the performance of the BCJR algorithm as a function of the graph girth. The implementation complexity is also addressed. In Section III, we show how to modify an original girth-4 graph to obtain an equivalent girth-6 graph over which the SP algorithm gives very good results. A performance analysis based on the average mutual information (AMI) [25]–[27] is presented in Section IV. Finally, numerical results are shown in Section V and conclusions are drawn in Section VI.

II. FACTOR GRAPH

In the considered transmission system, a sequence of M-ary code symbols $\boldsymbol{x} = \{x_n\}$ is obtained from the encoding of a sequence of information bits and a proper mapping on a multilevel constellation.[2] We will assume that the *a priori* distribution of the transmitted codewords is uniform and denote by $\chi(\boldsymbol{x})$ the code indicator function, equal to 1 if $\boldsymbol{x}$ is a valid codeword and to zero otherwise. The code sequence is transmitted over an ISI channel which also introduces additive white Gaussian noise (AWGN). At the output of a whitened matched filter, the received signal at discrete time n can be expressed as [28]

$$y_n = \sum_{\ell=0}^{L} f_\ell x_{n-\ell} + w_n \tag{1}$$

in which $\{w_n\}$ are complex independent Gaussian random variables with mean zero and variance σ^2 per component, L is the channel memory, and $\boldsymbol{f} = \{f_\ell\}_{\ell=0}^{L}$ represents the equivalent discrete-time channel impulse response, assumed perfectly known to the receiver.

The joint APP distribution of the transmitted symbols may be expressed as

$$P(\boldsymbol{x} \mid \boldsymbol{y}) \propto p(\boldsymbol{y} \mid \boldsymbol{x})P(\boldsymbol{x}) \propto \chi(\boldsymbol{x}) \prod_n g_n(x_n, \ldots, x_{n-L}) \tag{2}$$

where $\propto$ denotes proportionality, and

$$g_n(x_n, \ldots, x_{n-L}) = p(y_n \mid x_n, \ldots, x_{n-L})$$
$$\propto \exp\left\{-\frac{\left|y_n - \sum_{\ell=0}^{L} f_\ell x_{n-\ell}\right|^2}{2\sigma^2}\right\}. \tag{3}$$

For an ISI channel with $L = 4$ and $f_2 = 0$, the FG corresponding to the global function given in (2) is shown in Fig. 1. Factor nodes, associated with functions g_n, are represented by
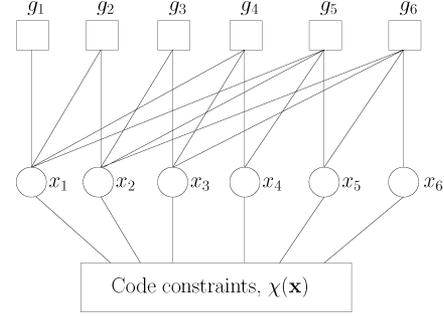
[2]An interleaver can be also used to permute the code symbols before transmission.

Fig. 1. Factor graph for an ISI channel with $L = 4$ and $f_2 = 0$.

small boxes whereas variable nodes, associated with variables x_n are represented by circles. In general, the code constraint function can be also represented by means of an FG. This graph is a portion of the overall graph and is connected with that representing the channel behavior.

The application of the SP algorithm to this graph allows the exact (in the absence of cycles) or approximate computation of the marginal APPs $\{P(x_n \mid \boldsymbol{y})\}$ necessary to implement maximum *a posteriori* probability (MAP) symbol detection. When the graph representing the ISI channel has cycles, as in general occurs, the application of the SP algorithm to this graph leads to an iterative detection process. The exchange of messages, representing the marginal APP mass functions of symbols x_n and hence assuming M values, fulfills the following rules. Denoting by $\mu_{x \to g}(x)$ the message sent from a variable node x to a function node $g(\mathcal{X})$, where $\mathcal{X}$ is the set of variables argument of g, by $\mu_{g \to x}(x)$ the message in the opposite direction, and by $\mathcal{N}(v)$ the set of neighbors of a given node v, the message computations performed at variable and factor nodes are, respectively, [1]

$$\mu_{x \to g}(x) = \prod_{h \in \mathcal{N}(x) \setminus \{g\}} \mu_{h \to x}(x) \tag{4}$$

$$\mu_{g \to x}(x) = \sum_{\sim\{x\}} \left[g(\mathcal{X}) \prod_{z \in \mathcal{N}(g) \setminus \{x\}} \mu_{z \to g}(z) \right] \tag{5}$$

where $\sum_{\sim\{x\}}$ is a *summary* operator, i.e., a sum over all variables of $g(\mathcal{X})$ excluding x. It may be observed that the message sent to an edge does not depend on the message previously received on the same edge, i.e., only *extrinsic information* is exchanged.

