

Graphical models for coded data transmission over inter-symbol interference channels

Michael Tüchler[†], Ralf Koetter[‡], Andrew C. Singer[‡]

[†]Munich University of Technology, Germany, email: `micha@int.ei.tum.de`.

[‡]University of Illinois at Urbana-Champaign, U.S.A, email: `{koetter,acsinger}@uiuc.edu`.

Abstract— We derive graphical models for coded data transmission over channels introducing inter-symbol interference. These models are factor graph descriptions of the transmitter section of the communication system, which serve at the same time as frame work to define the corresponding receiver. The graph structure governs the complexity and nature (e.g., non-iterative, iterative) of the receiver algorithm. A particular graph yields several algorithms optimizing various cost functions depending on the choice of messages communicated along the vertices of the graph. Our contribution is to study these different outcomes of message passing and how the corresponding receiver algorithms are related to existing ones. For example, we show that the tight original factor graph frame work must be slightly relaxed when linear processing is used in the receiver. Besides, we devise strategies to find suitable graphs of the communication problem of interest.

I. INTRODUCTION

We investigate communication systems, where data bits protected by an error-correction code (ECC) are transmitted over a channel introducing inter-symbol interference (ISI). In particular, we consider the serial concatenation of an ECC and an ISI channel separated by an interleaver. The cascade *ECC-interleaver-channel* can be thought of as a single code mapping a data bit sequence to a sequence of output symbols of the channel, which are disturbed by additive noise in the receiver front end. For an overview on serial concatenated systems we refer to [1–3].

A possible criterion to construct such a code is to optimize the performance of sequence- or symbol-based maximum-likelihood (ML) decoding [4], which minimizes the data sequence- or data bit- error rate, respectively. In this context, a good code is a set of channel output sequences with large minimum Euclidean distance. Alternatively, sequence pairs with small distance should occur infrequently. The code performance is majorly improved with increasing sequence length with suitable interleavers [1], but ML decoding for such optimized codes is most often prohibitively complex. After Berrou found with the iterative Turbo decoder [5] a suboptimal but powerful alternative to the ML decoder, this concept was applied to coded data transmission over ISI channels as well, where it is called Turbo equalization [2, 6, 7]. Soon after, the Turbo decoder and the iterative decoder for low-density parity check (LDPC) codes [8, 9] were rederived and analyzed using graphical descriptions [10]. This concept was generalized later to explain a wide array of algorithms in coding and system theory [11–14]. We derive such factor graph descriptions for the cascade *ECC-interleaver-channel* in our example communication system, which serve as frame work

to find suitable receiver algorithms. There is a considerable amount of related work [15–17] focussing on the analysis of a specific graphical model, such as the combination of an LDPC code and a fading channel in [15], or on the impact of the graph structure on the complexity and nature (e.g., non-iterative, iterative) of the receiver algorithm [16, 17].

A particular graph yields different receiver algorithms optimizing various cost functions depending on the choice of messages communicated along the vertices of the graph. Our contribution is to study these different outcomes of message passing and how the corresponding receiver algorithms are related to existing ones. We show that the traditional factor graph frame work must be slightly relaxed when linear processing is used in the receiver. Our main focus is on the detection part, i.e., we study in particular graphical descriptions of the ISI channel being one component of the cascade *ECC-interleaver-channel*. Because of space limitation, we restrict ourselves to the case of known channel characteristics, but we note that this assumption is not at all necessary to apply the graphical models introduced in this paper. In contrast, we hope that the insights enable the reader to easily include unknown channel characteristics into the graphical models as, e.g., in [15, 16].

The paper is organized as follows: Sec. II introduces notation, Sec. III gives a precise definition of the considered communication system and a few applications, Sec. IV briefly introduces the factor graph concept, and Sec. V applies factor graphs to our communication problem (receiver design). Sec. VI concludes the paper.

