

# Iterative Detection for Channels With Memory

*By examining various approaches and techniques, this paper aims to help designers to create improved algorithms for communications over noisy channels with memory.*

By ACHILLEAS ANASTASOPOULOS, Member IEEE, KEITH M. CHUGG, Member IEEE, GIULIO COLAVOLPE, Member IEEE, GIANLUIGI FERRARI, Member IEEE, AND RICCARDO RAHELI, Member IEEE

**ABSTRACT** | In this paper, we present an overview on the design of algorithms for iterative detection over channels with memory. The starting point for all the algorithms is the implementation of soft-input soft-output maximum *a posteriori* (MAP) symbol detection strategies for transmissions over channels encompassing unknown parameters, either stochastic or deterministic. The proposed solutions represent effective ways to reach this goal. The described algorithms are grouped into three categories: i) we first introduce algorithms for adaptive iterative detection, where the unknown channel parameters are explicitly estimated; ii) then, we consider finite-memory iterative detection algorithms, based on *ad hoc* truncation of the channel memory and often interpretable as based on an implicit estimation of the channel parameters; and iii) finally, we present a general detection-theoretic approach to derive optimal detection algorithms with polynomial complexity. A few illustrative numerical results are also presented.

**KEYWORDS** | Adaptive iterative detection; channels with memory; finite-memory detection; iterative detection/decoding; message passing detection algorithms; polynomial complexity detection; trellis-based detection

## I. INTRODUCTION

Over the last decades, the need for increasingly fast and reliable transmissions has motivated a significant research activity for the derivation of effective, yet simple, detection

algorithms. While the Viterbi algorithm, invented in the late 1960s [1], [2], found ever-increasing fame and applications in the following three decades, the *soft-output* algorithms implementing the maximum *a posteriori* (MAP) symbol detection strategy, and among them the *forward-backward* (FB) algorithm [3], [4], were considered as an interesting, yet marginal, theoretical contribution for many years. The advent of “turbo codes” and their suboptimal iterative decoding, introduced by the landmark work of Berrou and Glavieux in the early 1990s [5], [6], has revolutionized digital communications and, therefore, the design of detection algorithms. The birth of turbo codes has led to the rebirth of low density parity-check (LDPC) codes, introduced by Gallager in his Ph.D. thesis in the early 1960s [7]–[9] and the rediscovery of the relevant iterative *belief propagation* algorithm [10]–[12].

The iterative (turbo) decoding principle has then been extended to the detection process and generalized based on a standard set of rules for exchanging and updating soft-decision information between locally optimal processors. While the original turbo decoder is typically viewed as two processors running the FB algorithm and exchanging soft-decision information, the generalized view of this approach, deriving from Wiberg’s Ph.D. dissertation [13], [14] and further advances, is now commonly described as passing soft-decision messages on *cyclic graphical models*.

This paper provides an overview on the application of iterative processing to detection problems involving unknown parameters in addition to the unknown data to be detected. This is an aspect of considerable theoretical richness and practical interest. For example, one of the first concerns regarding the practicality of turbo codes was the ability of a receiver to estimate and track channel parameters at very low signal-to-noise ratio (SNR). Several generalizations of the message-passing algorithm have been developed, which address the issue of detection over channels with memory while attempting to maintain

Manuscript received April 21, 2006; revised January 27, 2007.

**A. Anastopoulos** is with the Electrical Engineering and Computer Science (EECS) Department, University of Michigan, Ann Arbor, MI 48109-2122 USA (e-mail: anastas@umich.edu).

**K. M. Chugg** is with the Electrical Engineering Department—Systems Division, Communication Sciences Institute, University of Southern California, Los Angeles, CA 90089-2565 USA (e-mail: chugg@usc.edu).

**G. Colavolpe**, **G. Ferrari**, and **R. Raheli** are with the Dipartimento di Ingegneria dell’Informazione, Università degli Studi di Parma, I-43100 Parma, Italy (e-mail: giulio.colavolpe@unipr.it; gianluigi.ferrari@unipr.it; riccardo.raheli@unipr.it).

Digital Object Identifier: 10.1109/JPROC.2007.896511

reasonable complexity. The exact update rules and their associated theoretical complexity are highly dependent on the parameter model assumed. One approach is to include parameter estimation and tracking into the message update process. Another is to attempt to average out the effects of the parameters under some statistical model. It has also been shown, somewhat surprisingly, that if one deviates substantially from the FB algorithm structure, there exists an optimal algorithm with *polynomial* (with respect to the channel memory) complexity for some simple channels. In this paper, we give an overview of iterative processing algorithms stemming from MAP symbol detection strategies. We also try to highlight commonalities and differences among these algorithms. We point out that other approaches to detection over channels with memory are possible, such as, for example, expectation-maximization (EM) detection [15] or space-alternating generalized expectation-maximization (SAGE) detection [16]. We will not consider, however, any of the latter detection strategies, and the interested reader is invited to compare the solutions presented in this paper with other relevant detection strategies well-documented in the literature.

The structure of this paper can be summarized as follows.

- After setting up the system model, the sum product algorithm on factor graphs is reviewed, followed by the derivation of the *optimal* FB algorithm structure for *finite-memory* channels.
- Then, we propose possible approaches to perform MAP symbol detection in general communication scenarios where the channel memory may not be strictly finite.
  - We first introduce the concept of *adaptive iterative detection*, i.e., joint data detection and (explicit) channel estimation.
  - We then consider *finite-memory iterative detection*, i.e., data detection without explicit channel estimation, and we compare directly these schemes with those introduced at the previous item.
  - Finally, we show that in some cases optimal detection can be carried out with *polynomial complexity*.

We underline that the focus of this tutorial paper is on the *algorithmic structure*, and its ambition is to provide the reader with a unifying view on the important problem of iterative detection over channels with memory. Only a few *illustrative* numerical results are presented at the end. The interested reader, however, can find much more accurate performance analyses in the papers where the described algorithms were originally introduced.

## II. SYSTEM MODEL

We consider a generic time-continuous digital transmission system, depicted in Fig. 1. A sequence of independent

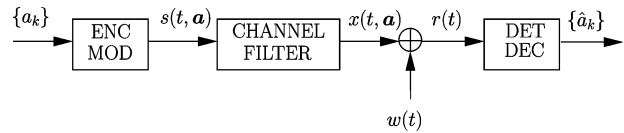


Fig. 1. Communication system.

and identically distributed (i.i.d.)  $M$ -ary information symbols  $\{a_k\}$  (with  $a_k \in \mathcal{A}$ ,  $\mathcal{A}$  being the symbol alphabet of cardinality  $|\mathcal{A}| = M$ ) are transmitted successively from epoch 0 to epoch  $K - 1$ . A sequence of information symbols is denoted in vector notation as

$$\mathbf{a}_{k_1}^{k_2} = (a_{k_1}, a_{k_1+1}, \dots, a_{k_2}) \quad k_2 \geq k_1.$$

For brevity, the entire sequence is denoted by  $\mathbf{a}$ . This sequence is input to the encoder and modulator. The coded and modulated signal is denoted as  $s(t, \mathbf{a})$  to emphasize its dependence on the information sequence. The channel is viewed as a noiseless filter (possibly stochastic) with output signal  $x(t, \mathbf{a}, \Theta)$ , where  $\Theta \in \Lambda$  denotes the set of unknown parameters that need to be (implicitly or explicitly) estimated during the detection process, rendered noisy by the addition of white noise  $w(t)$ . The domain  $\Lambda$  of the parameters  $\Theta$  depends on the considered communication system. The received signal

$$r(t) = x(t, \mathbf{a}, \Theta) + w(t) \quad (1)$$

is observed by the demodulation and decoding block, which outputs a sequence of decisions  $\{\hat{a}_k\}$ .

Without loss of generality, we assume that the encoder/modulator block in Fig. 1 is a system which evolves, upon receiving at its input the information sequence  $\mathbf{a}$ , through a sequence of states  $\{\mu_0, \mu_1, \dots\}$ . In many communication schemes, the encoder/modulator can be described as a time-invariant finite-state machine (FSM) (e.g., trellis coded modulation (TCM) [17] or continuous phase modulation (CPM) [18]). In this case, the state  $\mu_k$  belongs to a set of finite cardinality and a *time-invariant* “next-state” function  $\text{ns}(\cdot, \cdot)$  describes the evolution of the system as

$$\mu_{k+1} = \text{ns}(a_k, \mu_k). \quad (2)$$

Therefore, the evolution of the encoder/modulator can be described through a *trellis diagram*, in which there are  $M$  exiting branches (in correspondence with  $M$  different information symbols) from each state. A trellis branch

corresponds to a transition, defined as  $t_k \triangleq (a_k, \mu_k)$ . In the rest of the paper, the initial state  $\mu_0$  is assumed to be known.

In the case of a linear block code, a trellis representation is possible, but the trellis is time-variant, both in terms of states and branches [4]. In this case, the evolution of the encoder/modulator could be described by a *time-variant* next-state function  $ns_k(\cdot, \cdot)$ . A *Tanner graph* [13], [19] representation for a linear block code—where the parity checks determine the structure of the graph—may be more appealing, especially if the parity check equations involve a few code symbols as, for instance, in LDPC codes [9], [20]–[22].

By means of a *discretization process*, the received signal  $r(t)$  can be converted into a time-discrete sequence  $\mathbf{r}$  [23]. In particular, we assume that there is one observable  $r_k$  per information symbol  $a_k$ , or, formally,  $\mathbf{r} = \mathbf{r}_0^{K-1}$ , with a notation similar to that used for the information sequence. In the following, we consider a discretized observation model of the form

$$\mathbf{r} = \mathbf{x}(\mathbf{a}, \Theta) + \mathbf{w} \quad (3)$$

based on a sampling rate of one sample per symbol, which may be practically sufficient in many cases—the vector  $\mathbf{w}$  is the discretized version of the additive noise process  $w(t)$  in (1). In a more general setting, there may be two or more elements of  $\mathbf{r}$  per information symbol  $a_k$ , e.g., when a convolutional code or a time-varying channel is considered.

In the remainder of this paper, we will often consider two relevant channel models for wireless communications: i) the phase-uncertain channel and ii) the flat fading channel. In both cases, we will assume linear modulations, matched receiver filtering, Nyquist overall pulses, and channel variations slow enough so that no intersymbol interference arises and a sampling rate of one sample per information symbol is adequate. Under these assumptions, the general model (3) can be specialized as follows.

- In the phase-uncertain channel, denoting by  $\mathbf{c} = \{c_k\}_{k=0}^{K-1}$  the sequence of transmitted code symbols corresponding to the information sequence  $\mathbf{a}$ ,<sup>1</sup> the discrete-time observation at epoch  $k$  can be given as

$$r_k = c_k e^{j\phi_k} + w_k \quad (4)$$

where  $\{\phi_k\}$  is a discrete-time channel phase process (with proper statistics) and  $w_k$  is an

additive white Gaussian noise (AWGN) sample of variance  $\sigma^2$ . In case of trellis codes, the symbol  $c_k$  can be interpreted as the discrete-time output of the encoder/modulator FSM according to a suitable “output” function  $o(\cdot, \cdot)$ , such that  $c_k = o(a_k, \mu_k)$ —this function was not clearly introduced at the beginning of this section because in the communication system model in Fig. 1 the output of the encoder/modulator is represented as a continuous-time signal. In this case,  $\Theta = \{\phi_k\}_{k=0}^{K-1}$ .

- In the fading channel case, the observable at epoch  $k$  can be expressed as

$$r_k = f_k c_k + w_k \quad (5)$$

where  $\{f_k\}$  is a sequence of realizations of complex zero mean Gaussian random variables with proper autocorrelation function—several models have been proposed in the literature [24]. In this case,  $\Theta = \{f_k\}_{k=0}^{K-1}$ .

### III. PRELIMINARIES

#### A. Factor Graphs and the Sum-Product Algorithm

A factor graph (FG) is a bipartite graph which expresses the way a complicated joint probability mass function (pmf) or a joint probability density function (pdf) of many variables factors into the product of *local* functions (not necessarily pmf's or pdf's) [12]. Let  $V = \{v_1, \dots, v_N\}$  denote a set of variables and  $F(V)$  a multivariate function. Let  $V_1, \dots, V_m$  denote subsets of  $V$ . We say that  $F(V)$  admits a factorization with supports  $V_1, \dots, V_m$ , if  $F(V)$  can be written as the product of the functions  $\{F_j : j = 1, \dots, m\}$ , where  $F_j$  has the variables in  $V_j$  as arguments. The FG representing the factorization  $F = \prod_j F_j$  is a *bipartite* graph  $\mathcal{G} = \{\mathcal{V}, \mathcal{F}, \mathcal{E}\}$ , where nodes in  $\mathcal{V}$  (variable nodes) are associated with the variables  $v_i \in V$ , nodes in  $\mathcal{F}$  (factor nodes) are associated with the functions  $F_j$ , and there exists an edge  $e \in \mathcal{E}$  joining  $v_i$  and  $F_j$  if and only if  $v_i \in V_j$  (i.e., if  $v_i$  is an argument of  $F_j$ ).

Let  $F(V)$  be a pmf. Then, if the FG corresponding to the factorization of  $F$  has no cycles,<sup>2</sup> the marginal pmf's can be computed exactly in a finite number of steps by the sum-product algorithm (SPA) [12]. The SPA is defined by the computation rules at variable and at factor nodes, and by a suitable node activation schedule. Denoting by  $\mu_{v_i \rightarrow F_j}(v_i)$  a message sent from the variable node  $v_i$  to the factor

<sup>1</sup>Without loss of generality, it is assumed that the code sequence has the same length of the information sequence—the redundancy introduced by the encoder is accounted for by expanding the symbol cardinality.

<sup>2</sup>A cycle is a closed path in the graph and its length is defined as the corresponding number of path edges.

node  $F_j$ , by  $\mu_{F_j \rightarrow v_i}(v_i)$  a message in the opposite direction, and by  $\mathcal{A}_i$  the set of functions  $F_j$  having  $v_i$  as argument, the message computations performed at variable and factor nodes are, respectively [12]

$$\mu_{v_i \rightarrow F_j}(v_i) = \prod_{H \in \mathcal{A}_i \setminus \{F_j\}} \mu_{H \rightarrow v_i}(v_i) \quad (6)$$

$$\mu_{F_j \rightarrow v_i}(v_i) = \sum_{\sim\{v_i\}} \left[ F_j(\{w \in V_j\}) \prod_{w \in V_j \setminus \{v_i\}} \mu_{w \rightarrow F_j}(w) \right] \quad (7)$$

where, following the notation of [12], we indicate by  $\sum_{\sim\{v_i\}}$  the *summary* operator, i.e., a sum over all variables excluding  $v_i$ . The final marginals are given by

$$\sum_{\sim\{v_i\}} F(V) = \prod_{H \in \mathcal{A}_i} \mu_{H \rightarrow v_i}(v_i). \quad (8)$$

The SPA above is a special case of a general algorithm known as Belief Propagation [25], or message-passing [26]. These are terms used to describe algorithms where processing nodes exchange messages, combine these messages, and then marginalize these combined messages over local constraints (or factors). These algorithms exploit only the semi-ring properties of the marginalizing and combining operators [13], [27], most notably the generalized distributive law [26]. Therefore one can consider the above messages in the logarithmic domain by replacing the marginalizing and combining operators from sum-product to  $\max^*$ -sum, with the  $\max^*\{\cdot\}$  operator defined as

$$\max_u^* \{f(u)\} \triangleq \log \sum_u e^{f(u)}$$

where  $u$  belongs to a discrete set and  $f(\cdot)$  is a given function of  $u$ . Similarly, one can substitute the sum-product operators with max-product and the corresponding message exchange algorithm will eventually evaluate “marginals” of the form  $\max_{\sim\{v_i\}} F(V)$ . For a detection problem, such marginals correspond to symbol-level soft decisions that are consistent with optimal MAP *sequence* detection. For this reason, variations on the message-passing algorithm, and iterative processing in general, are often referred to in terms of the marginalizing and combining operators (e.g., marg.-comb. = max-sum).