The messages in (4) and (5) may be also computed in the logarithmic domain. Defining $\overline{\mu}_{g \to x}(x) = \ln \mu_{g \to x}(x)$ and $\overline{\mu}_{x \to g}(x) = \ln \mu_{x \to g}(x)$, the message computations performed at variable and factor nodes become

$$\overline{\mu}_{x \to g}(x) = \sum_{h \in \mathcal{N}(x) \setminus \{g\}} \overline{\mu}_{h \to x}(x) \tag{6}$$

$$\overline{\mu}_{g \to x}(x) = \ln\left\{ \sum_{\sim\{x\}} \exp\left[\ln g(\mathcal{X}) + \sum_{z \in \mathcal{N}(g) \setminus \{x\}} \overline{\mu}_{z \to g}(z) \right] \right\}. \tag{7}$$

The implementation of this latter rule does not require multiplications but only additions and the evaluation of a nonlinear function. In fact, by using the Jacobian logarithm [29], [30], it is well known that, if a_1 and a_2 are real numbers

$$\ln(e^{a_1} + e^{a_2}) = \max(a_1, a_2) + \ln(1 + e^{-|a_2 - a_1|}) \tag{8}$$

and $\ln\left(1 + e^{-|a_2 - a_1|}\right)$ is a nonlinear function whose evaluation can be performed by means of a lookup table. In our case, the evaluation of $\ln\left(e^{a_1} + e^{a_2} + \cdots + e^{a_k}\right)$ can be done recursively [30].

A *message-passing schedule* in an FG is the specification of the order in which messages are updated. In general, the so-called *flooding schedule* is adopted [17]: in each iteration, all variable nodes and subsequently all factor nodes, pass new messages to their neighbors. As it can be easily understood, this schedule is well suited for a fully parallel implementation of the detector. Other schedules may be adopted, serial or mixed serial–parallel, according to the specific implementation requirements.

Since we are interested in focusing on the performance of the detection scheme, let us consider for the moment the case of absence of coding. Hence, symbols x_n can be assumed to be independent and identically distributed and in the graph of Fig. 1, the lower part representing the code constraints is not present. The case of coding will be considered in detail in Section IV and in the numerical results.

As already mentioned, the graph representing an ISI channel may have cycles. Let us introduce another parameter which is relevant for the ISI channel. This parameter is the number L' of nonzero interferers. When $L' \ll L$, the channel is called *sparse*. Three possible cases can be distinguished.

a) If there is only one interferer $(L' = 1)$, the graph is cycle free.[3] In a cycle-free graph, the SP algorithm computes the marginal probability for each variable exactly, independently of the used schedule [31]. The adoption of the flooding schedule instead of a more natural serial schedule starting from leaf vertices and with a natural termination [1] has the advantage of allowing parallel detection at the expense of an increase in complexity due to the need for an iterative processing.

b) If the differences between the indexes of the nonzero channel interferers are all different, the graph has girth 6.

c) If the previous conditions are not verified, the graph has girth 4.

In the cases b) and c), the SP algorithm is approximate. However, as verified by means of extensive computer simulations, not shown for the sake of space limitation, if the girth of the graph is 6 and the channel is minimum- or maximum-phase, after a few iterations, the performance of the algorithm always converges to that of the BCJR algorithm. For mixed-phase channels, we verified that the convergence is obtained when there is only a dominant interferer. Nevertheless, a whitened matched filter producing a minimum-phase channel is usually employed and, on the other hand, a mixed-phase channel can always be converted into an equivalent minimum- or maximum-phase channel by means of a linear filter. For graphs with girth 4, in general, the proposed algorithm does not converge. However, as shown in the next section, the original girth-4 graph may be transformed into an equivalent girth-6 graph on which the SP algorithm converges. In addition, we will see

that the presence of a powerful channel code often helps the convergence even in the presence of cycles of length 4 in the part of the graph describing the channel behavior.

Since the most demanding computation is due to the *summary* operations performed at factor nodes (compare the updating rules (6) and (7)), we may define a cost per coded symbol $\mathcal{C}$ of the considered algorithms as related to the above mentioned *summary* operations. Hence, we may say that for the proposed algorithm $\mathcal{C}_{\text{prop}} = N(L' + 1)M^{L'+1}$, where N is the number of iterations performed. In fact, for the computation of the M values of a message sent on one of the $L' + 1$ edges coming out from a factor node, a *summary* operation (see (7)) involving $M^{L'}$ terms is needed. Note that the same unit can be used to measure the complexity of the BCJR algorithm. In fact, it is well known [1] that this algorithm can be viewed as the application of the SP algorithm to the cycle-free Wiberg graph [6] of the channel. Hence, functions associated with factor nodes are of the same form and the complexity comparison is fair. For the BCJR algorithm the cost per coded symbol is $\mathcal{C}_{\text{BCJR}} = 3M^{L+1}$. In fact, for the forward (backward) recursion, the computation of the M^L values of the propagating message requires a *summary* operation involving M terms, whereas the computation of the M values of the marginal APP of the kth symbol requires a *summary* operation involving M^L terms. As may be observed comparing $\mathcal{C}_{\text{prop}}$ and $\mathcal{C}_{\text{BCJR}}$, the proposed algorithm is a valid alternative to the BCJR for sparse ISI channels or when parallel detection is preferred.

The described algorithm is similar to the so-called *bit-based message passing* developed in [13] for partial-response channels. However, the authors of [13] observed that the graph has cycles and therefore opted for the *state-based message passing* which is the SP algorithm with a parallel schedule applied to the cycle-free Wiberg graph [6] of the channel, that is a BCJR algorithm with parallel schedule. This algorithm, which will be denoted in the following as p-BCJR, has a cost $\mathcal{C}_{\text{p-BCJR}} = 3NM^{L+1}$.