II. NOTATION

The notation used in this paper is as follows, $\mathbf{0}_i$ is a length- i column vector containing all zeros, \mathbf{I}_i is an $i \times i$ identity matrix, $(\cdot)^T$ is the transpose, $(\cdot)^H$ is the complex conjugate transpose, $\text{Diag}(\mathbf{a})$ is a diagonal matrix constructed from the vector \mathbf{a} , $E(\cdot)$ denotes expectation, and $\text{Cov}(\mathbf{a}, \mathbf{b})$ denotes $E(\mathbf{a}\mathbf{b}^H) - E(\mathbf{a})E(\mathbf{b}^H)$. A vector $\underline{A} = (A_1 A_2 \dots A_n)^T$ of n real-valued random variables A_i with realizations $\mathbf{a} = (a_1 a_2 \dots a_n)^T$ and probability density function (PDF)¹ $p(\mathbf{a})$ is Gaussian distributed with mean $E(\mathbf{a}) = \mathbf{m}$ and covariance matrix $\text{Cov}(\mathbf{a}, \mathbf{a}) = \Sigma$, when $p(\mathbf{a})$

¹We follow the convention that subscripts of the PDFs are dropped if the subscript is the capitalized version of the argument, i.e., we simply write $p(x)$ for the PDF $p_X(x)$ but never $p(y-x)$ for, say, $p_Z(y-x)$. The same holds for expectations, which are denoted the realizations of the random variable as argument, e.g., $E(x)$ denotes the expectation of the random variable X with PDF $p(x)$ but $E_{X|Y}(x)$ denotes the expectation of X over the conditional PDF $p(x|y)$.

is given by $\exp(-\frac{1}{2}(\mathbf{a}-\mathbf{m})^T\boldsymbol{\Sigma}^{-1}(\mathbf{a}-\mathbf{m})) / (2\pi \det(\boldsymbol{\Sigma}))^{n/2}$. In this case, we denote $p(\mathbf{a})$ simply as $\mathcal{N}_{\mathbb{R}}(\mathbf{m}, \boldsymbol{\Sigma})$. The realizations $\mathbf{a} \in \mathbb{C}^n$ of a vector of complex-valued variables A_i can be decomposed into its real and imaginary part, $\mathbf{a} = \mathbf{a}_R + \mathbf{a}_I j$, where $\bar{\mathbf{a}} = (\mathbf{a}_R^T \ \mathbf{a}_I^T)^T$. A complex Gaussian PDF $p(\bar{\mathbf{a}})$ defined over the real-valued argument $\bar{\mathbf{a}} \in \mathbb{R}^{2n}$ is given by $\mathcal{N}_{\mathbb{R}}(\bar{\mathbf{m}}, \bar{\boldsymbol{\Sigma}})$, where $\mathbb{E}(\bar{\mathbf{a}}) = \bar{\mathbf{m}}$ is the length- $2n$ mean vector and $\text{Cov}(\bar{\mathbf{a}}, \bar{\mathbf{a}}) = \bar{\boldsymbol{\Sigma}}$ the $2n \times 2n$ covariance matrix. The PDF $p(\bar{\mathbf{a}})$ is circularly symmetric [18, 19], when the two conditions $\text{Cov}(\mathbf{a}_R, \mathbf{a}_R) = \text{Cov}(\mathbf{a}_I, \mathbf{a}_I)$ and $\text{Cov}(\mathbf{a}_R, \mathbf{a}_I) = -\text{Cov}(\mathbf{a}_R, \mathbf{a}_I)^T$ hold. In this case, the PDF $p(\mathbf{a})$ over the complex-valued argument \mathbf{a} can be written as $\exp(-(\mathbf{a}-\mathbf{m})^H \boldsymbol{\Sigma}^{-1} (\mathbf{a}-\mathbf{m})) / (\pi \det(\boldsymbol{\Sigma}))^n$. Such a PDF is simply denoted as $\mathcal{N}_{\mathbb{C}}(\mathbf{m}, \boldsymbol{\Sigma})$. Circular symmetry follows as well from a vanishing pseudo covariance $\text{Cov}(\mathbf{a}, \mathbf{a}^*) = \text{Cov}(\mathbf{a}_R, \mathbf{a}_R) - \text{Cov}(\mathbf{a}_I, \mathbf{a}_I) + j(\text{Cov}(\mathbf{a}_R, \mathbf{a}_I) + \text{Cov}(\mathbf{a}_I, \mathbf{a}_R))$.

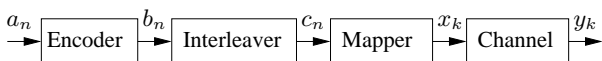


Fig. 1. System configuration.