In the following we examine the case of finite-memory channels, where the factorization of the pmf of interest results in an acyclic linear graph and the SPA takes the form of a simple forward and backward message update algorithm.

## B. Soft-Input Soft-Output MAP Detection for Finite-Memory Channels

We now derive the optimal soft-output MAP symbol detection algorithm for systems exhibiting a *finite* overall memory. We begin by introducing statistical notions of causality and finite memory, which represent minimal conditions for optimality.

A *causality condition* for the considered communication system can be formulated in terms of statistical dependence of the observation sequence  $\mathbf{r}_0^k$ , up to epoch  $k$ , on the information sequence. Accordingly, a system is causal if

$$p(\mathbf{r}_0^k | \mathbf{a}) = p(\mathbf{r}_0^k | \mathbf{a}_0^k). \quad (9)$$

Similarly, a *finite-memory condition* can be formulated, in statistical terms, as follows:

$$p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_0^k) = p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_{k-C}^k, \mu_{k-C}) \quad (10)$$

where  $C$  is a suitable *finite-memory parameter* and  $\mu_{k-C}$  represents the state, at epoch  $k-C$ , of the encoder/modulator.

The encoder/modulator block in Fig. 1 can often be decomposed into the cascade of an encoder and a memoryless mapper. In this case, the previous conditions imply analogous relations between the observation sequence  $\mathbf{r}$  and *code* sequence  $\mathbf{c} = \mathbf{c}_0^{N-1}$ . Accordingly, causality and finite-memory conditions can be formulated as follows:

$$p(\mathbf{r}_0^k | \mathbf{c}) = p(\mathbf{r}_0^k | \mathbf{c}_0^k) \quad (11)$$

$$p(r_k | \mathbf{r}_0^{k-1}, \mathbf{c}_0^k) = p(r_k | \mathbf{r}_0^{k-1}, \mathbf{c}_{k-C}^k). \quad (12)$$

We remark, however, that these conditions involve the transmission channel only and, in general, *do not* imply (9) and (10). A case of interest may be that of a linear block code followed by a memoryless modulator. In particular, a linear block code is not guaranteed to be causal and finite-memory<sup>3</sup> so that the channel causality (11) and finite memory (12) do not imply the system causality (9) and finite-memory condition (10).

The FB algorithm allows to implement the MAP *symbol* detection criterion, since it explicitly computes the APPs  $\{P\{a_k | \mathbf{r}\}\}$ . The FB algorithm is based on: i) a *forward* recursion, during which forward state metrics (typically denoted with the notation  $\alpha$ ) are recursively computed;

<sup>3</sup>Block-wise causality and finite-memory must be indeed satisfied.

ii) a *backward* recursion, during which backward state metrics (typically denoted with the notation  $\beta$ ) are recursively computed; and iii) a completion operation, which, at each epoch  $k$ , generates the corresponding APP  $P\{a_k|\mathbf{r}\}$  by properly averaging the forward state metrics, associated with the initial states of the  $k$ th trellis section, with the backward state metrics, associated with the final states of the same trellis section.

Based on the causality and finite-memory conditions (9) and (10), the following probabilistic derivation of an FB algorithm is obtained by marginalization:

$$\begin{aligned}
 P\{a_k|\mathbf{r}\} &= \sum_{\mathbf{a}_{k-C}^{k-1}} \sum_{\mu_{k-C}} P\{\mathbf{a}_{k-C}^k, \mu_{k-C}|\mathbf{r}\} \\
 &\propto \sum_{\mathbf{a}_{k-C}^{k-1}} \sum_{\mu_{k-C}} p(\mathbf{r}_{k+1}^{K-1}|\mathbf{r}_0^k, \mathbf{a}_{k-C}^k, \mu_{k-C}) \\
 &\quad \times p(r_k|\mathbf{r}_0^{k-1}, \mathbf{a}_{k-C}^k, \mu_{k-C}) p(\mathbf{r}_0^{k-1}|\mathbf{a}_{k-C}^k, \mu_{k-C}) \\
 &\quad \cdot P\{\mathbf{a}_{k-C}^k, \mu_{k-C}\} \quad (13)
 \end{aligned}$$

where Bayes and chain rules have been used, and  $\propto$  denotes proportionality. The causality and finite-memory properties (9) and (10) allow one to write<sup>4</sup>

$$\begin{aligned}
 p(\mathbf{r}_{k+1}^{K-1}|\mathbf{r}_0^k, \mathbf{a}_{k-C}^k, \mu_{k-C}) &= p(\mathbf{r}_{k+1}^{K-1}|\mathbf{r}_0^k, \mathbf{a}_{k-C+1}^k, \mu_{k-C+1}) \\
 p(\mathbf{r}_0^{k-1}|\mathbf{a}_{k-C}^k, \mu_{k-C}) &= p(\mathbf{r}_0^{k-1}|\mathbf{a}_{k-C}^{k-1}, \mu_{k-C}).
 \end{aligned}$$

Recalling the independence of the information symbols, (13) can be rewritten as follows:

$$\begin{aligned}
 P\{a_k|\mathbf{r}\} &\propto \sum_{\mathbf{a}_{k-C}^{k-1}} \sum_{\mu_{k-C}} p(\mathbf{r}_{k+1}^{K-1}|\mathbf{r}_0^k, \mathbf{a}_{k-C+1}^k, \mu_{k-C+1}) \\
 &\quad \times p(r_k|\mathbf{r}_0^{k-1}, \mathbf{a}_{k-C}^k, \mu_{k-C}) p(\mathbf{r}_0^{k-1}|\mathbf{a}_{k-C}^{k-1}, \mu_{k-C}) \\
 &\quad \cdot P\{\mathbf{a}_{k-C}^{k-1}, \mu_{k-C}\} P\{a_k\}. \quad (14)
 \end{aligned}$$

The finite-memory condition (10) is an extension of the *folding condition* introduced in [29], which accounts for the (possibly recursive) encoder/modulator state  $\mu_k$ . The considered model includes any definition of state  $\mu_k$  in terms of a suitable state variable, not necessarily defined in terms of input variables. This leads naturally to the introduction of an *augmented* trellis diagram with state

$$S_k = (\mathbf{a}_{k-C}^{k-1}, \mu_{k-C}) = (a_{k-1}, \dots, a_{k-C}, \mu_{k-C}) \quad (15)$$

<sup>4</sup>A formal proof can be found in [28], but intuition may be sufficient here.

and transition

$$(a_k, S_k) = (\mathbf{a}_{k-C}^k, \mu_{k-C}) = (a_k, \dots, a_{k-C}, \mu_{k-C}). \quad (16)$$

The state evolution over the augmented trellis diagram can be characterized by a next-state function  $\text{NS}(\cdot, \cdot)$  such that  $S_{k+1} = \text{NS}(a_k, S_k) = (a_k, \dots, a_{k-C+1}, \mu_{k-C+1})$ , where  $\mu_{k-C+1} = \text{ns}(a_{k-C}, \mu_{k-C})$ . Using these definitions of augmented state  $S_k$  and transition  $(a_k, S_k)$  and further defining

$$\begin{aligned}
 \beta_{k+1}(S_{k+1}) &\triangleq p(\mathbf{r}_{k+1}^{K-1}|\mathbf{r}_0^k, \mathbf{a}_{k-C+1}^k, \mu_{k-C+1}) \\
 \gamma_k(a_k, S_k) &\triangleq p(r_k|\mathbf{r}_0^{k-1}, \mathbf{a}_{k-C}^k, \mu_{k-C}) P\{a_k\} \\
 \alpha_k(S_k) &\triangleq p(\mathbf{r}_0^{k-1}|\mathbf{a}_{k-C}^{k-1}, \mu_{k-C}) \\
 &\quad \times P\{\mathbf{a}_{k-C}^{k-1}, \mu_{k-C}\} \quad (17)
 \end{aligned}$$

the symbol APP in (14) can be finally expressed as

$$P\{a_k|\mathbf{r}\} \propto \sum_{S_k} \beta_{k+1}[\text{NS}(a_k, S_k)] \gamma_k(a_k, S_k) \alpha_k(S_k). \quad (18)$$

Based on the causality and finite-memory conditions, the quantities  $\alpha_k(S_k)$  and  $\beta_{k+1}(S_{k+1})$  can be computed by means of forward and backward recursions, respectively, according to

$$\alpha_{k+1}(S_{k+1}) = \sum_{(a_k, S_k): S_{k+1}} \alpha_k(S_k) \gamma_k(a_k, S_k) \quad (19)$$

$$\beta_k(S_k) = \sum_{a_k} \beta_{k+1}[\text{NS}(a_k, S_k)] \gamma_k(a_k, S_k) \quad (20)$$

where the notation  $(a_k, S_k) : S_{k+1}$  indicates all transitions  $(a_k, S_k)$  compatible with state  $S_{k+1}$ , i.e., such that  $\text{NS}[(a_k, S_k)] = S_{k+1}$ . The detailed derivation of these recursions can be found in [28]. Here, it suffices to say that all is needed to run the algorithm according to (18), (19), and (20) is the quantity  $\gamma_k(a_k, S_k)$  in (17) and proper boundary conditions  $\{\alpha_0(S_0)\}$  and  $\{\beta_{K-1}(S_{K-1})\}$ .

The algorithm can be also formulated in the logarithmic domain by letting

$$\begin{aligned}
 \bar{\gamma}_k(a_k, S_k) &\triangleq \log \gamma_k(a_k, S_k) \\
 \bar{\alpha}_k(S_k) &\triangleq \log \alpha_k(S_k) \\
 \bar{\beta}_k(S_k) &\triangleq \log \beta_k(S_k) \quad (21)
 \end{aligned}$$

interpretable as *branch*, *forward state*, and *backward state* metrics, respectively. Accordingly, we obtain

$$\begin{aligned} \log P\{a_k|\mathbf{r}\} &\propto \max_{S_k}^* \{ \bar{\alpha}_k(S_k) + \bar{\gamma}_k(a_k, S_k) \\ &\quad + \bar{\beta}_{k+1}[\text{NS}(a_k, S_k)] \} \\ \bar{\alpha}_{k+1}(S_{k+1}) &= \max_{(a_k, S_k): S_{k+1}}^* \{ \bar{\alpha}_k(S_k) + \bar{\gamma}_k((a_k, S_k)) \} \\ \bar{\beta}_k(S_k) &= \max_{a_k}^* \{ \bar{\beta}_{k+1}(S_{k+1}) + \bar{\gamma}_k((a_k, S_k)) \}. \end{aligned}$$

The FB algorithm above is the same algorithm processing soft-decision information in two different formats (i.e., probabilistic soft decisions or the log-probability soft decisions) as discussed in the more general framework of message-passing algorithms.

In the above context, a close examination of the recursive equations reveals a factorization of the overall pmf  $P\{\mathbf{a}_{k-C}^k, \mu_{k-C}|\mathbf{r}\}$  implying an acyclic graph of local constraints corresponding to one time index of the FSM and the FB algorithm is the result of running standard message-passing rules. While the FB algorithm is optimal in this context, using the FB algorithm as a component of an iterative detector is equivalent to an *iterative* message-passing algorithm running on a graphical model that has cycles. In the absence of cycles, the algorithm converges to a globally optimal solution, whereas with cycles in the graphical model, iterative message-passing is only a heuristic.

We remark that the derived formulation of the FB algorithm for detection over channels with memory is based on very general causality and finite-memory statistical conditions, and it does not make any assumption on the specific nature of the channel. A general class of channels which satisfy the finite-memory property (10) are those exhibiting known intersymbol interference (ISI) of finite length, possibly encompassing nonlinear finite-memory effects, and additive white noise. In this case, (10) simplifies by dropping the conditioning observations to

$$p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_0^k) = p(r_k | \mathbf{a}_{k-C}^k, \mu_{k-C}). \quad (22)$$

Unfortunately, the general channel model (3) encompassing a *stochastic parameter*  $\Theta$  is not finite-memory, in the sense that a finite value of the parameter  $C$  does not exist. In these cases, two main approaches can be devised to *approximate* the soft-output MAP detector.

The first approach is based on the following parameter-conditional finite-memory property

$$p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_0^k, \Theta) = p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_{k-C}^k, \mu_{k-C}, \Theta) \quad (23)$$

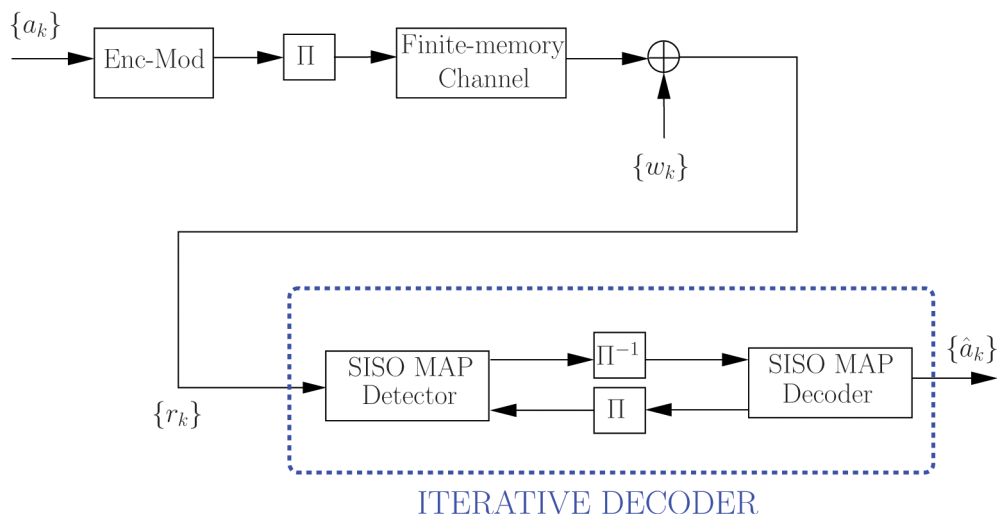
where  $C$  and  $\mu_{k-C}$  are defined as in (10). In this case, a classical estimation-detection decomposition may be used by simply assuming the parameter  $\Theta$  is known for detection and simultaneously estimating the parameter value. This approach requires an *explicit* estimation of the parameter and leads to *adaptive iterative detection* discussed in Section IV.

The second approach is based on the idea that, in any logical stochastic system model, signal samples taken at far apart time instants tend to be independent, for a given information sequence. As a consequence, the finite-memory property (10) can be imposed as an approximation to enable *finite-memory iterative detection*, as discussed in Section V. In this case, there is no attempt to explicitly estimate the parameter, although some specific solutions can be interpreted in terms of an *implicit* estimation procedure.