The complexity of the proposed algorithm may be reduced following a technique similar to that described in [23], [24] for BCJR algorithms or to reduced-state sequence detection (RSSD) used for maximum-likelihood sequence detection [32]–[34]. In fact, by choosing an integer $Q < L'$, the updating rule (7) at factor nodes modeling the channel can be simplified as follows: the *summary* is performed over the Q interfering symbols with strongest weights regardless of their position, substituting the remaining interfering symbols with a decision made on the basis of the messages on the graph.[4] In this way, the cost becomes $\mathcal{C}_{\text{red}} = N(Q + 1)M^{Q+1}$. Alternatively, the same complexity may be obtained by using a different strategy in which the $L' - Q$ symbols with highest reliabilities are hard-quantized. Compared with the Viterbi or the BCJR algorithm, the complexity of graph-based detection algorithms can be reduced more efficiently since we have no constraints imposed by the trellis structure.

III. MODIFIED GRAPH

In [1], an FG transformation, called *stretching*, is introduced to obtain a cycle-free graph. Denoting by $\mathcal{N}_2(x)$ the set of vari-

---

[3]In this case, if $L > 1 = L'$, the original graph is composed by L independent subgraphs [16] on which L independent instances of the algorithm have to be run.

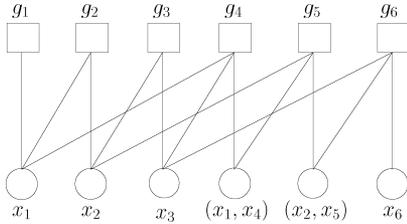[4]A partial representation by using set partitioning can be also adopted.

Fig. 2.   Part of the FG in Fig. 1 after stretching.

able nodes that can be reached from node x through a path of length 2, we can replace any node $z \in \mathcal{N}_2(x)$ with a node representing the pair (x, z). In this way, it is possible for an edge (or a variable node) to become redundant. Hence, it can be removed.

Our goal is to remove a minimum number of edges so that the girth of the transformed graph becomes 6 instead of 4 and the complexity of the resulting SP algorithm remains the same. As a consequence, we adopt the following stretching rule.

- Given an edge e connecting a variable node x to a factor node g, in order to preserve the information lost due to cutting this edge, we choose an arbitrary path $\mathcal{P}(x, g)$ connecting x and g and not involving e and stretch the variable node x to all variable nodes $z \in \mathcal{P}(x, g)$.

Obviously, for a given girth-4 graph, there are different choices of edges such that their removal can lead to a girth-6 graph. However, we verified that the application of the SP algorithm to the resulting girth-6 graphs gives a similar performance, although the minimum required number of iterations can be different, depending on the weight of the interfering symbol whose corresponding edge has been removed.

An example of girth-6 graph, obtained by transforming the girth-4 graph in Fig. 1, is shown in Fig. 2 (considering the relevant part describing only the channel behavior). In order to remove the edge connecting the variable node x_1 to factor node g_5, we chose the path $x_1 - g_4 - x_4 - g_5$ connecting x_1 to g_5 and not involving the above mentioned edge, and stretched x_1 to all variable nodes belonging to this path, i.e., to x_4. The SP algorithm applied to the transformed graph has a cost

$$\mathcal{C}_{\mathrm{mod}} = N(L' + 1 - R)M^{L'+1}$$

where R is the number of edges removed per factor node. Hence, the complexity remains of the same order of magnitude.[5]

A further application of the described stretching rule to obtain a cycle-free graph leads to a graph similar to the Wiberg graph of the channel—the SP algorithm with flooding schedule applied to this graph becomes the state-based message passing described in [13]. We do not further pursue this approach since it gives an unnecessary increase in complexity.

IV. ANALYSIS THROUGH AVERAGE MUTUAL INFORMATION

In [20]–[22], the performance of LDPC codes over binary ISI channels is analyzed by computing the noise tolerance threshold using density evolution [35]. Particular attention is devoted to partial response channels with the aim of designing good LDPC codes for the magnetic channel. Different schedules are consid-

ered in which the number of BCJR steps can be decreased to reduce the overall complexity. A different approach is used in [25]–[27], in which, through the AMI and the so-called extrinsic information transfer (EXIT) charts, an analysis of iterative decoding schemes, including turbo equalization, is described with focus on the convergence behavior. In the case of turbo equalization, the AMI transfer characteristics of the detector for different values of the signal-to-noise ratio and that of the decoder are employed to predict the potential performance improvement through iterative detection/decoding for different channel codes. In particular, in [27], the authors describe a method that can also be used to compute EXIT charts and threshold values in a turbo equalization scheme when the code is an LDPC code and the BCJR algorithm is adopted for detection. In this case, the EXIT curve corresponding to the set of the LDPC check nodes (the so-called check-node decoder (CND)) has to be used jointly with the EXIT curves of the combination of the set of variable nodes (the variable-node decoder (VND)) and the BCJR detector [27].

In this paper, we are first of all interested in the performance of the proposed detection schemes without taking into account the effect of a particular channel code. The term of comparison is represented by the performance of the BCJR algorithm and our aim is to find a computationally nonintensive method to predict if the proposed schemes can reach this limit and the number of necessary iterations. For this reason, we are interested in the AMI, computed by Monte Carlo simulation, at the detector output as a function of the signal-to-noise ratio, in the absence of input AMI since there is no *a priori* knowledge provided to the detector by the channel decoder as in a turbo equalization scheme. By using this tool, we further confirmed the above mentioned property that if the girth of the graph is 6 and the channel is minimum- or maximum-phase, after a few iterations, the algorithm always converges to the performance of the BCJR algorithm. In addition, when a graph transformation through stretching is necessary to ensure convergence, we are able to determine the best choice of edges to be removed.