III. SYSTEM MODEL

Consider the communication link in Fig. 1. The length- K data bit sequence $\mathbf{a} = [a_1 \ a_2 \ \dots \ a_K]^T$, $a_n \in \{0, 1\}$, is encoded to the length- N sequence $\mathbf{b} = [b_1 \ b_2 \ \dots \ b_N]^T$ of code bits $b_n \in \{0, 1\}$ using a binary rate- K/N ECC. The bits in \mathbf{b} are permuted with an interleaver to the sequence $\mathbf{c} = [c_1 \ c_2 \ \dots \ c_N]^T$, whose bits c_n are transmitted over the channel by pulse-amplitude modulation in a discrete-time baseband model. A group $\mathbf{c}_k = (c_{qk-q+1}, \dots, c_{qk})$ of q adjacent bits c_n is mapped into one modulation symbol x_k , where $k = \lfloor n/q \rfloor$. Assuming that $N = Lq$, the transmitted sequence is $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_L]^T$. The symbols x_k are chosen from the alphabet \mathcal{S} , which is typically a subset of the complex numbers \mathbb{C} . We restrict ourselves to alphabets \mathcal{S} of size $|\mathcal{S}| = 2^q$ with unit average power, $|\mathcal{S}|^{-1} \cdot \sum_{s \in \mathcal{S}} |s|^2 = 1$, and zero mean, $\sum_{s \in \mathcal{S}} s = 0$. The sequence of the received symbols $\mathbf{y} = [y_1 \ y_2 \ \dots \ y_L]^T$ is given by

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \quad (1)$$

where $\mathbf{n} = [n_1 \ n_2 \ \dots \ n_L]^T$ is a sequence of independent and identically distributed (I.I.D.) circularly symmetric Gaussian noise samples of variance ν , i.e., the PDF $p(\mathbf{n})$ is given by $\mathcal{N}_{\mathbb{C}}(\mathbf{0}_L, \nu \mathbf{I}_L)$ and the conditional PDF $p(\mathbf{y}|\mathbf{x}, \mathbf{H})$ of the received sequence \mathbf{y} is given by $\mathcal{N}_{\mathbb{C}}(\mathbf{H}\mathbf{x}, \nu \mathbf{I}_L)$. The (in general complex-valued) entries of the $L \times L$ matrix \mathbf{H} , the channel coefficients, cause ISI in the received symbol y_k whenever more than one entry in the k -th row of \mathbf{H} is non-zero. We assume in this paper that the matrix \mathbf{H} is exactly known to the receiver. Among the broad range of possible linear models (1), we study in particular two examples:

Example I: One-dimensional ISI channel

The L symbols x_k are transmitted sequentially over a channel with a length- L_h channel impulse response (CIR). The received symbols are given by

$$y_k = n_k + \sum_{l=0}^{L_h-1} h_{k,l}^* x_{k-l} = n_k + \mathbf{h}_k^H \mathbf{x}_k, \quad k=1, \dots, L, \quad (2)$$

where $\mathbf{h}_k = (h_{k,0} \ \dots \ h_{k,L_h-1})^T$ is the CIR at time step k and $\mathbf{x}_k = (x_k \ \dots \ x_{k-L_h+1})^T$. The corresponding channel ma-

trix \mathbf{H} is lower triangular and banded with bandwidth L_h . The symbols x_k , $k < 1$, transmitted prior to x_1 are assumed to be 0. Such a model is extensively used to explain data transmission over frequency-selective wireline channels, wireless channels with multi-path propagation or in magnetic recording [20].

Example II: Two-dimensional ISI channel

For this model we assume that the L symbols x_k are transmitted in an two-dimensional square array of dimension $S = \sqrt{L}$ such as in two-dimensional magnetic recording [17], where $x_{i,j} = x_{(i-1)S+j}$ is the symbol in the i -th row and j -th column. Thus, the x_k are written into the array row by row from the top left to the bottom right. Given the horizontal length L_h and the vertical length L_v of such a channel, the received symbol $y_{i,j} = y_{(i-1)S+j}$ at position (i, j) , $i, j = 1, 2, \dots, S$, is given by

$$y_{i,j} = n_{i,j} + \sum_{l=0}^{L_h-1} \sum_{m=0}^{L_v-1} h_{i,j,l,m}^* x_{i-l,j-m} = n_{i,j} + \mathbf{h}_{i,j}^H \mathbf{x}_{i,j}, \quad (3)$$

where $n_{i,j} = n_{(i-1)S+j}$. The length- $L_v L_h$ vectors $\mathbf{h}_{i,j}$ (the CIR at position (i, j)) and $\mathbf{x}_{i,j}$ describe the summation in (3). The corresponding channel matrix \mathbf{H} is lower triangular and banded with a bandwidth of $(L_v-1)S + L_h$, where the band consists of a width- L_h band located at the main diagonal and L_v-1 equally spaced width- (L_h-1) bands in the lower triangular part. The symbols $x_{i,j}$, $i, j < 1$, transmitted prior to $x_{1,1}$ are assumed to be 0.