This FB algorithm can be directly employed in iterative detection schemes, where soft information is circulated among several detection and decoding units [30]. To this purpose, soft information can be input to the algorithm in terms of *a priori* probability of the information symbols, i.e.,  $P\{a_k\}$ . Likewise the soft-output information can be obtained by the computed APP, by extracting the so-called *extrinsic information* component, given by  $\log P\{a_k|\mathbf{r}\} - \log P\{a_k\}$ . A typical example of iterative detection scheme is shown in Fig. 2, where a coded modulation signal is transmitted, after interleaving, over a *finite-memory* channel (e.g., static intersymbol interference channel). At the receiver side, the *iterative decoder* is composed by two blocks, which exchange soft information: the *inner* block is a soft-input soft-output (SISO) MAP detector where a detection algorithm of the family proposed in this section can be used; the *outer* block is a SISO MAP decoder, where the standard FB algorithm can be used. We underline that the example shown in Fig. 2 is just one of the many possible iterative detection/decoding scenarios where finite-memory soft-output detection algorithms can be applied. Finally, we point out that although the FB algorithm (as well as other algorithms that will be presented later) is *recursive* in nature (i.e., it requires a forward and a backward recursion through the received observation), it is activated in an *iterative* manner when it is part of a larger system, as shown in Fig. 2.

#### IV. ADAPTIVE ITERATIVE DETECTION

Adaptive iterative detection (AID) is the generalization of iterative detection to the case where relevant parameters are unknown and the estimation of these parameters is included into the message-passing algorithm (e.g., [27], [31]–[36]). This means that the estimates of the parameters computed explicitly, or in some cases implicitly, will change as the iterative algorithm is run. As a concrete example, consider the model of (5) where the channel symbols  $\{c_k\}$  have been coded by a powerful modern



**Fig. 2.** Communication system with iterative detection: the inner detector is associated with the finite-memory channel, whereas the outer decoder is associated with the used code.

code. In a non-AID approach, the channel gains  $\{f_k\}$  would be estimated once using the observation sequence  $\{r_k\}$ , driven by a signal with embedded, known (training) symbols in  $\{c_k\}$ . These estimates would then be used to compute channel likelihoods which would be used to perform iterative decoding of the code. This approach can perform poorly because, while the coded system may be capable of operating reliably at a very low SNR, reliable estimation of the channel may require much higher SNR or require a very large overhead for training symbols.

In an AID-based system, these channel estimates can be used to start the iterative processing, but they are updated as soft-decision information on  $\{c_k\}$  becomes available. For example, after each iteration of the iterative decoder of the code, updated soft decisions are available for these channel symbols. Reestimation of the channel can be done that takes this new information into account, thus changing the estimates of the channel. These updated channel estimates can be used to recompute the channel symbol likelihoods and proceed with further iterations of the decoder.

The primary value of AID approaches is that structure added into the transmitted signal can be used to aid with channel parameter estimation. For example, a decision directed estimation scheme without AID cannot benefit from structure added by a code, but the AID approach allows the coding gain to be exploited in a decision directed parameter estimator that reestimates the channel parameters as part of the iterative processing.

Many variations on this theme have been explored and most are based on truncation of the estimator memory or by some form of decision feedback on a recursive estimator. The form of the associated estimator is dependent

on the statistical model (or lack thereof) assumed for the parameter. Assumption of Gauss–Markov models or deterministic models tends to lead to recursive estimators while the assumption of more general statistical models leads to nonrecursive estimators. Modeling choices for the overall system can also impact the resulting AID algorithm. For example, one often has the choice of combining the model of the channel parameter and a part of the transmitter structure or separately modeling the channel. In the latter case, especially when the portion of the transmitter modeled with the channel is an FSM, it is natural to seek algorithms that are generalizations of the FB algorithm that include the ability to reestimate the channel parameters (see Section IV-A). For the former case, it is reasonable to proceed roughly as described in the above example (see Section IV-B). Once more we emphasize that the term *adaptive* refers to the overall receiver architecture, while the specific algorithm within a given block can be forward/backward *recursive*.

#### A. Trellis-Based Adaptive Iterative Detection

The key idea underpinning the FB algorithm is that conditioning on the value of the FSM state at a given time decouples the past and the future and allows one to solve the two associated problems separately. This leads to the FB algorithm which is based on dynamic programming [37]. When the value of a parameter relevant to the FSM model is unknown, typically, no such decoupling occurs and in theory finding the best sequences requires searching over a number of sequences that grows exponentially in the length of the data block. However, it is possible to use heuristics to approximate this search while still maintaining the basic structure of the FB algorithm which is the focus of the rest of this section.

1) *Forward-Adaptive Algorithms*: Consider a concrete example where a convolutional code with binary inputs  $a_k$ , state  $\mu_k$ , and output symbol  $c_k$  is the input to the flat fading channel model in (5). If the channel values  $\{f_k\}$  were known, the FB algorithm could be used to perform optimal soft-decision message updates. When these channel values are not known completely, the FB algorithm is not directly applicable and heuristics need to be applied. Assume that the channel is modeled as a first order, autoregressive Gauss–Markov process

$$f_k = \rho f_{k-1} + v_k \quad (24)$$

where  $\{v_k\}$  is a discrete-time complex white Gaussian process with variance  $(1 - \rho^2)$  and independent components. Assuming that the initial state of the code is known, a given sequence is defined by specifying the input sequence  $\mathbf{a}_0^{K-1}$  and computing the APP of the sequence, i.e.,  $P\{\mathbf{a}_0^{K-1} | \mathbf{r}_0^{K-1}\}$ , is equivalent to computing the joint probability distribution function<sup>5</sup>  $P(\mathbf{a}_0^{K-1}, \mathbf{r}_0^{K-1})$ . For the model assumed, this can be performed recursively using

$$P(\mathbf{a}_0^{K-1}, \mathbf{r}_0^{K-1}) \propto \prod_{k=0}^{K-1} p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_0^{k-1}) P\{a_k\} \quad (25)$$

which follows directly using the rules of conditional probability [38] and the assumption that the input bits are independent, with marginal distribution  $P\{a_k\}$  (different for different values of  $k$  in general).

For the statistical models of (5) and (24) the conditional density  $p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_0^{k-1})$  is a complex circular Gaussian with mean and variance that can be computed recursively using a Kalman Filter (KF). Specifically, the mean is  $c_k(\mathbf{a}_0^k) \hat{f}_{k|k-1}$ , where  $\hat{f}_{k|k-1} = E\{f_k | \mathbf{r}_0^{k-1}, \mathbf{a}_0^{k-1}\}$ , and the variance is  $\sigma^2 + |c_k(\mathbf{a}_0^k)|^2 F_{k|k-1}$ , where

$$F_{k+1|k} = |\rho|^2 [1 - g_k c_k(\mathbf{a}_0^k)] F_{k|k-1} + \sigma_v^2 \quad (26)$$

$$g_k = \frac{F_{k|k-1} c_k^*(\mathbf{a}_0^k)}{\sigma^2 + |c_k(\mathbf{a}_0^k)|^2 F_{k|k-1}} \quad (27)$$

comprise the standard KF recursions. Note that the conditional mean and variance depend on the hypothesized path (although this dependency on the Kalman recursions shown above is suppressed for brevity).

The above result shows that for each possible sequence, the sequence APP can be computed recursively using a KF.

<sup>5</sup>We use the term probability distribution function to denote a continuous probability density function (pdf) with some discrete probability masses. For a probability distribution function, we still use the symbol  $P(\cdot)$ .

If one were to consider each possible path, a tree with  $2^k$  leaves would be constructed. When  $k = K - 1$  is reached, the APP of each sequence has been computed and the soft output values can be obtained by marginalizing these sequence probabilities over all sequences consistent with a conditional value of a variable. This optimal processing is referred to as forward recursive estimator-correlator [27]. Similar to the (non-recursive) estimator-correlator [39], [40], optimal processing is impractical because the complexity is exponential in  $K$ . However, the recursive structure allows one to apply decision feedback principles to obtain simple, effective approximations.

Specifically, if one applies a Viterbi-based algorithm to the tree-search problem, we obtain a greedy approximation that performs the same recursions as the Viterbi state recursions, but substitutes the KF-based sequence probability recursion for the standard path probabilities used in the known channel case. In the known channel case, the recursion on sequence probabilities requires conditioning only on  $\mu_k$ , the state of the FSM. However, for the unknown channel case considered above, one must condition on the entire past  $\mathbf{a}_0^k$ . To avoid growing the exponential tree, one can use decision feedback to simplify the search. For example, consider a trellis defined by a state  $S_k$  that contains the information in  $\mu_k$  plus possibly more information about the past,  $S_k = (a_{k-1}, \dots, a_{k-C}, \mu_{k-C})$  for some fixed  $C$ —this definition of an *augmented* state will be crucial for the finite-memory approach described in Section III-B. The per-path KF recursions can then be executed with a condition on  $S_k, S_{k+1}, a_k$ , and a sequence of previous decisions on  $\{a_i\}$  conditioned on the path terminating in  $S_k$ . These decisions are exactly the survivor sequences in the state recursion update. For this reason, such processing is known as per-survivor processing (PSP) [41]. Thus PSP is a method of computing the state probability recursion using decision feedback to forcefully fold the tree search to a trellis with state  $S_k$ .

If only hard decisions are desired, PSP can use survivor traceback as in the Viterbi algorithm. To obtain the desired soft decisions, a backward state recursion is required. The most direct method to achieve this is to store the state transition probabilities (i.e., the conditional means and variance of the conditional Gaussian densities) and then process the backward recursion of the FB algorithm using these transition probabilities. This is a forward-adaptive FB algorithm since channel estimation (adaptation) is performed only during the forward recursion using a causal estimator (e.g., a KF).

At this point it is important to note that any algorithm that approximates the Viterbi algorithm for unknown channels by performing a forward state recursion can be modified to provide soft decisions by the above method—i.e., by storing the transition probabilities during the forward recursion. Many such “hard-out” algorithms can be found in literature (e.g., [38], [41]–[45]). Also, since the state used for the trellis processing is arbitrary, one



can trade off trellis complexity against better approximations of the optimal tree-search processing. This tradeoff has also been noted and studied for these various algorithms. In particular, even if the underlying state has cardinality one (i.e., the system is memoryless when all parameters are known), the above trellis-based processing can be applied since, in theory, a tree-search computation is required due to dependency of the estimator on the entire hypothesized path.

2) *Forward-Adaptive, Backward-Adaptive Algorithms*: The forward-adaptive algorithm described above was based on recursive, causal estimation (i.e., filtering) of the unknown channel parameter. Since one eventually expects to process the data in a noncausal manner (during the backward state recursion), it is reasonable to consider noncausal estimation (i.e., smoothing) of the channel. Using the same model as in the previous development, it is possible to show that the sequence probability can be factored in a noncausal manner [27], [32]

$$P(\mathbf{a}_0^{K-1}, \mathbf{r}_0^{K-1}) = \underbrace{P(\mathbf{a}_0^k, \mathbf{r}_0^k)}_{\text{past/present}} \underbrace{P(\mathbf{a}_{k+1}^{K-1}, \mathbf{r}_{k+1}^{K-1} | \mu_{k+1})}_{\text{future}} \\ \times \underbrace{\int \frac{p(f_k | \mathbf{a}_0^k, \mathbf{r}_0^k) p(f_k | S_{k+1}, \mathbf{a}_{k+1}^{K-1}, \mathbf{r}_{k+1}^{K-1})}{p(f_k)} df_k}_{\text{binding term}}. \quad (28)$$

The relation in (28) indicates that  $P(\mathbf{a}_0^{K-1}, \mathbf{r}_0^{K-1})$  can be split into three factors, of which the first two depend on the past/present and future, respectively, while the third can be viewed as a weighting factor that binds them together. The past-present term can be computed via the per-path KF as described above and the future term can be computed via a similar backward Kalman recursion [27], [32]. The binding term completes a per-path Kalman smoother and may be viewed as an adjustment to the sequence probability that decreases the probability if the forward channel estimate and backward channel estimates disagree significantly.

This per-path Kalman smoother can also be viewed as implying a tree-search. In this case, there is a tree grown in the forward direction to time  $k$ , with  $2^{k+1}$  leaves and another grown in the backward direction, with  $2^{K-k-1}$  leaves. The APP of any sequence can be computed by binding these together according to (28). This is referred to as the forward-backward recursive estimator-correlator [27] and, again, it is not a practical solution due to the complexity. However, decision feedback can be applied in both the forward and backward trees, resulting in an algorithm that performs an adaptive forward state recursion (as in the previous example) and an adaptive backward state recursion. Such algorithms are called forward-adaptive backward-adaptive FB algorithms. Note

that these algorithms store portions of both forward and backward survivor sequences and perform PSP in both the forward and backward recursions.

Generalization of other previously developed hard-out algorithms to forward-adaptive backward-adaptive soft-out algorithms is less straightforward than in the forward-adaptive case because these existing algorithms typically perform only filtering. This has been considered for the model of (5) when the channel is Gaussian, but with arbitrary correlation [46]. Also, one can specify approximate smoothers based on causal and anti-causal estimators (e.g., see [35] for this applied to the phase tracking problem).

Forward-adaptive backward-adaptive algorithms typically perform better than forward-adaptive algorithms in the presence of high channel dynamics. However, the latter offer some practical advantages. For example, in a packet-based format, a forward-adaptive backward-adaptive algorithm is best suited for a packet with training at both ends, while a forward-adaptive algorithm needs training at only one end of the packet. Also, forward-adaptive algorithms are more compatible with hardware architectures typically used to implement forward-backward recursions in hardware [47]–[51].

## B. Non-Trellis-Based Adaptive Iterative Detection

In some cases, it is necessary to resort to a different, non-trellis-based, approach. This occurs, for example, when an LDPC code is employed, but the same approach can be also used in the case of turbo or serially concatenated codes. In this section, we will assume that the unknown parameters  $\Theta$  are modeled as random variables with known statistics. Hence, we adopt a Bayesian approach to detection. A different non-Bayesian approach is employed in [52]. To illustrate a possible approach to adaptive iterative detection, we will employ FGs and the SPA.