In the presence of an LDPC code, the method described in [27] can be used to compute EXIT charts and threshold values when the serial BCJR algorithm is used for detection. These values obviously represent a benchmark to assess the performance of the proposed detection algorithms. For these latter schemes, the computation of the EXIT curves is more difficult. In fact, since these algorithms are iterative, the corresponding EXIT curves change with iteration. Let us consider the exchange of extrinsic information in Fig. 3. The EXIT curve of the outer CND can be easily computed as described in [27]. In fact, assuming that all check nodes have degree d_c,[6] the CND EXIT curve, i.e., the AMI at the output of the CND as a function of the AMI at its input, is

$$I_{\mathrm{out,CND}}(I_{\mathrm{in,CND}}) \simeq 1 - J\left(\sqrt{d_c - 1}J^{-1}(1 - I_{\mathrm{in,CND}})\right) \tag{9}$$

where the function $J(\sigma)$ is [27]

$$J(\sigma) = 1 - \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty} e^{-(x-\sigma^2/2)^2/2\sigma^2}\log_2(1 + e^{-x})\,dx. \tag{10}$$

---

[5]We are neglecting the increase in computational complexity at variable nodes related to the need of a marginalization of the messages associated to the pair of symbols, since this computation is lower than that associated to the factor nodes.

[6]A more general irregular check node degree distribution can be taken into account as described in [27]
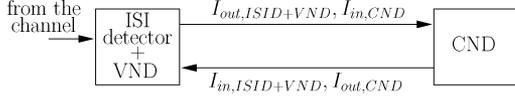
Fig. 3.   Exchange of extrinsic information.

Considering now the combination of the inner ISI detector (ISID) and the VND, since the proposed ISID is iterative and the block ISID + VND retains messages that are used on successive iterations, the corresponding EXIT curve depends not only on the signal-to-noise ratio E_b/N_0, E_b being the received signal energy per information bit and N_0 the one-sided noise power spectral density, but also on the number N of iterations performed. Hence this EXIT curve, that must be computed by Monte Carlo simulations, is a function $I_{\text{out,ISID+VND}}(I_{\text{in,ISID+VND}}, E_b/N_0, N)$ (the dependence on the variable node degree distribution is omitted for simplicity). In the exchange of extrinsic information, the AMI evolves as follows: at the first iteration, the ISID + VND increases the AMI to $I_1^{(1)} = I_{\text{out,ISID+VND}}(0, E_b/N_0, 1)$. After the CND, the mutual information becomes $I_2^{(1)} = I_{\text{out,CND}}(I_1^{(1)})$. At the second iteration, these values will be updated to $I_1^{(2)} = I_{\text{out,ISID+VND}}(I_2^{(1)}, E_b/N_0, 2)$ and $I_2^{(2)} = I_{\text{out,CND}}(I_1^{(2)})$, and so on. In this way, it is possible to compute the *decoding trajectory* whose convergence toward high values of the AMI guarantees low bit-error rate (BER) values. Note that with the aim of finding the threshold values, it is sufficient to consider, along with the CND EXIT curve, only the curve $I_{\text{out,ISID+VND}}(I_{\text{in,ISID+VND}}, E_b/N_0, \infty)$ (in practice, values of $N > 10 \div 15$ are sufficient). In fact, if the tunnel between this curve and the CND EXIT curve is open, sooner or later the AMI will go to one despite the presence of a "bottleneck" for low values of N—the ISI detector will perform "self-iterations" until a tunnel in the EXIT diagram will be opened. In other words, if the channel noise is slightly below the noise threshold, and the number of ISID iterations N is small, then it is possible that the decoding progress will become stuck. However, if N is increased, then the bottleneck opens up, and AMI can continue to improve. The results of this analysis will be shown in the next section.

V. NUMERICAL RESULTS

In this section, the performance of the proposed detection schemes is assessed by computer simulations in terms of BER versus E_b/N_0. We also provide some examples of the above mentioned analysis through AMI and EXIT charts. For each considered channel, the performance of the BCJR algorithm is given as a benchmark. Although improper, since BCJR and p-BCJR algorithms are also instances of the SP algorithm, in the following figures the curves corresponding to the proposed algorithms are labeled "SP."

In Fig. 4, we consider a sparse ISI channel characterized by $L = 5$ and

$$\boldsymbol{f} = (0.408, 0, 0, 0, 0.816, 0.408)$$

($L' = 2$) whose corresponding FG has girth 6. In fact, the set of differences between the indexes of the channel interferers is
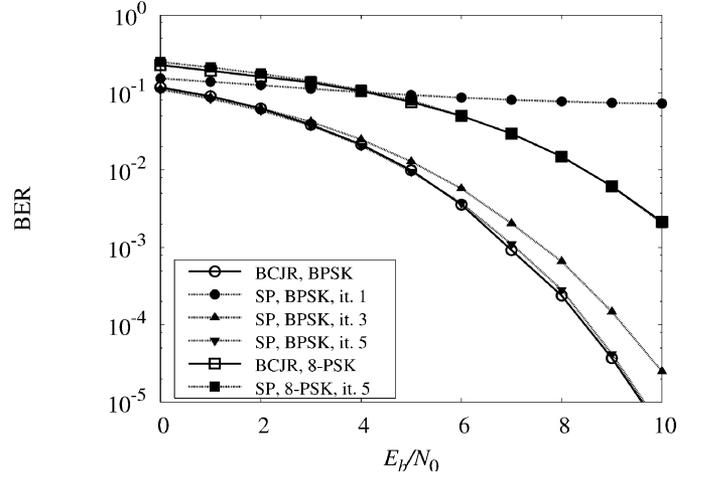

Fig. 4.   Performance for a sparse channel.