Also possible are extensions of *Example I* to multiple-input multiple-output channels with C inputs and outputs (such as C transmit and receive antennas), yielding a channel matrix \mathbf{H} being $C \times C$ block-diagonal (without ISI in time, but ISI in space) or being $C \times C$ block-wise lower triangular and banded (ISI in time and space).

IV. THE FACTOR GRAPH FRAMEWORK

We present here only a brief introduction of the factor graph concept intensively studied in [11–14]. Consider 6 variables v_i from the alphabet \mathcal{V} combined in $\mathbf{v} = (v_1 \ v_2 \ \dots \ v_6)^T$ and the function $f_A(\mathbf{v}) : \mathcal{V}^6 \mapsto \mathbb{R}$. Suppose that $f_A(\cdot)$ factors into $f_B(v_1, v_2, v_3) \cdot f_C(v_3, v_4) \cdot f_D(v_4) \cdot f_E(v_3, v_5, v_6)$, which is depicted graphically in Fig. 2 on the right side. In this graph, the boxes (function nodes) denote the factors of $f_A(\mathbf{v})$, the circles (variable nodes) denote the variables v_i , and the vertices specify the dependencies between factors and variables. This description leads to the efficient calculation of global functions such as the marginalization over all variables except v_i . Using the notation $g_x(v_i) = \sum_{\mathcal{V}(v_a, v_b, v_i) \text{ except } v_i} f_x(v_a, v_b, v_i)$, we can denote this global function as $g_A(v_i)$. Using the factorization of $f_A(\mathbf{v})$, $g_A(v_6)$ is calculated efficiently as follows:

$$\begin{aligned} g_A(v_6) &= \sum_{\mathcal{V}(v_3, v_5)} \left(\sum_{\mathcal{V}(v_1, v_2)} f_B(v_1, v_2, v_3) \right) \cdot \\ &\quad \left(\sum_{\mathcal{V}v_4} f_C(v_3, v_4) \cdot f_D(v_4) \right) \cdot f_E(v_3, v_5, v_6) \\ &= \sum_{\mathcal{V}(v_3, v_5)} g_B(v_3) \cdot g_F(v_3) \cdot f_D(v_3, v_5, v_6) = g_G(v_3), \end{aligned}$$

where $g_F(v_3)$ is the sum over $f_F(v_3, v_4) = f_B(v_3, v_4) f_C(v_4)$ and $g_G(v_3)$ over $f_G(v_3, v_5, v_6) = g_B(v_3) g_F(v_3) f_C(v_3, v_5, v_6)$.

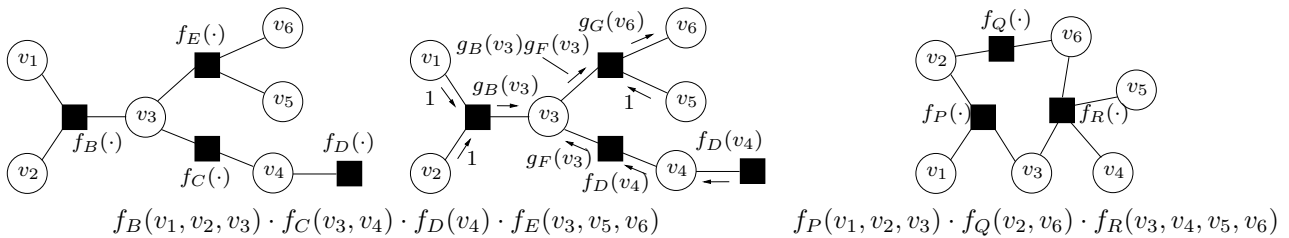


Fig. 2. Factor graphs of different factorizations of the function $f_A(\mathbf{v})$.