For illustration purposes, we consider a channel model for which  $x_k(\mathbf{a}, \Theta) = x_k(c_k, \theta_k)$ , that is the noiseless observation at discrete time  $k$  only depends on the  $k$ th code symbol and a channel parameter. This is, for example, the case of the channel models (4) and (5). The joint probability distribution function of symbols and channel parameters can be expressed as

$$P(\mathbf{a}, \mathbf{c}, \Theta | \mathbf{r}) \propto P\{\mathbf{a}\} P\{c|\mathbf{a}\} p(\Theta) p(\mathbf{r} | \Theta, \mathbf{c}) \\ \propto \chi[c = \eta_C(\mathbf{a})] p(\Theta) p(\mathbf{r} | \Theta, \mathbf{c}) \\ \propto \chi[c = \eta_C(\mathbf{a})] p(\Theta) \prod_{k=0}^{K-1} p(r_k | c_k, \theta_k) \\ \propto \chi[c = \eta_C(\mathbf{a})] p(\Theta) \prod_{k=0}^{K-1} g_k(c_k, \theta_k) \quad (29)$$

having defined by  $\eta_C$  the encoding function associating the information sequence  $\mathbf{a}$  to the codeword  $\mathbf{c}$ , by  $\chi[c = \eta_C(\mathbf{a})]$

the code indicator function, equal to 1 if  $\mathbf{c}$  is the codeword corresponding to  $\mathbf{a}$  and to zero otherwise, and

$$g_k(c_k, \theta_k) \triangleq \exp \left\{ \frac{1}{\sigma^2} \Re[r_k x_k^*(c_k, \theta_k)] - \frac{|x_k(c_k, \theta_k)|^2}{2\sigma^2} \right\} \\ \propto \exp \left\{ -\frac{1}{2\sigma^2} |r_k - x_k(c_k, \theta_k)|^2 \right\} \quad (30)$$

and having assumed that the information symbols are also uniformly distributed (hence  $P\{\mathbf{a}\}$  is constant).

Let us now suppose that the channel parameters satisfy the following Markov property:

$$p(\theta_k | \theta_{k-1}, \theta_{k-2}, \dots, \theta_0) = p(\theta_k | \theta_{k-1}). \quad (31)$$

as for a Wiener process or an autoregressive model of first order (AR(1)) [53].<sup>6</sup> In this case, using the chain rule, we can further factor

$$p(\Theta) = p(\theta_0) \prod_{k=1}^{K-1} p(\theta_k | \theta_{k-1}). \quad (32)$$

Hence

$$P(\mathbf{a}, \mathbf{c}, \Theta | \mathbf{r}) \propto \chi[\mathbf{c} = \eta_{\mathbf{c}}(\mathbf{a})] p(\theta_0) g_0(c_0, \theta_0) \\ \cdot \prod_{k=1}^{K-1} g_k(c_k, \theta_k) p(\theta_k | \theta_{k-1}) \quad (33)$$

whose corresponding FG is shown in Fig. 3.

The application of the SPA to this FG allows the exact (in the absence of cycles in the graph) or approximate (if cycles are present) computation of the marginal  $a$  posteriori probabilities  $P\{a_k | \mathbf{r}\}$  [12], thus implementing MAP symbol detection. We used extensively the proportionality relationship, since the SPA is defined up to scaling its messages by positive factors, independent of the variables represented in the graph. Since the channel parameters, which are continuous random variables, are explicitly represented in the graph, the application of the SPA becomes impractical since it involves integral computations. To solve this problem, the method of *canonical distributions*, suggested in [54], can be adopted. This consists of constraining the messages from/to the continuous variables to a prescribed family of pdf's, that admits a compact parametric representation. Hence, the message computation reduces to the computation of the

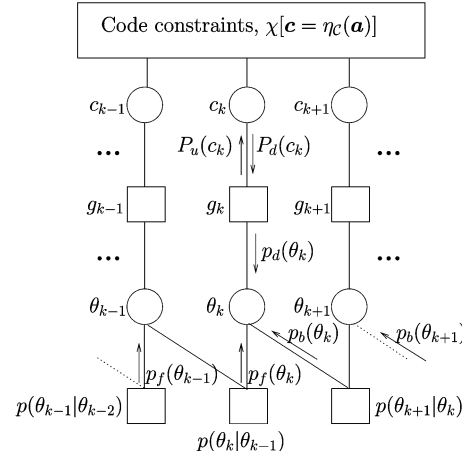


Fig. 3. Factor graph corresponding to equation (33).

pdf parameters. This representation can be exact or, more often, may involve some approximations.

In the following examples, for specific channels, we detail some detection algorithms which can be derived from this approach. We concentrate on the message computation and exchange in the lower part of the graph since the SPA applied to the FG in the upper box, corresponding to the code constraints, consists of the decoding algorithm whose efficient implementation depends on the structure of the code and needs no details here. We stress that the messages the decoder exchanges with the detector are represented by an estimate of the code symbol  $a$  posteriori probabilities only. For this reason, we refer to this scheme as *non-trellis-based* or *separate adaptive iterative detection*, since the detector operates without taking the code constraints into consideration. Omitting, for the sake of notational simplicity, the explicit reference to the current iteration, we will denote by  $P_d(c_k)$  the message from variable node  $c_k$  to factor node  $g_k$ , and by  $P_u(c_k)$  the message in the opposite direction (see Fig. 3).

1) *Example—Channel Affected by a Phase Noise Modeled as a Wiener Process*: Let us consider the channel model (4) and assume that the channel phase  $\phi_k$  satisfy the discrete-time Wiener model described by

$$\phi_k = \phi_{k-1} + \Delta_k \quad (34)$$

where the increments  $\Delta_k$  are i.i.d. Gaussian<sup>7</sup> random variables with mean zero and known standard deviation  $\sigma_\Delta$ .

<sup>6</sup>The generalization to AR models of higher order or autoregressive moving average (ARMA) processes is straightforward.

<sup>7</sup>Note that, since the channel phase is defined modulo  $2\pi$ , the pdf  $p(\phi_k | \phi_{k-1})$  can be approximated as Gaussian only if  $\sigma_\Delta \ll 2\pi$ .

With reference to the messages in Fig. 3, where now  $\theta_k = \phi_k$ , one obtains that the message  $p_d(\phi_k)$  from factor node  $g_k$  to variable node  $\phi_k$  can be expressed as

$$p_d(\phi_k) \propto \sum_{c_k} P_d(c_k) g_k(c_k, \phi_k). \quad (35)$$

We also assume that in the lower part of the FG, a forward-backward node activation schedule is adopted. Therefore, the messages  $p_f(\phi_k)$ , from factor node  $p(\phi_k|\phi_{k-1})$  to variable node  $\phi_k$ , and  $p_b(\phi_k)$ , from factor node  $p(\phi_{k+1}|\phi_k)$  to variable node  $\phi_k$ , can be recursively computed as follows:

$$p_f(\phi_k) \propto \int_0^{2\pi} p_d(\phi_{k-1}) p_f(\phi_{k-1}) p(\phi_k|\phi_{k-1}) d\phi_{k-1} \quad (36)$$

$$p_b(\phi_k) \propto \int_0^{2\pi} p_d(\phi_{k+1}) p_b(\phi_{k+1}) p(\phi_{k+1}|\phi_k) d\phi_{k+1} \quad (37)$$

with uniform pdf's as initial conditions

$$p_f(\phi_0) = p_b(\phi_{N-1}) = \frac{1}{2\pi}. \quad (38)$$

Finally, the message  $P_u(c_k)$  from  $g_k$  to  $c_k$  is given by

$$P_u(c_k) \propto \int_0^{2\pi} p_f(\phi_k) p_b(\phi_k) g_k(c_k, \phi_k) d\phi_k. \quad (39)$$

Different canonical distributions can now be adopted to approximately compute the messages in (36), (37), and (39), leading to algorithms with different performance and complexity. The first one is based on a discretization of channel phase. This case corresponds to letting the canonical distribution be a weighted sum of impulses. This approach has been adopted for Viterbi-like and FB-like receivers in [55] and [56]–[59], respectively. The channel phase  $\phi_k$  is assumed to take on the following  $L$  values:  $\{0, 2\pi/L, \dots, 2\pi(L-1)/L\}$ . In [56], the authors found that for  $M$ -PSK signals,  $L = 8M$  values are sufficient to have no performance loss. Obviously, this approach becomes “optimal” (in the sense that it approaches the performance of the exact SPA) for a sufficiently large number of discretization levels, at the expense of an increasing computational complexity.

The second approach we report is based on a Tikhonov canonical distribution [60]–[62]. Let us consider (35). If the messages  $P_d(c_k)$  were the exact probabilities of the code symbols, it would hold

$$p_d(\phi_k) \propto \sum_{c_k} P_d(c_k) g_k(c_k, \phi_k) \propto p(r_k|\phi_k). \quad (40)$$

We approximate  $p_d(\phi_k)$  by the nearest Gaussian pdf in the sense of divergence (Kullback–Leibler distance) [63]. This yields the Gaussian pdf with mean  $E[r_k|\phi_k]$  and variance  $\text{var}(r_k|\phi_k)$  [60]. Hence, letting  $\alpha_k$  and  $\beta_k$  be the first- and second-order moments of  $c_k$ , given by

$$\alpha_k \triangleq \sum_{c_k} c_k P_d(c_k), \quad \beta_k \triangleq \sum_{c_k} |c_k|^2 P_d(c_k) \quad (41)$$

we obtain

$$\begin{aligned} p_d(\phi_k) &\simeq \frac{1}{\pi(2\sigma^2 + \beta_k - |\alpha_k|^2)} \exp\left\{-\frac{|r_k - \alpha_k e^{j\phi_k}|^2}{2\sigma^2 + \beta_k - |\alpha_k|^2}\right\} \\ &\propto \exp\left\{2 \frac{\text{Re}[r_k \alpha_k^* e^{-j\phi_k}]}{2\sigma^2 + \beta_k - |\alpha_k|^2}\right\}. \end{aligned} \quad (42)$$

Substituting (42) in the forward recursion (36), it follows

$$\begin{aligned} p_f(\phi_k) &\simeq \int_0^{2\pi} \exp\left\{2 \frac{\text{Re}[r_{k-1} \alpha_{k-1}^* e^{-j\phi_{k-1}}]}{2\sigma^2 + \beta_{k-1} - |\alpha_{k-1}|^2}\right\} \\ &\quad \cdot p_f(\phi_{k-1}) p(\phi_k|\phi_{k-1}) d\phi_{k-1}. \end{aligned} \quad (43)$$

When the channel phase is slowly varying, i.e., for  $\sigma_\Delta \rightarrow 0$ , we have  $p(\phi_k|\phi_{k-1}) = \delta(\phi_k - \phi_{k-1})$ . In this case, the solution of the recursion given by (43) with initial condition (38) is a sequence of Tikhonov pdf's, given by

$$p_f(\phi_k) \propto \exp\{\Re[a_{f,k} e^{-j\phi_k}]\} \quad (44)$$

where  $a_{f,k}$  can be recursively computed as

$$a_{f,k} = a_{f,k-1} + 2 \frac{r_{k-1} \alpha_{k-1}^*}{2\sigma^2 + \beta_{k-1} - |\alpha_{k-1}|^2} \quad (45)$$

with the initial condition  $a_{f,0} = 0$ . Similarly, the solution of the backward recursion (37) under the above approximations is the sequence of Tikhonov pdf's

$$p_b(\phi_k) \propto \exp\{\Re[a_{b,k}e^{-j\phi_k}]\} \quad (46)$$

where  $a_{b,k}$  can be recursively computed as

$$a_{b,k} = a_{b,k+1} + 2 \frac{r_{k+1}\alpha_{k+1}^*}{2\sigma^2 + \beta_{k+1} - |\alpha_{k+1}|^2} \quad (47)$$

with the initial condition  $a_{b,K-1} = 0$ . From (44), (46), and (39), we obtain

$$P_u(c_k) \propto \exp\left\{-\frac{|c_k|^2}{2\sigma^2}\right\} I_0\left(\left|a_{f,k} + a_{b,k} + \frac{r_k c_k^*}{\sigma^2}\right|\right). \quad (48)$$

When the phase varies more rapidly, so that the approximation  $p(\phi_k|\phi_{k-1}) \simeq \delta(\phi_k - \phi_{k-1})$  is no longer valid, it is shown in [60] that the distributions  $p_f(\phi_k)$  and  $p_b(\phi_k)$  are still approximately given in the form (44) and (46), where now the coefficients  $a_{f,k}$  and  $a_{b,k}$  are updated by properly modified forward and backward recursions.  $\square$

2) *Example—Flat Fading Channel Following an AR(1) Model*: Let us consider the channel model (5) and assume that the discrete-time fading process satisfy (possibly in an approximate sense) the AR(1) model described by (24). By applying the SPA rules to the lower part of the graph in Fig. 3, where now  $\theta_k = f_k$ , we obtain that message  $p_d(f_k)$  from factor node  $g_k$  to variable node  $f_k$  can be expressed as

$$p_d(f_k) \propto \sum_{c_k} P_d(c_k) g_k(c_k, f_k). \quad (49)$$

Similarly, we may compute the recursions for messages  $p_f(f_k)$  and  $p_b(f_k)$ , and the update of the *a posteriori* probabilities  $P_u(c_k)$ . In the literature, message  $p_d(f_k)$ , which is a Gaussian mixture, is usually approximated as [33], [64]

$$p_d(f_k) = \frac{1}{2\pi\sigma^2} \exp\left\{-\frac{|r_k - f_k \hat{c}_k|^2}{2\sigma^2}\right\} \quad (50)$$

where  $\hat{c}_k = \arg \max_{c_k} P_d(c_k)$  in case of hard decision feedback or  $\hat{c}_k = \alpha_k$ , with  $\alpha_k$  given by (41), in case of soft-

decision feedback. When adopting this approximation, the exact solution of the integral recursions for  $p_f(f_k)$  and  $p_b(f_k)$  is a sequence of Gaussian pdf's. As shown in [12], in this case the application of the SPA gives the well-known Kalman smoother, already proposed for this detection problem in [33] and [64]. This approach can be straightforwardly extended to the case of a fading process modeled as AR processes of higher order or as ARMA processes.

As a conclusion, for general fading channels we also mention the approach based on a sliding-window Wiener filter adopted for example in [31], [33], [65].  $\square$

## V. FINITE-MEMORY ITERATIVE DETECTION

If the channel is affected by *stochastic uncertainty*, i.e., it encompasses a set of random parameters, the observations  $\{r_k\}$  at its output are conditionally *dependent*, so that the channel memory may not be finite. From (3), a general parametric model for the observation  $r_k$  is the following:

$$r_k = g(\mathbf{a}_{k-L}^k, \mu_{k-L}, \boldsymbol{\theta}_0^k) + w_k \quad (51)$$

where  $L$  is an integer,  $\boldsymbol{\theta}_0^k$  is a sequence of stochastic parameters independent of  $\mathbf{a}$ , and  $g(\cdot)$  denotes the functional dependence of the noiseless channel output on the information sequence and the parameter set—note that  $\Theta = \boldsymbol{\theta}_0^{K-1}$  in this case. Under this channel model, the following *conditional Markov property*

$$p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_0^k) = p(r_k | \mathbf{r}_{k-R}^{k-1}, \mathbf{a}_0^k) \quad (52)$$

where  $R$  is the order of Markovianity, is sufficient to guarantee a finite-memory property (10). In fact, (52) can be shown to imply the following [28]:

$$\begin{aligned} p(r_k | \mathbf{r}_0^{k-1}, \mathbf{a}_0^k) &= p(r_k | \mathbf{r}_{k-R}^{k-1}, \mathbf{a}_{k-C}^k, \mu_{k-C}) \\ &= p(r_k | \mathbf{r}_{k-R}^{k-1}, a_k, S_k) \end{aligned} \quad (53)$$

where the finite-memory parameter is  $C = R + L$  and  $S_k$  is defined as in (15).