$\{1, 4, 5\}$ and hence they are all different (see condition b)). Uncoded binary phase-shift keying (BPSK) and 8-PSK modulations are considered. It can be observed that the performance of the SP algorithm applied to the original graph converges in five iterations to the optimal performance of the BCJR for both modulation formats. In addition, the dramatic performance improvement from the first and the third iterations can be noted. Similar considerations also hold for multiamplitude modulation formats such as quadrature amplitude modulation (QAM) schemes.

In Fig. 5, the effect of the application of the complexity reduction techniques to the BCJR and the proposed SP algorithm is analyzed. The considered girth-6 minimum-phase channel has $L = 5$ and coefficients

$$\boldsymbol{f} = (0.83, 0, 0, 0, 0.25, 0.5).$$

Also in this case, an uncoded BPSK modulation is adopted. A reduced-complexity 16-state BCJR algorithm may be obtained only by defining a reduced state as $(a_{n-1}, a_{n-2}, a_{n-3}, a_{n-4})$ and computing the branch metrics by recovering the symbol a_{n-5} in the survivor history [23], [24]. Hence, the only possibility we have is the truncation, in the trellis definition, of the farthest symbol. On the contrary, in the case of the proposed SP algorithm we are not constrained by the trellis definition and therefore we may truncate the interfering symbol with lower weight, i.e., symbol a_{n-4}. In the computation of the messages at factor nodes, this symbol can be substituted with a decision made on the basis of the messages on the graph. The advantage of about 1 dB with respect to the reduced-complexity BCJR is highlighted by Fig. 5.

We now consider the so-called $\text{E}^2\text{PR}4$ magnetic channel, i.e., an ISI channel with $L = 4$, $L' = 3$, and

$$\boldsymbol{f} = \frac{1}{\sqrt{10}}(1, 2, 0, -2, -1).$$

An uncoded binary pulse amplitude modulation (PAM) with symbols $x_k \in \{\pm 1\}$ is considered. The corresponding graph, shown in the upper part of Fig. 1, has girth 4. In this case, the SP algorithm, if directly applied to this graph, does not converge to the performance of the BCJR, as shown in Fig. 6, where a loss of about 1 dB is observed, even for a large number of iterations.
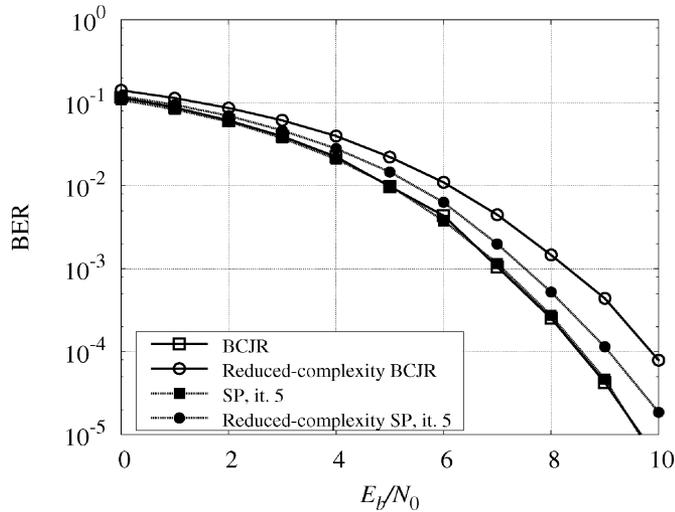
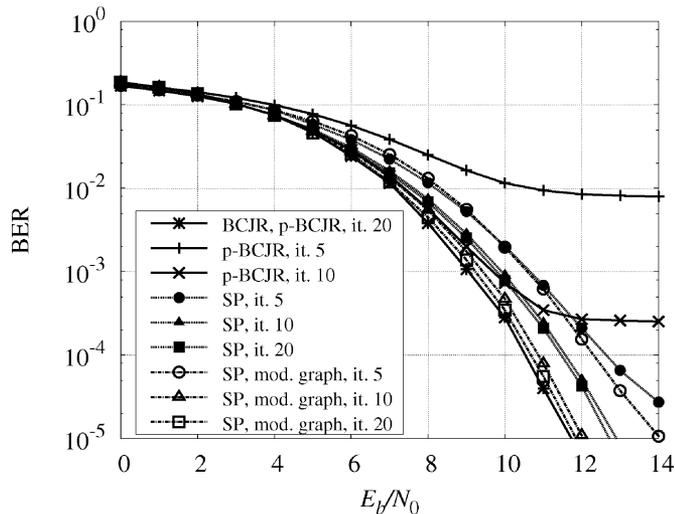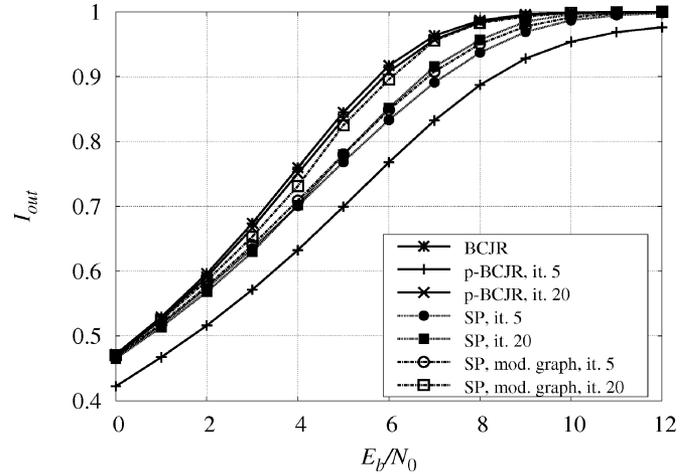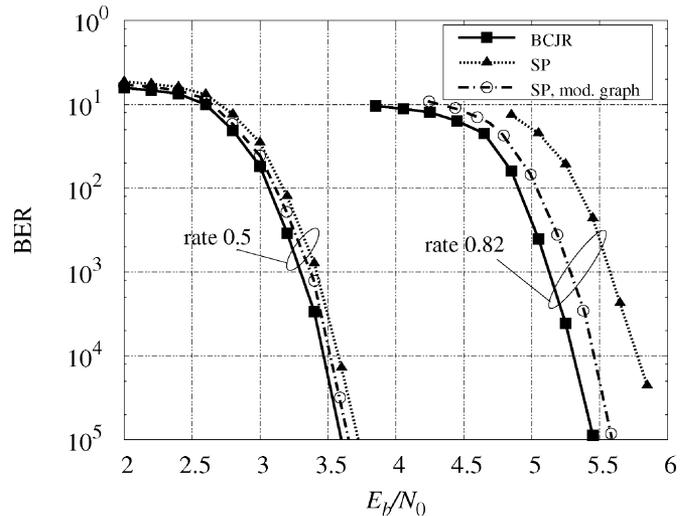Fig. 5.   Performance in the case of complexity reduction.