The calculation of $g_A(v_6)$ using the step-by-step procedure above can be depicted with a flow of messages towards the node of v_6 in the factor graph of $f_A(\mathbf{v})$ in Fig. 2 (middle), where the function nodes sum the product of the incoming messages and the node function $f_x(\cdot)$ over all variables adjacent to the function node except the one the message is sent to. The variable nodes transmit either the value 1 when they are leaves of the graph or they multiply all their incoming messages and transmit the result to the output. The general update rules for messages from and towards function and variable nodes are shown in Fig. 3, case (1). Once these rules are set, the factor graph can be used to calculate $g_A(v_6)$ and all other global functions $g_A(v_i)$ efficiently by changing the direction of the message flow in the graph towards the node v_i .

The efficiency of calculating $g_A(v_i)$ for all i is evaluated with the number $d \cdot |\mathcal{V}|^{d-1}$ of required summations per function node [14], where $|\mathcal{V}|$ is the size of \mathcal{V} and d , the degree of the node, is the number of vertices connected to that node ($|\mathcal{V}|^{d-1}$ summations per message, overall d messages need to be generated). For $d = 1$, no summations are required. In our example, we need $6 \cdot |\mathcal{V}|^5$ summations to compute $g_A(v_i)$ via $f_A(\cdot)$, but only $3 \cdot |\mathcal{V}|^2 + 3 \cdot |\mathcal{V}|^2 + 2 \cdot |\mathcal{V}|^1$ (nodes for $f_B(\cdot)$, $f_E(\cdot)$, and $f_C(\cdot)$) summations using message passing on the factor graph of $f_A(\mathbf{v})$ (the few extra multiplications are neglected). We state the following strategies to increase the efficiency of calculating global functions:

Factorization. A function of many variables is factorized into factors depending on only a few variables. Thus, a function node with large degree is replaced by a set of new nodes with smaller degree.

Introduction of states (internal variables). A function is augmented with new variables such that a suitable factorization becomes possible. This step complicates the global function, since the number of variables to be summed over increases, but the factorization may yield a tremendous overall increase in efficiency.

Unfortunately, the factor graph approach does not apply to all factorizations of $f_A(\mathbf{v})$. Suppose that $f_A(\mathbf{v})$ factors into $f_P(v_1, v_2, v_3) \cdot f_Q(v_2, v_6) \cdot f_R(v_3, v_4, v_5, v_6)$, which yields the factor graph in Fig. 2 on the left side. Generating and communicating messages according to the update rules in this graph does not produce the correct global functions $g_A(v_i)$ in the variable nodes. Moreover, a prescribed schedule of how to pass the messages as in the tree-type graph in the example before does not exist. Instead, the cycles indicate that the result of message passing may differ de-

pending on how the message update is scheduled, how long the messages are allowed to circulate, and how the messages produced in the variable nodes being part of cycles should be initialized. This problem has been extensively studied, e.g. in [21, 22], and is beyond the scope of this paper. The factor graphs considered here often contain cycles and we assume that the result of message passing is sufficiently close to the desired global function for a suitable number of "passing iterations" and initial conditions.

The type of $f_A(\mathbf{v})$ and the global function can be quite arbitrary as long as the distributive law between the operation of the factorization and that of the global function holds [11, 14]. Recall that our aim is to develop factor graphs leading to efficient receiver algorithms for the communication system in Fig. 1 (even though it is just a representative for many others). To do that, we start by defining global function of type $g(\cdot) : \mathbb{R} \mapsto \mathbb{R}$ and functions $f(\cdot) : \mathcal{V}^6 \mapsto \mathbb{R}$ suitable to our communication problem. The latter will be subject to optimization in later sections using the two previously defined strategies *factorization* and *add states*. Without loss of generality, we still consider the length-6 vector of variables \mathbf{v} .

Marginalization of random variables: Suppose that \mathbf{v} is the realization of a vector of random variables. When \mathcal{V} is a finite-size alphabet, we may regard the function $f(\mathbf{v})$ as probability mass function (PMF), i.e., it satisfies $\sum_{\mathbf{v}} f(\mathbf{v}) = 1$. The global function

$$g(v_i) = \sum_{\mathbf{v} \text{ except } v_i} f(\mathbf{v}) \quad (4)$$

is the marginalization of all variables except v_i , i.e., $g(v_i)$ is the PMF of v_i . The update rules for message passing in Fig. 3, case (2), are identical to those in Fig. 3, case (1), except that the messages traveling on the vertices in the factor graph of $f(\mathbf{v})$ are normalized to be PMFs. This normalization softens numerical problems due to finite precision and/or fixed-comma arithmetic. The PMFs $f(\mathbf{v})$ of interest later often contain factors of type $I(A, B, \dots)$, which is an indicator function over the expressions A, B, \dots :

$$I(A, B, \dots) = I(A) \cdot I(B) \cdot \dots = \begin{cases} 1, & A \text{ and } B \text{ and } \dots \text{ are true,} \\ 0, & \text{else.} \end{cases}$$