It is immediate to recognize that (53) represents a special case of (10) and can be directly used to derive the basic branch metric of the FB algorithm defined in (21) and (17). In other words, (53) is the key relation that “links” a generic finite-memory detection algorithm with the specific channel with memory. A statistical description

of the stochastic channel parameter allows one to compute (53) as

$$\begin{aligned} p(r_k | \mathbf{r}_{k-R}^{k-1}, a_k, S_k) &= \frac{p(\mathbf{r}_{k-R}^k | a_k, S_k)}{p(\mathbf{r}_{k-R}^{k-1} | S_k)} \\ &= \frac{\mathbb{E}_{\theta_0^k} \{p(\mathbf{r}_{k-R}^k | a_k, S_k, \theta_0^k)\}}{\mathbb{E}_{\theta_0^{k-1}} \{p(\mathbf{r}_{k-R}^{k-1} | S_k, \theta_0^{k-1})\}}. \end{aligned} \quad (54)$$

Use of the above basic branch metrics leads to the design of a wide range of finite-memory iterative detection (FMID) schemes.

The above exact result, although theoretically limited by the fact that *in realistic scenarios* the conditional Markov property (52) is seldom met *exactly*, suggests a reasonable approach to devise effective approximate detection algorithms whenever the conditional observations are asymptotically independent for increasing index difference [29], [44].

### A. Trellis-Based Finite-Memory Iterative Detection

1) *Example—Channel Affected by an Unknown Phase*: As a first example, we assume that the channel introduces an unknown phase rotation, modeled as a time-invariant random variable  $\phi$  with uniform distribution in  $[0, 2\pi)$ . This is a special case of the generic phase-uncertain channel observation model (4) with  $\phi_k = \phi$  and also a special case of model (51) with  $L = 0$  ( $C = R$ ) and a dependence from a single time-invariant stochastic parameter. It is immediate to conclude that, being  $\phi$  a random variable, the channel memory is infinite. Hence, the conditional Markov property can be claimed in an approximate sense only. On the basis of the considered phase model,  $p(r_k | \mathbf{r}_{k-C}^{k-1}, a_k, S_k)$  can be expressed using (54) as

$$\begin{aligned} p(r_k | \mathbf{r}_{k-C}^{k-1}, a_k, S_k) &= p(r_k | \mathbf{r}_{k-C}^{k-1}, \mathbf{c}_{k-C}^k) \\ &= \frac{\mathbb{E}_\phi \{p(\mathbf{r}_{k-C}^k | \phi, \mathbf{c}_{k-C}^k)\}}{\mathbb{E}_\phi \{p(\mathbf{r}_{k-C}^{k-1} | \phi, \mathbf{c}_{k-C}^{k-1})\}} \\ &\propto \exp\left\{-\frac{|c_k|^2}{2\sigma^2}\right\} \frac{I_0\left(\frac{1}{\sigma^2} \left| \sum_{i=0}^C r_{k-i} c_{k-i}^* \right| \right)}{I_0\left(\frac{1}{\sigma^2} \left| \sum_{i=1}^C r_{k-i} c_{k-i}^* \right| \right)} \end{aligned} \quad (55)$$

where  $I_0(x)$  is the zeroth-order modified Bessel function of the first kind [66]. This result, obtained here as a special case of (54), is equivalent to previous solutions devised for noncoherent detection [34], [67]–[69]. The same metric can then be used, through a further approximation step, also in scenarios with time-varying channel phase.  $\square$

2) *Example—Flat Fading Channel*: As a second example, we consider transmission over a flat Rayleigh fading channel ( $L = 0$  and  $C = R$ ). The observation model is given by (5), and we assume that the fading process  $\{f_k\}$  has an autocovariance sequence modeled according to isotropic scattering [70], i.e., given by  $\mathbb{E}\{f_k f_{k-n}^*\} = J_0(2\pi Bn)$ , where  $J_0(\cdot)$  is the zeroth-order Bessel function [66] and  $B$  is the normalized Doppler rate. In this case, the conditional Markov property is an approximation as well, and  $p(r_k | \mathbf{r}_{k-C}^{k-1}, a_k, S_k)$  can be computed, considering linear prediction, according to

$$\begin{aligned} p(r_k | \mathbf{r}_{k-C}^{k-1}, a_k, S_k) &= p(r_k | \mathbf{r}_{k-C}^{k-1}, \mathbf{c}_{k-C}^k) \\ &= \frac{1}{2\pi\sigma_k^2} e^{-\frac{1}{\sigma_k^2} \left| r_k - \sum_{i=1}^C r_{k-i} \frac{c_k}{c_{k-i}} p_i \right|^2} \end{aligned} \quad (56)$$

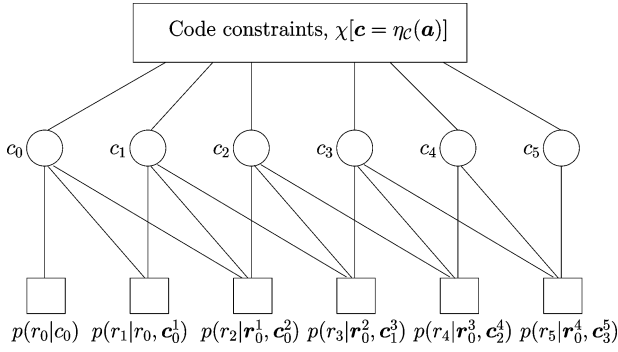
where the order of Markovianity  $C$  can be interpreted as the *prediction order*,  $\{p_i\}_{i=1}^C$  are the prediction coefficients (which depend on state  $S_k$ , but not on symbol  $a_k$ ), and  $\sigma_k^2$  represents the mean square prediction error at epoch  $k$ . The result in (56), which can be derived from (53) owing to the Gaussianity of the observable, was obtained in [42], [43], and [71]–[75] as a solution for maximum likelihood sequence detection over fading channels.  $\square$

### B. Non-Trellis-Based Finite-Memory Iterative Detection

As in Section IV-B, we use the FG/SPA framework. Using the causality and finite-memory conditions (11) and (12), the joint *a posteriori* probability mass function of the information symbols can be expressed as

$$P\{\mathbf{a}, \mathbf{c} | \mathbf{r}\} \propto \chi[\mathbf{c} = \eta\mathbf{c}(\mathbf{a})] \prod_{k=0}^{K-1} p(r_k | \mathbf{r}_0^{k-1}, \mathbf{c}_{k-C}^k) \quad (57)$$

where the causality condition has been applied. The corresponding FG is shown in Fig. 4 for  $C = 2$ , and represents both the code constraints (described by  $\chi(\cdot)$ ) and the channel behavior. With respect to the case of a transmission over a memoryless channel, *additional factor nodes* must be added at the bottom of the graph, as shown in Fig. 4. Due to the presence of cycles in the FG, the application of the SPA allows to derive only an approximation of the marginal *a posteriori* probabilities  $P\{a_k | \mathbf{r}\}$ . In this case of presence of cycles, a major role is also played by the adopted schedule. As an example, the so-called *flooding schedule* [76], suited for a fully parallel implementation of the detector/decoder, can be adopted. Following the SPA rules, the additional factor nodes introduced in the FG of Fig. 4 perform a marginalization, based on the channel model, without taking into account the code



**Fig. 4.** Overall factor graph for  $C = 2$ .

constraints. The complexity of this marginalization is, in general, exponential in  $C$ .

The quality of the convergence of the SPA to the exact marginal probabilities is in general determined by the girth of the graph. As an example, in designing LDPC codes, cycles of length 4 must be avoided to ensure decoding convergence. The FG derived from the proposed factorization has, in general, girth 4 (as one can see in Fig. 4). Surprisingly, we verified by computer simulations that these length-4 cycles involving two factor nodes which model the channel behavior often do not affect the convergence of the algorithm. If this is not the case, as for transmissions over ISI channels, FG transformations can be adopted [77].

The complexity of the proposed algorithms may be reduced following a technique similar to that described in [78] for FB algorithms or to reduced-state sequence detection (RSSD) used for MAP sequence detection [79]–[81]. The details can be found in [82] and [83]. For equal energy signals, a modified version of the described FGs for noncoherent and flat fading channels may be devised. In fact, in this case each additional factor node in Fig. 4 can be slit into  $C$  two-edge factor nodes [82], as will be described in detail in the following example.

1) *Example—Equal Energy Signals Transmitted Over a Flat Fading Channel:* In this case,  $p(r_k | \mathbf{r}_{k-C}^{k-1}, \mathbf{c}_{k-C}^k)$  given by (56) can be further simplified. In fact, the prediction coefficients and the mean square prediction error become independent of the considered sequence. Taking into account that  $|c_k| = 1$  and denoting  $\sigma_k^2 = \sigma_e^2, \forall k$ , after straightforward manipulations we have

$$p(r_k | \mathbf{r}_{k-C}^{k-1}, \mathbf{c}_{k-C}^k) \propto \exp \left\{ \frac{2}{\sigma_e^2} \sum_{i=1}^C \Re [p_i r_k r_{k-i}^* c_k^* c_{k-i}] \right\} \cdot \exp \left\{ -\frac{2}{\sigma_e^2} \sum_{i=1}^C \sum_{\ell=i+1}^C \Re [p_i p_\ell r_{k-i} r_{k-\ell}^* c_{k-i}^* c_{k-\ell}] \right\}. \quad (58)$$

Substituting (58) into (57), it can be easily shown, by grouping the exponential terms which depend on the same argument  $r_{k-i} r_{k-\ell}^* c_{k-i}^* c_{k-\ell}$ , that the resulting joint a posteriori probability of the information symbols becomes

$$P(\mathbf{a}, \mathbf{c} | \mathbf{r}) \propto \chi[\mathbf{c} = \eta_C(\mathbf{a})] \prod_{k=0}^{K-1} \prod_{i=1}^C g_{k-i,k}(c_{k-i}, c_k) \quad (59)$$

having defined

$$g_{k-i,k}(c_{k-i}, c_k) = e^{\frac{2}{\sigma_e^2} \Re [q_i r_k r_{k-i}^* c_k^* c_{k-i}]} \quad (60)$$

where  $q_i = p_i - \sum_{\ell=1}^{C-i} p_\ell p_{\ell+i}$ .

This further factorization has a direct impact on the graph structure. In fact, each factor node can be decomposed into  $C$  simpler two-edge factor nodes [82]. Hence, for increasing values of  $C$ , the number of factor nodes increases linearly but the computational burden at each factor node remains the same. Hence, the algorithm complexity is *linear* in  $C$ . This is a fundamental difference with respect to trellis-based linear predictive receivers [42], [43], [71], [73], [75], whose complexity is *exponential* in the prediction order, and suggests that by using the tool represented by FGs and SPA new computationally efficient algorithms can be derived. In addition, in this modified FG there are no cycles of length 4 in the part of the graph modeling the channel and, by adopting the flooding schedule, the derived algorithm is also well suited for a fully parallel implementation of the detector/decoder.

## VI. OPTIMAL DETECTION WITH POLYNOMIAL COMPLEXITY

We now revisit the problem of adaptive iterative detection and show that when certain assumptions hold true, the exact solution has complexity growing only polynomially with the sequence length  $K$ .

Consider the discretized observation model in (3). Furthermore, assume uncoded transmission, i.e.,

$$\mathbf{x}(\mathbf{a}, \Theta) = (x_0(a_0, \Theta), x_1(a_1, \Theta), \dots, x_{K-1}(a_{K-1}, \Theta)). \quad (61)$$

The assumption of uncoded transmission does not mean that the proposed technique is only useful for detection of uncoded sequences. Rather it reveals a structure similar to the one depicted in Figs. 3 and 4, where the demodulator/estimator is loosely connected to the decoder through the exchange of symbol soft decisions. Thus the demodulator/

estimator operates under the assumption of uncoded symbols with given *a priori* probabilities.<sup>8</sup>

We are interested in obtaining soft decisions on the symbols  $a_k \in \mathcal{A}$  in the presence of  $\Theta$ . Clearly this problem is intimately related to the problem of detecting the sequence  $\mathbf{a}$  in the presence of  $\Theta$ . For this reason we first examine the sequence detection problem and then we make a connection to the symbol soft-decision generation problem.

### A. Sequence Detection

The sequence detection problem under consideration can be formally expressed as

$$\hat{\mathbf{a}} = \arg \max_{\mathbf{a} \in \mathcal{A}^K} \max_{\Theta \in \Lambda} M(\mathbf{r}, \mathbf{a}, \Theta) \quad (62)$$

where  $M(\cdot, \cdot, \cdot)$  is a suitably defined metric. Due to the uncoded assumption, the metric (for a fixed  $\Theta$ ) can be decomposed as

$$M(\mathbf{r}, \mathbf{a}, \Theta) = \sum_{k=0}^{K-1} M_k(r_k, a_k, \Theta) \quad (63)$$

with  $M_k(\cdot, \cdot, \cdot)$  are symbol-wise metrics. Note that the maximization over the unknown parameter  $\Theta$  resembles a generalized likelihood approach to the detection/estimation problem (i.e., maximization of  $P(\mathbf{a}, \Theta | \mathbf{r})$  over both  $\mathbf{a}$  and  $\Theta$ ). It is emphasized however that there are two problems of interest for which generalized likelihood coincides with average likelihood (i.e., maximization of  $P(\mathbf{a} | \mathbf{r})$  over  $\mathbf{a}$ ). The first one is detection of equal-energy constellations in the presence of a uniformly distributed phase rotation (the original idea behind the polynomial complexity solution was first introduced for this model in [85] and reinvented in [86]), and the second one is detection of equal-energy constellations in the presence of complex Gaussian fading (the polynomial complexity solution for this model was first presented in [87]). Observe that although the metric in (63) is decomposed for any fixed  $\Theta$ , this is not the case for the maximized metric over  $\Theta$ . In view of the double maximization nature of the problem, it is natural to define a parameter-conditional sequence detector as follows:

$$\begin{aligned} \hat{\mathbf{a}}(\Theta) &= \arg \max_{\mathbf{a} \in \mathcal{A}^K} M(\mathbf{r}, \mathbf{a}, \Theta) \\ &= (\hat{a}_0(\Theta), \dots, \hat{a}_{K-1}(\Theta)), \end{aligned} \quad (64)$$

<sup>8</sup>A polynomial complexity result has also been established for joint estimation and decoding of simple two-state trellis codes in [84] for the complex Gaussian fading channel.

where the dependence on the observation  $\mathbf{r}$  has been suppressed for notational simplicity. Based on the decomposition in (63), the above parameter-conditional sequence detector can be performed with complexity linear in  $K$ , since the sequence maximization is equivalent to  $K$  symbol-wise maximizations. Similarly, a data-conditional parameter estimator can be defined as

$$\hat{\Theta}(\mathbf{a}) = \arg \max_{\Theta \in \Lambda} M(\mathbf{r}, \mathbf{a}, \Theta) \quad (65)$$

where, again, dependence on  $\mathbf{r}$  is implicitly assumed. It is further assumed that the estimator complexity is polynomial in  $K$ . This is true for both aforementioned problems of interest, i.e., detection of equal-energy constellations in the presence of a uniformly distributed phase rotation, and detection of equal-energy constellations in the presence of complex Gaussian fading, where the complexity is in fact linear in  $K$ . When this is not true, one can still talk about polynomial complexity with the understanding that complexity refers only to the number of sequences that need to be tested before the optimal sequence is found and disregarding the complexity involved in evaluating the metric for a specific sequence  $\mathbf{a}$  through  $\max_{\Theta} M(\mathbf{r}, \mathbf{a}, \Theta) = M(\mathbf{r}, \mathbf{a}, \hat{\Theta}(\mathbf{a}))$ . This approach is followed in the problem of detection of equal-energy constellations in the presence of a phase rotation and frequency jitter [88].