Fig. 6.   Performance of the SP algorithm on a modified graph.

Fig. 7.   Average mutual information at the detector output in the case of $I_{\text{in}} = 0$.

Fig. 8.   BER in the case of application to LDPC codes. The E^2 PR4 magnetic channel is considered.

Therefore, a graph transformation is necessary. By using the described stretching technique, we obtain the graph in Fig. 2. On this graph, the SP algorithm converges in about 10 iterations to the optimal performance. In Fig. 6, the performance of the p-BCJR algorithm is also shown. We may observe that, although for a sufficiently large number of iterations this algorithm has an optimal performance, at least in the considered BER range, for five and ten iterations it performs worse than the SP algorithm and exhibits an error floor. This floor has been already observed in [13], and a precoder has been introduced to overcome this problem. Its presence is due to the structure of the FGs. In fact, in the graphs of Figs. 1 and 2, the message related to a variable node rapidly propagates to other nodes, whereas in the Wiberg graph this propagation, in the case of parallel schedule, is slower.

The behavior of the BER curves for the considered algorithms is related to that of the AMI at the detector output as a function of the signal-to-noise ratio (see Fig. 7). We are considering an uncoded transmission and, therefore, we are interested in the

case of absence of *a priori* knowledge on the information symbols, i.e., $I_{\text{in}} = 0$. The fact that the p-BCJR for five iterations performs worse than the SP algorithm, on the original and the modified graph for the same number of iterations, is verified in terms of I_{out}, along with the convergence, for 20 iterations, of the p-BCJR and the SP algorithm on the modified graph to the performance of the BCJR.

We now consider the case of application to combined detection and decoding in the presence of a LDPC code. Due to the structure of a LDPC code, an interleaver is not necessary. Two magnetic channels are considered, namely the $E^2 PR4$ channel in Fig. 8, and the EPR4 channel in Fig. 9. This latter channel is characterized by $L = 3$, $L' = 3$, and

$$f = \frac{1}{2}(1, 1, -1, -1).$$

The modulation format is the previously mentioned binary PAM. Two LDPC codes are considered, namely, a regular $(3, 6)$ code of rate 0.5 and codeword length 4000 and an irregular

TABLE I
THRESHOLDS FOR THE CONSIDERED SYSTEMS

	rate 0.5			rate 0.82		
	BCJR	SP mod. graph	SP	BCJR	SP mod. graph	SP
EPR4	2.24 dB	2.24 dB	2.25 dB	4.07 dB	4.08 dB	4.29 dB
E^2PR4	2.76 dB	2.78 dB	2.81 dB	4.92 dB	5.04 dB	5.35 dB
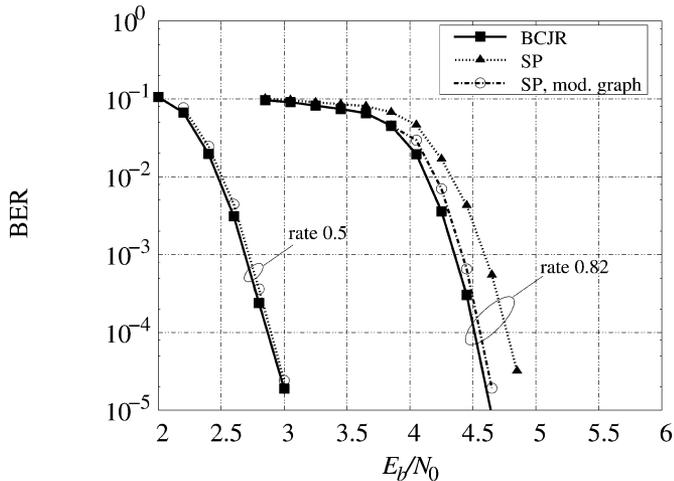
Fig. 9.   BER in the case of application to LDPC codes. The EPR4 magnetic channel is considered.