For example, the marginalization of the variable v_2 in $I(v_1 = v_2)$, i.e., $\sum_{\mathcal{V} \setminus v_2} I(v_1 = v_2)$, yields v_1 . When \mathcal{V} is a continuous, infinite-size alphabet such as \mathbb{R} or \mathbb{C} , we may regard $f(\mathbf{v})$ as PDF and the marginalization

$$g(v_i) = \int \dots \int f(\mathbf{v}) dv_1 \dots dv_{i-1} dv_{i+1} \dots dv_6 \quad (5)$$

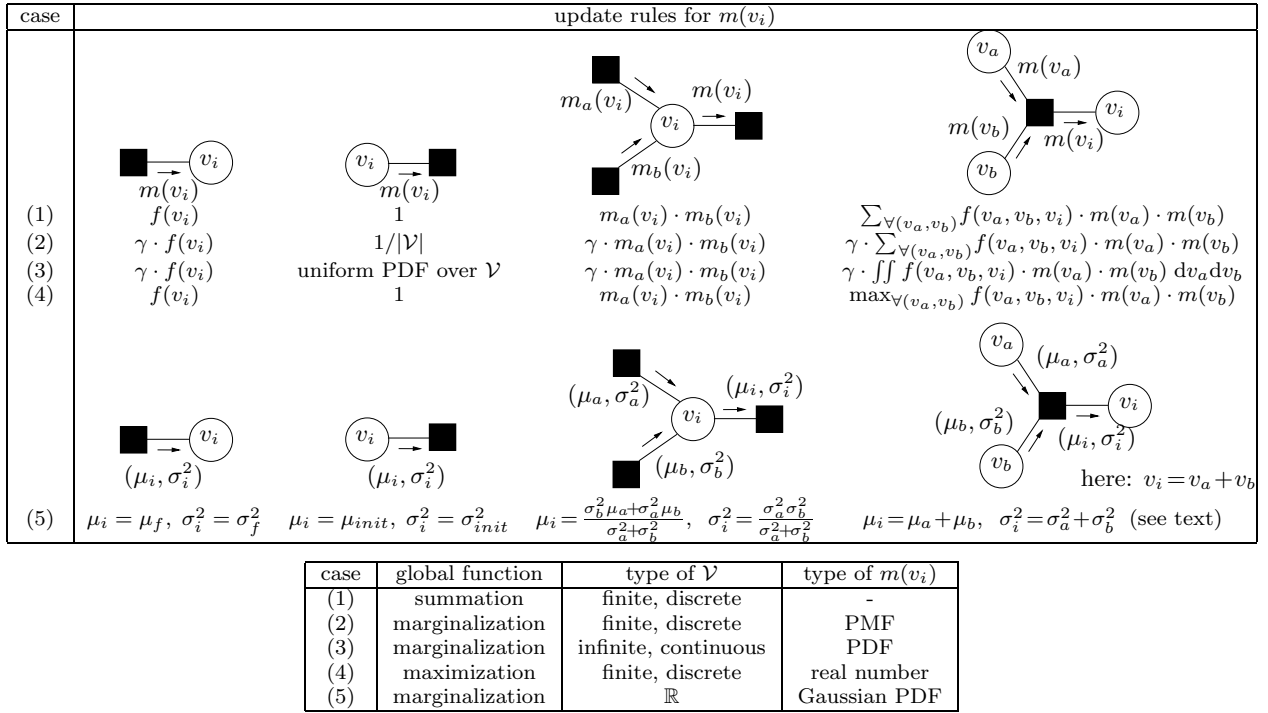


Fig. 3. Update rules for message passing on a factor graph. The constant γ is chosen such that $m(v_i)$ is a PMF, i.e., $1 = \sum_{v_i} m(v_i)$ or a PDF, i.e., $1 = \int m(v_i) dv_i$, respectively.