The basic idea behind the polynomial complexity solution is as follows. If it were possible to scan the entire parameter space  $\Lambda$ , and for each  $\Theta \in \Lambda$  obtain the corresponding sequence estimate according to (64), then the global estimate  $\hat{\mathbf{a}}$  would be found. More precisely, if we define the set

$$\mathcal{T} = \{\hat{\mathbf{a}}(\Theta) | \Theta \in \Lambda\} \quad (66)$$

it is easy to see that  $\hat{\mathbf{a}} \in \mathcal{T}$ . Thus the original problem reduces to constructing the set  $\mathcal{T}$ , and then performing the maximization

$$\hat{\mathbf{a}} = \arg \max_{\mathbf{a} \in \mathcal{T}} M(\mathbf{r}, \mathbf{a}, \hat{\Theta}(\mathbf{a})). \quad (67)$$

Of course, the above-described procedure for constructing  $\mathcal{T}$  is not practical since the parameter space is usually infinite—indeed uncountable in most cases of interest. In the following we show that the construction of  $\mathcal{T}$  can be performed with polynomial complexity in  $K$  and furthermore, its size is only polynomial in  $K$ . These two facts are sufficient to prove the original claim of polynomial complexity joint sequence detection and parameter

estimation. Intuitively one expects that points in the parameter space that are close to each other result in the same sequence estimate  $\hat{\mathbf{a}}(\Theta)$ . More precisely, one can partition  $\Lambda$  into sets  $\Lambda_{\mathbf{a}}$  defined as

$$\Lambda_{\mathbf{a}} = \{\Theta \in \Lambda | \hat{\mathbf{a}}(\Theta) = \mathbf{a}\} \quad (68a)$$

$$= \{\Theta \in \Lambda | \forall \mathbf{t} \in \mathcal{A}^K \quad M(\mathbf{r}, \mathbf{a}, \Theta) \geq M(\mathbf{r}, \mathbf{t}, \Theta)\}. \quad (68b)$$

Now it should be clear that in order to construct  $\mathcal{T}$  it is sufficient to “sample” only one point from each set  $\Lambda_{\mathbf{a}}$  in the partition. This reduces the problem of constructing  $\mathcal{T}$  to that of generating the aforementioned partition, or in other words, to determine all nonempty sets  $\Lambda_{\mathbf{a}}$ . Form the definition of  $\Lambda_{\mathbf{a}}$  it is not clear that this task can be accomplished efficiently. However, an efficient decomposition of  $\Lambda_{\mathbf{a}}$  can be obtained (see [87], [89] for a proof of this decomposition), which has the form

$$\Lambda_{\mathbf{a}} = \bigcap_{k=0}^{K-1} \bigcap_{b \in \mathcal{A}} \Lambda_{a_k, b}^{(k)} \quad (69)$$

where

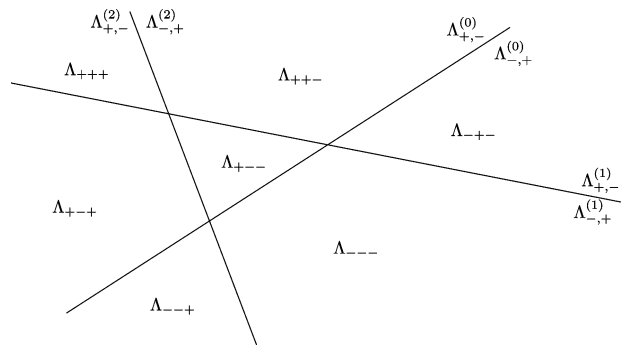
$$\Lambda_{a, b}^{(k)} = \{\Theta \in \Lambda | M_k(r_k, a, \Theta) \geq M_k(r_k, b, \Theta)\}. \quad (70)$$

We have now reduced the problem of constructing  $\mathcal{T}$  into constructing all nonempty sets  $\bigcap_{k=0}^{K-1} \bigcap_{b \in \mathcal{A}} \Lambda_{a_k, b}^{(k)}$ . Based on the definition (70), if we consider the boundary equations of the form

$$M_k(r_k, a, \Theta) = M_k(r_k, b, \Theta) \quad (71)$$

for all  $k \in \{0, \dots, K-1\}$  and for all  $a, b \in \mathcal{A}$  with  $a \neq b$ , the above-described partition will emerge. There are exactly  $Q = K \times M(M-1)/2$  such equations defining an equal amount of boundaries in  $\Lambda$ . This geometric problem can be quite complex. However, for the specific problems discussed above, each of these equations represents a hyperplane in  $\Lambda$ . The problem of generating all possible partitions of the  $d$ -dimensional Euclidean space  $\mathcal{R}^d$  by superimposing  $Q$  hyperplanes has been studied in [90] and has been shown to have complexity  $Q^d$ . Furthermore, the number of convex polytopes generated is on the order of  $Q^d$ . Since  $Q$  depends only linearly in  $K$ , our claim has been established. An example of this partitioning is shown in Fig. 5 for the case of  $\Lambda$  being the two-dimensional Euclidean space.

We conclude this subsection by pointing out that a new suboptimal algorithm emerges from the optimal procedure



**Fig. 5. Partition of  $\Lambda$  for the case of  $K=3$  and  $\mathcal{A} = \{+1, -1\}$ . The three boundaries between the sets  $\Lambda_{\mathbf{a},b}^{(k)}$  are labeled with the appropriate set on each side. Observe that in this example, sequence  $(-1, +1, +1)$  need not be examined. The development in the section essentially guarantees that virtually all but a polynomial number of sequences need be examined without loss of optimality.**

outlined above. In particular, since most of the complexity is associated with the optimal partitioning of  $\Lambda$ , one can skip this step and sample the parameter space in a meaningful and efficient manner so as to “hit” as many sets  $\Lambda_{\mathbf{a}}$  as possible. This procedure involves the selection of  $P$  sample points  $\{\Theta_i\}_{i=1}^P$ , where  $P$  is a design parameter and determines the algorithm complexity. The algorithm then proceeds by finding an approximation of  $\mathcal{T}$  by evaluating the sequences  $\{\mathbf{a}_i = \hat{\mathbf{a}}(\Theta_i)\}_{i=1}^P$  and then maximizing the metric over these sequences.

## B. Symbol Soft-Decision Generation

We now return to the main problem under consideration, that of generating symbol soft decisions for the observation model in (3) under the uncoded assumption of (61). The quantities of interest are of the form

$$S_k(a) = \max_{\mathbf{a}: a_k = a} \max_{\Theta \in \Lambda} M(\mathbf{r}, \mathbf{a}, \Theta). \quad (72)$$

Again we point out that in general the above metric implies soft decision of the form  $\max_{\mathbf{a}: a_k = a} \max_{\Theta} P(\mathbf{a}, \Theta | \mathbf{r})$ , but for the two special problems of interest mentioned above, the generated soft decisions are of the form  $\max_{\mathbf{a}: a_k = a} P(\mathbf{a} | \mathbf{r})$ . Using (63) one can rewrite (72) as follows:

$$S_k(a) = \max_{\Theta \in \Lambda} \left[ M_k(r_k, a, \Theta) + \sum_{i=0, i \neq k}^{K-1} M_i(r_i, \hat{a}_i(\Theta), \Theta) \right]. \quad (73)$$

Following a reasoning similar to the one followed in the previous section for the sequence detection problem, if we



define the set

$$\mathcal{T}_k(a) = \{(\hat{a}_0(\Theta), \dots, \hat{a}_{k-1}(\Theta), a, \hat{a}_{k+1}(\Theta), \dots, \hat{a}_{K-1}(\Theta)) | \Theta \in \Lambda\} \quad (74)$$

then the symbol soft decision in (72) can be found as

$$S_k(a) = \max_{a \in \mathcal{T}_k(a)} M(\mathbf{r}, \mathbf{a}, \hat{\Theta}(\mathbf{a})). \quad (75)$$

Furthermore, there is a straightforward way to obtain  $\mathcal{T}_k(a)$  from the original set  $\mathcal{T}$ ; all that is needed is to change the  $k$ th symbol of each sequence of  $\mathcal{T}$  to  $a$ . Thus, for each candidate sequence in  $\mathcal{T}$ , there are at most  $K \times (M - 1)$  new sequences that need to be tested, which implies that the complexity of evaluating  $S_k(a)$  for all  $k = 0, 1, \dots, K - 1$  and all  $a \in \mathcal{A}$  remains polynomial.

Finally it should be mentioned that a suboptimal algorithm can be derived in a similar way it was derived for the sequence detection problem. First, an approximation of  $\mathcal{T}$  is obtained as outlined in the previous subsection. This approximate set is then enriched by flipping the  $k$ th symbol of all existing sequences, thus obtaining an approximation of all sets of the form  $\mathcal{T}_k(a)$  for  $k = 0, 1, \dots, K - 1$  and all  $a \in \mathcal{A}$ . Soft decisions are then evaluated according to (75) using the approximate  $\mathcal{T}_k(a)$ .

## VII. A FEW ILLUSTRATIVE NUMERICAL RESULTS

We now present a few illustrative results relative to communication systems with iterative detectors using the proposed algorithms. In particular, we distinguish between trellis-based and non-trellis-based schemes.

### A. Trellis-Based Iterative Detection

The performance of the proposed receivers is assessed by means of computer simulations mainly in terms of bit-error rate (BER) versus the SNR  $E_b/N_0$ ,  $E_b$  being the received energy per information bit and  $N_0$  the AWGN single-sided power spectral density.

1) *Phase-Uncertain Communications*: In order to provide the reader with a concise overview of the performance of the proposed detection strategies in a scenario with phase noisy communication channels, we consider a scheme which makes use of a serially concatenated convolutional code (SCCC). We assume that  $N_p = 1$  pilot symbol is inserted every  $N_d$  symbols, and the SNR loss due to the insertion of pilot symbols is accounted for in all the results presented herein. The performance of the considered systems under dynamic channel conditions is investigated. The time-varying phase process  $\{\phi_k\}$  is modeled as a

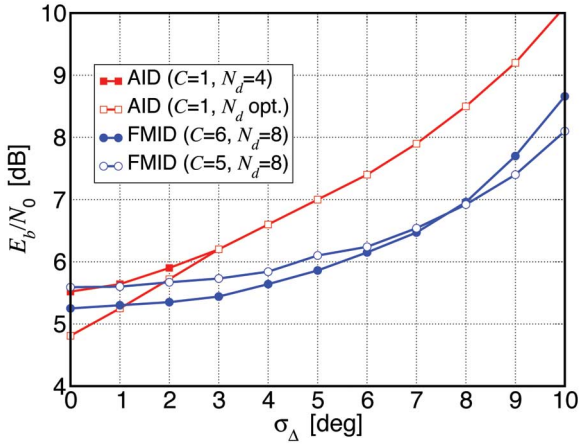
Wiener process, with variance over a signaling interval equal to  $\sigma_\Delta^2$ . Our goal is to compare the AID approach, outlined in Section IV-A, with the FMID approach, outlined in Section V-A. More details on this comparison and the underlying assumptions can be found in [91]. In the FMID case, state reduction techniques (to limit the computational complexity) are applied [78].

The considered SCCC consists of an outer 4-state, rate-2/3 nonrecursive convolutional code and an inner 4-state, rate-3/3 recursive convolutional code, connected through a length-1024 symbol interleaver [92]. The outer encoder is obtained by parallelizing two identical encoders with generator

$$G_o(D) = \left[ 1 \frac{1 + D^2}{1 + D + D^2} \right]$$

and puncturing every other coded bit. The inner code is essentially the anti-rotational invariant version of Code 1 in [93]. The three output bits are mapped to an 8-ary phase shift keying (8-PSK) symbol with natural mapping. The spectral efficiency of the overall code is then 2 bits/s/Hz. At the receiver side, the inner detector/decoder makes use of either an AID-based FB algorithm, whose basic branch metric can be found<sup>9</sup> in [35], or an FMID-based FB algorithm, whose basic branch metric is given by (55). The outer decoder makes use of a coherent FB algorithm. The performance of the two considered detection strategies for this communication system is analyzed by evaluating the SNR necessary to obtain a prescribed BER of  $10^{-3}$ , at ten decoding iterations, as a function of the jitter standard deviation  $\sigma_\Delta$ . The results are presented in Fig. 6. Two curves for each of the two detection strategies are shown. In the AID case one curve corresponds to  $N_d = 4$ , while the other curve is obtained by optimizing the insertion rate for each specific value of the phase noise jitter standard deviation  $\sigma_\Delta$ . In the FMID case, both curves correspond to the optimized value  $N_d = 8$ : one of the curves corresponds to the case with  $C = 6$  (with number of states reduced from  $4^C \times 4 = 4^7$  to  $4^4$ ), while the other curve refers to the case with  $C = 5$  (with number of states reduced from  $4^C \times 4 = 4^6$  to  $4^3$ ). As one can see, the AID detection strategy is better than the FMID detection strategy for low phase jitter standard deviation, while it worsens for increasing standard deviation. On the other hand, the FMID approach is more robust at high phase dynamics, and the required SNR to attain the desired BER is almost constant for  $\sigma_\Delta \leq 5$  degrees. Moreover, the optimized value  $N_d$  in the FMID case does not seem to depend on the phase jitter standard deviation  $\sigma_\Delta$ . For increasing phase jitter the FMID scheme with a short observation

<sup>9</sup>Note that in Section IV-A, only the derivation of an FB algorithm for a scenario with fading is considered. The extension of this approach to a scenario with phase noise is detailed in [35].



**Fig. 6.**  $E_b/N_0$  required to obtain a BER of  $10^{-3}$  at 10 decoding iterations versus the phase jitter standard deviation  $\sigma_\Delta$ . Both the AID and FMID strategies are considered.

window ( $C = 5$ ) outperforms the one with a larger observation window ( $C = 6$ ). The overall conclusion is that there exists a threshold value for  $\sigma_\Delta$  on the order of 1 to 2 degrees, such that AID is better than FMID for  $\sigma_\Delta$  values lower than the threshold, and FMID is better than AID for  $\sigma_\Delta$  values higher than the threshold.