Fig. 10.   EXIT charts with iterative decoding trajectories for the LDPC code with rate 0.5 and the E^2PR 4 channel.

code of rate 0.82 and codeword length 4095.[7] In the case of the SP algorithm on the original or the modified graph, a maximum number of 150 iterations is allowed. When detection is performed by means of the BCJR algorithm, a BCJR step is executed each iteration of the LDPC decoder (for a maximum number of allowed iterations of 150). This choice has been shown to be optimal in [13]. Figs. 8 and 9, demonstrate that in the case of a powerful channel code, the proposed SP algorithm has a BER performance very close to the optimal performance even on the original girth-4 graph. For the rate-0.82 LDPC code, the SP algorithm run over the original graph exhibits a performance loss of about 0.5 dB in the case of the E^2PR4 channel and 0.3 dB in the case of the EPR4 channel, whereas this loss is reduced to about 0.15 and 0.05 dB, respectively, for the SP algorithm working on the modified graph. Hence, side information provided in the iterative detection/decoding process by the decoder of a powerful code helps the detector convergence even in the presence of cycles of length 4 in the part of the graph describing the ISI channel.

These results are confirmed by the threshold values, computed for the codes with the above mentioned degree distributions, by means of the described analysis based on EXIT charts, and shown in Table I: when the code with rate 0.5 is used, it is not necessary to adopt graph transformations, whereas when the code with rate 0.82 is adopted, the SP algorithm on the modified graph has a negligible performance loss with respect to the BCJR algorithm.

Finally, in Fig. 10, the EXIT charts and the decoding trajectories for the rate-0.5 code, the E^2PR4 channel, and $E_b/N_0 =$

3.5 dB is shown. The SP algorithm on the original girth-4 graph is considered. The CND EXIT curve (dotted line) and the ISID + VND curves (solid lines) for $N = 1, 2, 3$ iterations are shown (for $N \geq 3$ the EXIT curves practically coincide). As already mentioned, although for $N = 1$ there is a bottleneck, the iterative process converges since the tunnel is "open" for $N \to \infty$.

VI. CONCLUSION

In this paper, simple detection schemes for ISI channels have been proposed. These algorithms have been derived from the application of the SP algorithm to an FG representing the joint *a posteriori* distribution of the transmitted symbols or to a modified girth-6 graph. The complexity of the proposed algorithms depends on the number of nonzero channel interferers only and therefore they are useful in the case of sparse channels. The performance of the proposed algorithms converges to that of the BCJR algorithm when the FG has girth 6 or a powerful channel code is used. If this is not the case, the described simple graph transformation guarantees the performance convergence of the SP algorithm. Hence, it has been shown that in some cases, the application of the proposed schemes can be favorable with respect to the BCJR algorithm.

[7]For this code, the variable and check degree distributions, defined as in [5], are $\lambda(x) = x^2$ and $\rho(x) = 0.33x^{15} + 0.67x^{16}$, respectively.
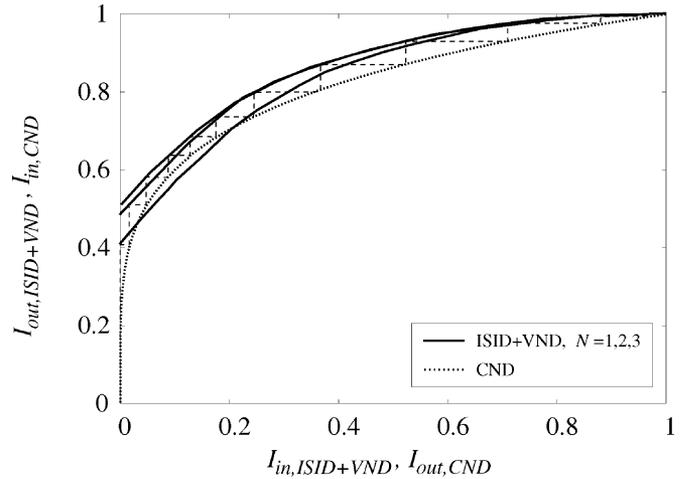
REFERENCES

[1] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[2] R. G. Gallager, "Low density parity check codes," *IEEE Trans. Inf. Theory*, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.

[3] D. J. C. MacKay, "Good error correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, Feb. 1999.

[4] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. Spielman, and V. Stemann, "Practical loss-resilient codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.

[5] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

[6] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 1996.

[7] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 325–343, Mar. 2000.

[8] A. P. Worthen and W. E. Stark, "Unified design of iterative receivers using factor graphs," *IEEE Trans. Inf.. Theory*, vol. 47, no. 2, pp. 843–849, Feb. 2001.

[9] R. Nuriyev and A. Anastasopoulos, "Pilot-symbol-assisted coded transmission over the block-noncoherent AWGN channel," *IEEE Trans. Commun.*, vol. 51, no. 1, pp. 953–963, Jun. 2003.