2) *Faded Communications*: In this case, we compare AID and FMID strategies in a scenario with transmission of an SCCC over a Rayleigh flat fading channel with normalized bandwidth  $f_D T = 0.01$ . The code consists of an outer 4-state, rate-1/2 convolutional code connected through a length-1024 pseudorandom interleaver [6] to an inner 4-state, rate-1/2 convolutional code. The respective generator matrices are given by

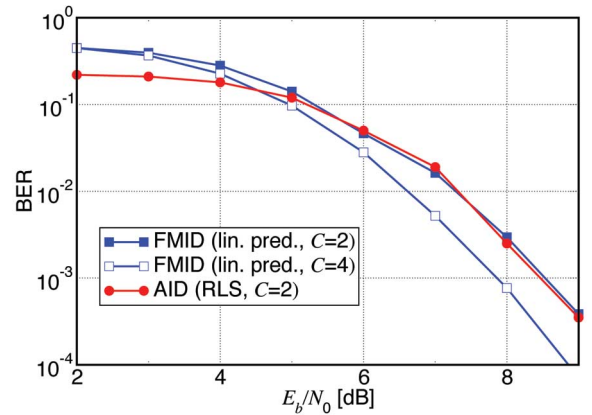
$$\begin{aligned} G_o(D) &= [1+D+D^2 \quad 1+D^2] \\ G_i(D) &= \begin{bmatrix} 1 & \frac{1+D^2}{1+D+D^2} \end{bmatrix}. \end{aligned} \quad (76)$$

The output symbols are mapped to a quaternary PSK (QPSK) constellation with Gray mapping, resulting in an overall code of spectral efficiency 0.5 bit/s/Hz. The outer decoder makes use of a standard BCJR algorithm, whereas the inner detector/decoder makes use of either a linear predictive FB algorithm with branch metric given by (56) (FMID case) or the recursive least square (RLS)-based forward-adaptive/backward adaptive FB algorithm described in Section IV-A (AID case), with RLS forgetting factor optimized by trial and error. In the AID case, one pilot QPSK symbol is introduced every eight data QPSK data symbols and the corresponding power loss (due to pilots) is properly taken into account in the definition of

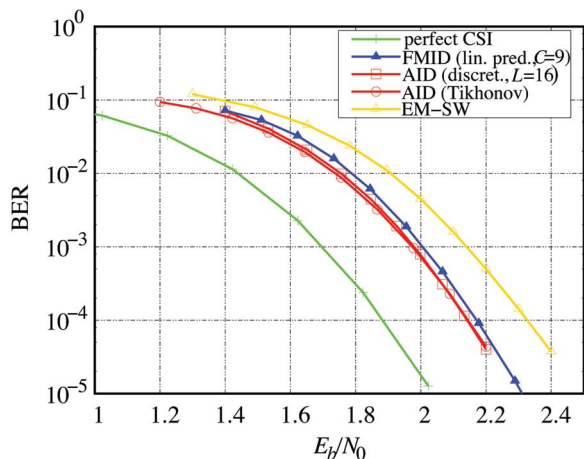
the SNR. Therefore, the comparison between FMID and AID schemes is fair. In Fig. 7, the BER is shown as a function of the SNR. In all cases, five decoding iterations are considered. In the FMID case, two possible values for the finite-memory parameter  $C$  are considered, although the number of states of the inner detector/decoder is reduced to 8 in all cases. In the AID case, the number of states of the inner code (equal to 4) is “expanded” to 8, as in the FMID case—this corresponds to considering  $C = 2$  in the AID case. As one can see, the performance of FMID and AID case, in the case with  $C = 2$ , is the same at sufficiently large values of the SNR, whereas the AID scheme shows an improved performance at low SNR. As expected, increasing the finite-memory parameter improves the performance.

## B. Non-Trellis-Based Iterative Detection

We now consider non-trellis-based iterative detection and the algorithms described in Sections IV-B and V-B. In all simulations, a scheme making use of a rate-1/2 (3, 6)-regular LDPC code of length 4000 [94] with BPSK modulation is considered and a maximum of 200 iterations of the SPA on the overall graph is allowed. In all simulated cases, one pilot symbol every 20 code symbols is inserted in the transmitted codeword in order to make the iterative decoding algorithms bootstrap. Pilot symbols involve a slight decrease of the effective information rate, resulting in an increase in the required signal-to-noise ratio. This increase has been introduced artificially in the curve labeled “perfect CSI” for the sake of comparison. Hence, the gap between the “perfect CSI” curve and the others is uniquely due to the need for phase estimation/compensation, and not to the rate decrease due to pilot symbols.



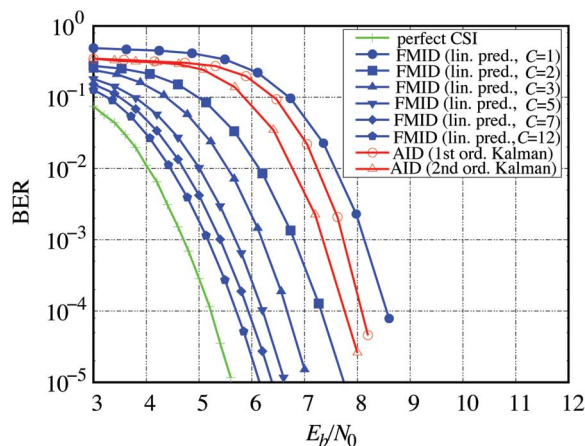
**Fig. 7.** BER of a SCCC, with inner linear predictive combined detection and decoding (FMID strategy), over a Rayleigh flat fading channel with normalized Doppler bandwidth  $f_D T = 0.01$ . In all cases, five decoding iterations are considered.



**Fig. 8.** BER of a LDPC, considering both FMID and AID strategies, over a channel with phase noise ( $\sigma_{\Delta} = 6$  degrees).

1) *Phase-Uncertain Communications:* In Fig. 8, for a channel with phase noise, we compare the performance of the FMID approach with a couple of AID-based strategies. The FMID receiver is based on linear prediction. The two AID-based receivers are based on the canonical distribution approaches described in Section IV-B, Example 2, namely the practically optimal approach based on discretization and that based on the Tikhonov distribution. In this specific application, the AIM approach based on the Tikhonov distribution is the most convenient solution, from both a performance and a complexity point of view. In this figure, the performance of the EM-based algorithm described [52] is also shown for the sake of comparison. In order to adapt this algorithm to a time-varying channel phase, a sliding-window version of the EM algorithm is used where the window size was optimized by computer simulation. The resulting algorithm is denoted by sliding-window EM (EM-SW). We found that the optimal window has width of 60 symbols for the considered phase noise. In this case, the performance loss is due to the fact that this algorithm is designed for a different phase model, i.e., a block-constant phase.

2) *Faded Communications:* Finally, in Fig. 9 we consider a flat correlated Rayleigh fading channel with normalized Doppler rate  $f_D T = 10^{-2}$ . The performance in the case of perfect CSI is also considered for comparison. The performance of the FMID approach based on linear prediction is shown for different values of parameter  $C$ . Obviously, for increasing values of the memory parameter  $C$  there is a performance improvement. A similar behavior is observed for different values of the normalized Doppler rate. In addition, due to the linear complexity of the detection algorithm, it is possible to implement receivers with values of  $C$  higher than those used in [42], [43], [71],



**Fig. 9.** BER of a LDPC, considering both FMID and AID strategies, over a Rayleigh flat fading channel with normalized Doppler bandwidth  $f_D T = 0.01$ .

[73], and [75] for Viterbi- or BCJR-based algorithms, thus closer approaching the performance of the receiver with perfect CSI.

The flat fading channel is a case in which the FMID approach is favorable with respect to AIM. In fact, as mentioned in Section IV-B, the explicit representation of the channel parameters into the FG and a fading model autoregressive of order  $N$  (AR( $N$ )) leads to an algorithm for joint decoding and estimation in which the fading estimate is obtained through a Kalman smoother [33], [64]. In this case, however, the number of parameters to be recursively updated in the forward and backward recursions is  $N + N(N + 1)/2$  and therefore the complexity is *quadratic* in  $N$ . Hence, for a value of  $N \simeq C$ , the FMID approach is less complex. In addition, a fully parallel schedule, such that adopted by the the FMID algorithm, would heavily degrade the performance of the Kalman smoother.

## VIII. CONCLUSION

In this paper, we have considered iterative detection for channels with memory. The focus of this paper has been on algorithmic structures, but a few numerical results have also been presented. We proposed a simple classification for describing iterative detection algorithms, distinguishing between adaptive iterative detection (where explicit estimation of the channel parameters is considered), finite-memory detection (where no explicit estimation of channel parameters is considered), and optimal detection with polynomial complexity (where channel parameters and data are viewed as composing a single observation space). We have tried to show commonalities and differences between the various approaches, taking into account a common starting point, consisting of the

implementation of a MAP detection strategy. The proposed classification is by no means *the* classification for iterative detection. More precisely, the presented overview should stimulate the reader to go ahead and derive *novel* detection algorithms by taking the inspiration from those presented here. ■

## REFERENCES

- [1] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theory*, vol. IT-13, no. 2, pp. 260–269, Apr. 1967.
- [2] G. D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. 61, no. 3, pp. 268–278, Mar. 1973.
- [3] P. L. McAdam, L. R. Welch, and C. L. Weber, "MAP bit decoding of convolutional codes," in *Proc. IEEE Int. Symp. Information Theory*, 1972, p. 91.
- [4] 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. 20, no. 2, pp. 284–287, Mar. 1974.
- [5] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE Int. Conf. Communication (ICC)*, 1993, pp. 1064–1070.
- [6] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.
- [7] R. G. Gallager, "Low density parity check codes," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, 1960.
- [8] R. G. Gallager, "Low density parity check codes," *IEEE Trans. Inf. Theory*, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.
- [9] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.
- [10] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEE Electron. Lett.*, vol. 33, pp. 457–458, Mar. 1997.
- [11] B. J. Frey and D. J. C. MacKay, "A revolution: Belief propagation in graphs with cycles," in *Advances in Neural Information Processing Systems*, M. I. Jordan, M. I. Kearns, and S. A. Solla, Eds. Cambridge, MA: MIT Press, 1998, pp. 470–485.
- [12] 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.
- [13] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 1996.
- [14] N. Wiberg, H.-A. Loeliger, and R. Kötter, "Codes and iterative decoding on general graphs," *Eur. Trans. Telecommun.*, vol. 6, no. 5, pp. 513–525, Sep.–Oct. 1995.
- [15] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc.*, vol. 39, no. 1, pp. 1–38, 1977.
- [16] J. A. Fessler and A. O. Hero, "Space-alternating generalized expectation-maximization algorithm," *IEEE Trans. Signal Process.*, vol. 42, no. 10, pp. 2664–2677, Oct. 1994.
- [17] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 1, pp. 55–67, Jan. 1982.
- [18] J. Anderson, T. Aulin, and C.-E. Sundberg, *Digital Phase Modulation*. New York: Plenum, 1986.
- [19] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. IT-27, no. 5, pp. 533–547, Sep. 1981.
- [20] 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.
- [21] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman, and V. Stemann, "Efficient erasure correcting codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.
- [22] 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.
- [23] H. Meyer, M. Oerder, and A. Polydoros, "On sampling rate, analog prefiltering, and sufficient statistics for digital receivers," *IEEE Trans. Commun.*, vol. 42, no. 12, pp. 3208–3214, Dec. 1994.
- [24] M. K. Simon and M.-S. Alouini, *Digital Communication Over Generalized Fading Channels: A Unified Approach to the Performance Analysis*. New York: Wiley, 2000.
- [25] R. J. McEliece, D. J. C. MacKay, and J. F. Cheng, "Turbo decoding as an instance of Pearl's 'belief propagation' algorithm," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 140–152, Feb. 1998.
- [26] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 325–343, Mar. 2000.
- [27] K. M. Chugg, A. Anastasopoulos, and X. Chen, *Iterative Detection: Adaptivity, Complexity Reduction, and Applications*. Norwell, MA: Kluwer, 2001.
- [28] G. Ferrari, G. Colavolpe, and R. Raheli, "A unified framework for finite-memory detection," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 9, pp. 1697–1706, Sep. 2005.
- [29] K. M. Chugg, "The condition for the applicability of the Viterbi algorithm with implications for fading channel MLSD," *IEEE Trans. Commun.*, vol. 46, no. 9, pp. 1112–1116, Sep. 1998.
- [30] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Soft-Input Soft-Output modules for the construction and distributed iterative decoding of code networks," *Eur. Trans. Telecommun.*, vol. 9, no. 2, pp. 155–172, Mar./Apr. 1998.
- [31] M. C. Valenti and B. D. Woerner, "Refined channel estimation for coherent detection of turbo codes over flat-fading channels," *IEE Electron. Lett.*, vol. 34, no. 17, pp. 1033–1039, Aug. 1998.
- [32] A. Anastasopoulos and K. M. Chugg, "Adaptive soft-input soft-output algorithms for iterative detection with parametric uncertainty," *IEEE Trans. Commun.*, vol. 48, no. 10, pp. 1638–1649, Oct. 2000.
- [33] J. Heo and K. M. Chugg, "Adaptive iterative detection for turbo codes on flat fading channels," in *Proc. Wireless Communications and Networking Conf. (WCNC)*, 2000, vol. 1, pp. 134–139.
- [34] G. Colavolpe, G. Ferrari, and R. Raheli, "Noncoherent iterative (turbo) detection," *IEEE Trans. Commun.*, vol. 48, no. 9, pp. 1488–1498, Sep. 2000.
- [35] A. Anastasopoulos and K. M. Chugg, "Adaptive iterative detection for phase tracking in turbo coded systems," *IEEE Trans. Commun.*, vol. 49, no. 12, pp. 2135–2144, Dec. 2001.
- [36] C. Kominakis and R. D. Wesel, "Pilot-aided joint data and channel estimation in flat correlated fading," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM) (Communications Theory Symp.)*, 1999, pp. 2534–2539.
- [37] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. Cambridge, MA: MIT Press, 2002.
- [38] R. A. Iltis, "A Bayesian maximum-likelihood sequence estimation algorithm for a priori unknown channels and symbol timing," *IEEE J. Sel. Areas Commun.*, vol. 10, pp. 579–588, Apr. 1992.
- [39] T. Kailath, "Correlation detection of signals perturbed by a random channel," *IRE Trans. Inf. Theory*, vol. IT-6, pp. 361–366, Jun. 1960.
- [40] T. Kailath, "A general likelihood-ratio formula for random signals in Gaussian noise," *IEEE Trans. Inf. Theory*, vol. IT-15, no. 3, pp. 350–361, May 1969.
- [41] R. Raheli, A. Polydoros, and C. Tzou, "Per-survivor processing: A general approach to MLSE in uncertain environments," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 354–364, Feb.–Apr. 1995.
- [42] J. Lodge and M. Moher, "Maximum likelihood estimation of CPM signals transmitted over Rayleigh flat fading channels," *IEEE Trans. Commun.*, vol. 38, no. 6, pp. 787–794, Jun. 1990.
- [43] G. M. Vitetta and D. P. Taylor, "Maximum likelihood decoding of uncoded and coded PSK signal sequences transmitted over Rayleigh flat-fading channels," *IEEE Trans. Commun.*, vol. 43, no. 11, pp. 2750–2758, Nov. 1995.
- [44] G. Colavolpe and R. Raheli, "Noncoherent sequence detection," *IEEE Trans. Commun.*, vol. 47, no. 9, pp. 1376–1385, Sep. 1999.
- [45] K. M. Chugg and A. Polydoros, "MLSE for an unknown channel—Part I: Optimality considerations," *IEEE Trans. Commun.*, vol. 44, pp. 836–846, Jul. 1996.
- [46] A. Hansson, K. M. Chugg, and T. Aulin, "On forward-adaptive versus forward/backward-adaptive SISO algorithms for Rayleigh fading channels," *IEEE Commun. Lett.*, vol. 5, no. 12, pp. 477–479, Dec. 2001.
- [47] J. Heo, K. M. Chugg, and A. Anastasopoulos, "A comparison of forward-only and bi-directional fixed-lag adaptive sisos," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2000, pp. 1660–1664.
- [48] A. J. Viterbi, "An intuitive justification and a simplified implementation of the MAP decoder for convolutional codes," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 260–264, Feb. 1998.
- [49] S. Yoon and Y. Bar-Ness, "A parallel MAP algorithm for low latency turbo decoding," *IEEE Commun. Lett.*, vol. 6, no. 7, pp. 288–290, Jul. 2002.
- [50] J. Dielissen and J. Huisken, "State vector reduction for initialization of sliding windows MAP," in *Proc. 2nd Int. Symp. Turbo Codes and Related Topics*, 2000, pp. 387–390.

- [51] A. Abbasfar and K. Yao, "An efficient and practical architecture for high speed turbo decoders," in *Proc. IEEE 58th Vehicular Technology Conf.*, 2003, vol. 1, pp. 337–341.
- [52] C. Herzet, M. Moeneclaey, H. Wymeersch, N. Noels, and M. Luise, "Iterative synchronization and related lower bounds: An overview," *Proc. IEEE*, submitted for publication.
- [53] A. Papoulis, *Probability, Random Variables and Stochastic Processes*. New York: McGraw-Hill, 1991.
- [54] 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.
- [55] O. Macchi and L. Scharf, "A dynamic programming algorithm for phase estimation and data decoding on random phase channels," *IEEE Trans. Inf. Theory*, pp. 581–595, Sep. 1981.
- [56] M. Peleg, S. Shamai (Shitz), and S. Galán, "Iterative decoding for coded noncoherent MPSK communications over phase-noisy AWGN channel," *IEE Proc.—Commun.*, vol. 147, no. 2, pp. 87–95, Apr. 2000.
- [57] J. Dauwels and H.-A. Loeliger, "Joint decoding and phase estimation: An exercise in factor graphs," in *Proc. IEEE Symp. Information Theory (ISIT)*, 2003, p. 231.
- [58] R.-R. Chen, R. Koetter, U. Madhow, and D. Agrawal, "Joint noncoherent demodulation and decoding for the block fading channel: A practical framework for approaching Shannon capacity," *IEEE Trans. Commun.*, vol. 51, no. 10, pp. 1676–1689, Oct. 2003.
- [59] J. Dauwels and H.-A. Loeliger, "Phase estimation by message passing," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2004, vol. 1, pp. 523–527.
- [60] G. Colavolpe, A. Barbieri, and G. Caire, "Algorithms for iterative decoding in the presence of strong phase noise," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 9, pp. 1748–1757, Sep. 2005.
- [61] A. Barbieri, G. Colavolpe, and G. Caire, "Joint iterative detection and decoding in the presence of phase noise and frequency offset," *IEEE Trans. Commun.*, vol. 55, no. 1, pp. 171–179, Jan. 2007.
- [62] A. Barbieri and G. Colavolpe, "Soft-output decoding of rotationally invariant codes over channels with phase noise," *IEEE Trans. Commun.*, to be published.
- [63] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.
- [64] H. Niu, M. Shen, J. A. Ritcey, and H. Liu, "A factor graph approach to iterative channel estimation and LDPC decoding over fading channels," *IEEE Trans. Wireless Commun.*, vol. 4, no. 4, pp. 1345–1350, Jul. 2005.
- [65] H.-J. Su and E. Geraniotis, "Low-complexity joint channel estimation and decoding for pilot symbol-assisted modulation and multiple differential detection systems with correlated rayleigh fading," *IEEE Trans. Commun.*, vol. 50, no. 2, pp. 249–261, Feb. 2002.
- [66] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series and Products*. New York: Academic, 1980.
- [67] I. D. Marsland and P. T. Mathiopoulos, "On the performance of iterative noncoherent detection of coded M-PSK signals," *IEEE Trans. Commun.*, vol. 48, no. 4, pp. 588–596, Apr. 2000.
- [68] G. Ferrari, G. Colavolpe, and R. Raheli, "Noncoherent iterative decoding of spectrally efficient coded modulations," *Ann. Telecommun.*, vol. 56, no. 7–8, pp. 409–421, Jul./Aug. 2001.
- [69] G. Ferrari, G. Colavolpe, and R. Raheli, "Noncoherent iterative decoding of spectrally efficient coded modulations," in *Turbo Codes: Error-Correcting Codes of Widening Application*, M. Jézéquel and R. Pyndiah, Eds. London, U.K.: Hermes Penton Science, 2002, pp. 59–81.
- [70] R. H. Clarke, "A statistical theory of mobile radio reception," *Bell Syst. Tech. J.*, vol. 47, pp. 957–1000, Aug. 1968.
- [71] X. Yu and S. Pasupathy, "Innovations-based MLSE for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 1534–1544, Feb.–Apr. 1995.
- [72] I. D. Marsland and P. T. Mathiopoulos, "Multiple differential detection of parallel concatenated convolutional (turbo) codes in correlated fast Rayleigh fading," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 265–275, Feb. 1998.
- [73] P. Hoehner and J. Lodge, "'Turbo DPSK': Iterative differential PSK demodulation and channel decoding," *IEEE Trans. Commun.*, vol. 47, no. 6, pp. 837–843, Jun. 1999.
- [74] D. Makrakis, P. T. Mathiopoulos, and D. Bouras, "Optimal detection of coded differentially encoded QAM and PSK signals with diversity reception in correlated fast fading channels," *IEEE Trans. Veh. Technol.*, vol. 42, no. 3, pp. 245–258, Aug. 1993.
- [75] D. Makrakis, P. T. Mathiopoulos, and D. Bouras, "Optimal decoding of coded PSK and QAM signals in correlated fast fading channels and AWGN: A combined envelope, multiple differential and coherent detection approach," *IEEE Trans. Commun.*, vol. 42, no. 1, pp. 63–75, Jan. 1994.
- [76] F. Kschischang and B. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 219–230, Feb. 1998.
- [77] G. Colavolpe and G. Germe, "On the application of factor graphs and the sum-product algorithm to ISI channels," *IEEE Trans. Commun.*, vol. 53, no. 5, pp. 818–825, May 2005.
- [78] 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.
- [79] M. V. Eyuboğlu and S. U. Qureshi, "Reduced-state sequence estimation with set partitioning and decision feedback," *IEEE Trans. Commun.*, vol. 36, no. 1, pp. 13–20, Jan. 1988.
- [80] A. Duel-Hallen and C. Heegard, "Delayed decision feedback estimation," *IEEE Trans. Commun.*, vol. 37, no. 5, pp. 428–436, May 1989.
- [81] 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.
- [82] G. Colavolpe, "On LDPC codes over channels with memory," *IEEE Trans. Wireless Commun.*, vol. 5, no. 7, pp. 1757–1766, Jul. 2006.
- [83] G. Ferrari, G. Colavolpe, and R. Raheli, *Detection Algorithms for Wireless Communications*. Chichester, U.K.: Wiley, 2004.
- [84] C.-H. Hsu and A. Anastasopoulos, "Maximum likelihood decoding of trellis codes in fading channels with no receiver CSI is a polynomial-complexity problem," in *Proc. Int. Symp. Information Theory*, 2004, p. 147.
- [85] J. K. M. Mackenthun, "A fast algorithm for multiple-symbol differential detection of MPSK," *IEEE Trans. Commun.*, vol. 33, no. 2/3/4, pp. 1471–1474, Feb./Mar./Apr. 1994.
- [86] W. Sweldens, "Fast block noncoherent decoding," *IEEE Commun. Lett.*, vol. 5, no. 4, pp. 132–134, Apr. 2001.
- [87] I. Motedayen-Aval and A. Krishnamoorthy, and A. Anastasopoulos, "Optimal joint detection/estimation in fading channels with polynomial complexity," *IEEE Trans. Inf. Theory*, vol. 53, no. 1, pp. 209–223, Jan. 2007.
- [88] C.-H. Hsu and A. Anastasopoulos, "Design and analysis of joint data detection and frequency/phase estimation algorithms," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 9, pp. 1707–1717, Sep. 2005.
- [89] I. Motedayen-Aval and A. Anastasopoulos, "Polynomial-complexity noncoherent symbol-by-symbol detection with application to adaptive iterative decoding of turbo-like codes," *IEEE Trans. Commun.*, vol. 51, no. 2, pp. 197–207, Feb. 2003.
- [90] H. Edelsbrunner, J. O'Rourke, and R. Seidel, "Constructing arrangements of lines and hyperplanes with applications," *SIAM J. Comput.*, vol. 15, pp. 341–363, 1986.
- [91] G. Ferrari, A. Anastasopoulos, G. Colavolpe, and R. Raheli, "Adaptive iterative detection for the phase uncertain channel: Limited-tree-search versus truncated-memory detection," *IEEE Trans. Veh. Technol.*, vol. 53, no. 2, pp. 433–442, Mar. 2004.
- [92] D. Divsalar, S. Dolinar, and F. Pollara, "Serial concatenation of trellis coded modulation with rate-1 inner code," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, 2000, pp. 777–782.
- [93] R. Nuriyev and A. Anastasopoulos, "Design and robustness analysis of rotationally invariant SCTCM," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2001, vol. 7, pp. 2226–2230.
- [94] D. J. C. MacKay, *Regular LDPC online database*. [Online]. Available: <http://www.inference.phy.cam.ac.uk/mackay/>

## ABOUT THE AUTHORS

**Achilleas Anastasopoulos** (Member, IEEE) was born in Athens, Greece, in 1971. He received the diploma degree in electrical engineering from the National Technical University of Athens, Greece, in 1993, and the M.S. and Ph.D. degrees in electrical engineering from University of Southern California, Los Angeles, in 1994 and 1999, respectively.



He is currently an Associate Professor in the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor. He is the coauthor of the book *Iterative Detection: Adaptivity, Complexity Reduction, and Applications*, (Reading, MA: Kluwer Academic, 2001). His research interests lie in the general area of communication theory, with emphasis in joint parameter estimation and data detection, adaptive iterative algorithms, and coding for fading channels.

Dr. Anastasopoulos is the recipient of the “Myronis Fellowship” in 1996 from the Graduate School at the University of Southern California, and the NSF CAREER Award in 2004. He served as a technical program committee member for ICC 2003 and Globecom 2004. He currently serves on the editorial board of the IEEE TRANSACTIONS ON COMMUNICATIONS.

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



Since 1997, he has been with the University of Parma, where he is now an Associate Professor of Telecommunications. In 2000, he was also 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 several 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*, published by Wiley.

Dr. Colavolpe received the best paper award for the paper “VLSI Design of a fully parallel decoder for LDPC-like codes,” 13th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2005). He is also the principal investigator of several research projects funded by the European Space Agency (ESA-ESTEC) and important telecommunications companies.

**Keith M. Chugg** (Member, IEEE) received the B.S. degree (high distinction) in engineering from Harvey Mudd College, Claremont, CA, in 1989 and the M.S. and Ph.D. degrees in electrical engineering (EE) from the University of Southern California (USC), Los Angeles, in 1990 and 1995, respectively.



During the 1995–1996 academic year he was an Assistant Professor with the Electrical and Computer Engineering Department, University of Arizona, Tucson. In 1996 he joined the EE Department at USC in 1996, where he is currently an Associate Professor. He is also a cofounder of TrellisWare Technologies Inc., San Diego, CA, where he is Chief Scientist. His research interests are in the general areas of signaling, detection, and estimation for digital communication and data storage systems. He is also interested in architectures for efficient implementation of the resulting algorithms. Along with his former Ph.D. students A. Anastasopoulos and X. Chen, he is coauthor of the book *Iterative Detection: Adaptivity, Complexity Reduction, and Applications*, published by Kluwer Academic Press.

Prof. Chugg received the Fred W. Ellersick award for the best unclassified paper at MILCOM 2003. He has served as an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS and was Program Co-Chair for the Communication Theory Symposium at Globecom 2002.

**Gianluigi Ferrari** (Member, IEEE) was born in Parma, Italy, in November 1974. He received the laurea degree (five-year program, *summa cum laude*) and the Ph.D. degree in electrical engineering from the University of Parma in 1998 and 2002, respectively.



From July 2000 to December 2001, he was a Visiting Scholar at the Communication Sciences Institute, University of Southern California, Los Angeles. Since 2002, he has been a Research Professor with the Department of Information Engineering, University of Parma, where he is now the coordinator of the Wireless Ad-hoc and Sensor Networks (WASN) Laboratory. Between 2002 and 2004, he visited several times, as a Research Associate, the Electrical and Computer Engineering Department at Carnegie Mellon University, Pittsburgh, PA. He has published more than 80 papers in leading international conferences and journals. He is coauthor of the books *Detection Algorithms for Wireless Communications, with Applications to Wired and Storage Systems* (Wiley, 2004), *Teoria della probabilità e variabili aleatorie con applicazioni* (McGraw-Hill, 2005), and *Ad Hoc Wireless Networks: A Communication-Theoretic Perspective* (Wiley, 2006). His research interests include digital communication systems design, ad hoc wireless networking, adaptive signal processing (with particular emphasis on iterative detection techniques for channels with memory), and information theory.

Dr. Ferrari is a co-recipient of a best student paper award at the 2006 International Workshop on Wireless Ad hoc Networks (IWWAN'06). He acts as a frequent reviewer for many international journals and conferences, and also as a technical program member for several international conferences. Since 2007, he has been on the Editorial Board of the *Open Electrical & Electronic Engineering Journal* (TOEEJ).

**Riccardo Raheli** (Member, IEEE) received the Dr. Ing. degree (laurea) in electrical engineering (*summa cum laude*) from the University of Pisa, Italy, in 1983, the M.S. degree in electrical and computer engineering from the University of Massachusetts, Amherst, in 1986, and the Ph.D. degree (Perfezionamento) in electrical engineering (*summa cum laude*) from the Scuola Superiore di Studi Universitari e di Perfezionamento (now S. Anna), Pisa, in 1987.



From 1986 to 1988, he was with Siemens Telecomunicazioni, Cassina de' Pecchi, Milan, Italy. From 1988 to 1991, he was a Research Professor at the Scuola Superiore di Studi Universitari e di Perfezionamento S. Anna. In 1990, he was a Visiting Assistant Professor at the University of Southern California, Los Angeles. Since 1991, he has been with the University of Parma, Italy, where he is currently a Professor of Communications Engineering. His scientific interests are in the general

area of statistical communication theory, with application to wireless, wired and storage systems, and special attention to data detection in uncertain environments, iterative information processing, and adaptive algorithms for communications. His research work has led to numerous scientific publications in leading international journals and conference proceedings, as well as a few industrial patents. He is coauthor of the book *Detection Algorithms for Wireless Communications, with Applications to Wired and Storage Systems* (Wiley, 2004). In 1990, he conceived (with A. Polydoros) the principle of per-survivor processing.

Prof. Raheli served on the Editorial Board of the IEEE TRANSACTIONS ON COMMUNICATIONS as an Editor for Detection, Equalization, and Coding from 1999 to 2003. He also served as a Guest Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, special issue on differential and noncoherent wireless communications, published in September 2005. Since 2003, he has been on the Editorial Board of the *European Transactions on Telecommunications* as an Editor for Communication Theory.