[10] G. Colavolpe, "LDPC codes over channels with memory," in *Proc. IEEE Int. Conf. Communications*, Paris, France, Jun. 2004, pp. 757–761.

[11] ——, "Design and performance of turbo Gallager codes," *IEEE Trans. Commun.*, vol. 52, no. 11, pp. 1901–1908, Nov. 2004.

[12] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo-equalization," *Europ. Trans. Telecommun.*, vol. 6, no. 5, pp. 507–511, Sep./Oct. 1995.

[13] B. M. Kurkoski, P. H. Siegel, and J. K. Wolf, "Joint message-passing decoding of LDPC codes and partial-response channels," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1410–1422, Jun. 2002.

[14] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. IT-20, no. 2, pp. 284–287, Mar. 1974.

[15] N. Benvenuto and R. Marchesani, "The Viterbi algorithm for sparse channels," *IEEE Trans. Commun.*, vol. 44, no. 3, pp. 287–289, Mar. 1996.

[16] N. C. McGinty, R. A. Kennedy, and P. Hoeher, "Parallel trellis Viterbi algorithm for sparse channels," *IEEE Commun. Lett.*, vol. 2, no. 5, pp. 143–145, May 1998.

[17] F. R. Kschischang and B. J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 219–231, Feb. 1998.

[18] T. V. Souvignier, M. Öberg, P. H. Siegel, R. E. Swanson, and J. K. Wolf, "Turbo decoding for partial response channels," *IEEE Trans. Commun.*, vol. 48, no. 8, pp. 1297–1308, Aug. 2000.

[19] T. Mittelholzer, A. Dholakia, and E. Eleftheriou, "Reduced-complexity decoding of low-density parity check codes for generalized partial response channels," *IEEE Trans. Magn.*, vol. 37, no. 2, pp. 721–728, Mar. 2001.

[20] A. Thangaraj and S. W. McLaughlin, "Thresholds and scheduling for LDPC-coded partial response channels," *IEEE Trans. Magn.*, vol. 38, no. 5, pp. 2307–2309, Sep. 2002.

[21] 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.

[22] A. Kavčić, X. Ma, and M. Mitzenmacher, "Binary intersymbol interference channels: Gallager codes, density evolution, and code performance bounds," *IEEE Trans. Inf. Theory*, vol. 49, no. 7, pp. 1636–1652, Jul. 2003.

[23] G. Colavolpe, G. Ferrari, and R. Raheli, "Reduced-state BCJR-type algorithms," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 5, pp. 848–859, May 2001.

[24] P. Hoeher, "Kohärenter Empfang Trelliscodierter PSK-Signale auf Frequenzselektiven Mobilfunkkanälen—Entzerrung, Decodierung und Kanalparameterschätzung," Ph.D. dissertation (in German), University of Kaiserslautern; VDI-Verlag, 1990.

[25] S. ten Brink, "Designing iterative decoding schemes with the extrinsic information transfer chart," *AEU Int. J. Electronic. Commun.*, vol. 54, no. 6, pp. 389–398, Dec. 2000.

[26] ——, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[27] 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.

[28] G. D. Forney, Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inf. Theory*, vol. IT-18, no. 3, pp. 284–287, May 1972.

[29] W. Koch and A. Baier, "Optimum and sub-optimum detection of coded data disturbed by time-varying intersymbol interference," in *Proc. IEEE Global Telecommun. Conf.*, San Diego, CA, Dec. 1990, pp. 807.5.1–5.

[30] P. Roberston, E. Villebrun, and P. Hoeher, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *Europ. Trans. Telecommun.*, vol. 8, no. 2, pp. 119–125, Mar./Apr. 1997.

[31] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA: Morgan Kaufmann, 1988.

[32] M. V. Eyuboğlu and S. U. Qureshi, "Reduced-state sequence estimation with set partitioning and decision feedback," *IEEE Trans. Commun.*, vol. 38, no. 1, pp. 13–20, Jan. 1988.

[33] A. Duel-Hallen and C. Heegard, "Delayed decision feedback estimation," *IEEE Trans. Commun.*, vol. 37, no. 5, pp. 428–436, May 1989.

[34] P. R. Chevillat and E. Eleftheriou, "Decoding of trellis-encoded signals in the presence of intersymbol interference and noise," *IEEE Trans. Commun.*, vol. 36, no. 7, pp. 669–676, Jul. 1989.

[35] T. Richardson and R. Urbanke, "The capacity of low density parity check codes under message passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

Giulio Colavolpe was born in Cosenza, Italy, in 1969. He received the Dr. Ing. degree in telecommunications engineering (*cum laude*) from the University of Pisa, Pisa, Italy, in 1994 and the Ph.D. degree in information technology from the University of Parma, Parma, Italy, in 1998.

Since 1997, he has been at the University of Parma, where he is now an Associate Professor of Telecommunications. In 2000, he was a Visiting Scientist at the Institut Eurécom, Valbonne, France. His main research interests include digital transmission theory, channel coding, and signal processing. His research activity has led to more than 70 scientific publications in leading international journals and conference proceedings and a few industrial patents. He is also coauthor of the book *Detection Algorithms for Wireless Communications* (New York: Wiley, 2004).

Gianpietro Germi was born in Parma, Italy, in 1978. He received the Dr. Ing. degree in telecommunications engineering (*cum laude*) from the University of Parma, in 2003.

Now he is with Selta S.p.A., Roveleto di Cadeo, Piacenza, Italy.