type of $m(v_i)$	conversion rules between $m(v_i)$ and (μ_i, σ_i^2)		
	$m(v_i)$	(μ_i, σ_i^2)	(μ_i, σ_i^2)
	$\xrightarrow{\square}$	$\xrightarrow{\square}$	$\xrightarrow{\square}$
PMF	$\mu_i = \sum_{v_i} v_i \cdot m(v_i)$	$\sigma_i^2 = \sum_{v_i} (v_i - \mu_i)^2 \cdot m(v_i)$	$m(v_i) = \gamma \cdot \exp(-(v_i - \mu_i)^2 / (2\sigma_i^2)), v_i \in \mathcal{V}$
PDF	$\mu_i = \int v_i \cdot m(v_i) dv_i$	$\sigma_i^2 = \int (v_i - \mu_i)^2 \cdot m(v_i) dv_i$	$m(v_i) = \exp(-(v_i - \mu_i)^2 / (2\sigma_i^2)) / (2\pi\sigma_i^2)^{1/2}$

Fig. 4. Conversion between a message $m(v_i)$ (being a PMF or PDF) over a real-valued alphabet \mathcal{V} and the parameters of an MMSE estimator (estimate μ_i and estimation error variance σ_i^2). The constant γ is chosen such that $m(v_i)$ is a PMF, i.e., $1 = \sum_{v_i} m(v_i)$.

yields the PDF $g(v_i)$ of v_i . Indicator functions are in this case defined like Dirac-delta functions such that for example $\int_{\mathcal{V}} I(v_1 = v_2) dv_2 = v_1$ holds. The corresponding update rules for message passing are depicted in Fig. 3, case (3), including a normalization step yielding that all messages communicated along the vertices of the graph are PDFs. The integration in the function node update in Fig. 3, case (3), is often cumbersome, but this operation can be simple for certain types of messages such as Gaussian PDFs. Therefore, let us assume for now that the variables v_i are real-valued, i.e., all messages are PDFs with real-valued arguments, and investigate under what conditions the update rules in Fig. 3, case (3), produce messages being real-valued Gaussian distributions of type $\mathcal{N}_{\mathbb{R}}(\mu, \sigma^2)$.

The message $m(v_i)$ generated in the leftmost update of Fig. 3, case (3), is Gaussian if the factor $f(v_i)$ takes the quadratic form $\exp(-(v_i - \mu_f)^2 / (2\sigma_f^2))$ such that $m(v_i)$ is the PDF $\mathcal{N}_{\mathbb{R}}(\mu_f, \sigma_f^2)$ after normalization. The message $m(v_i)$ generated in the next update (single variable to function node) should be a uniform PDF over \mathbb{R} in the factor graph setup, which may be modeled with a zero-mean Gaussian PDF with infinite variance, i.e. $\mathcal{N}_{\mathbb{R}}(0, \infty)$. However, we show in Sec. V that this PDF should sometimes be Gaussian with prescribed mean μ_{init} and vari-

ance σ_{init}^2 , which is somewhat an inconsistency of the factor graph framework. The message $m(v_i)$ generated in the next update (two or more messages merge in a variable node) is the product of the Gaussian PDFs $m(v_a)$ given by $\mathcal{N}_{\mathbb{R}}(\mu_a, \sigma_a^2)$ and $m(v_b)$ given by $\mathcal{N}_{\mathbb{R}}(\mu_b, \sigma_b^2)$, which is again Gaussian with variance $\sigma_i^2 = (\sigma_a^{-2} + \sigma_b^{-2})^{-1}$ and mean $\mu_i = \sigma_i^2(\sigma_a^{-2}\mu_a + \sigma_b^{-2}\mu_b)$ after normalization (combination of quadratic terms). The message $m(v_i)$ generated in the last update (two or more messages merge in a function node) follows from integrating $f(v_a, v_b, v_i)m(v_a)m(v_b)$ over (v_a, v_b) . Given the Gaussian PDFs $m(v_a)$ and $m(v_b)$, $m(v_i)$ is Gaussian if $f(v_a, v_b, v_i)$ contains (or factors into) a quadratic expression of the vector $(v_a, v_b, v_i)^T$ (or a subset of it) or indicator functions with linear constraints on v_a, v_b, v_i such as $pv_a + qv_b = v_i, p, q \in \mathbb{R}$. For example, when $f(v_a, v_b, v_i) = I(pv_a + qv_b = v_i)$, we find that $m(v_i)$ is Gaussian with mean $\mu_i = p\mu_a + q\mu_b$ and variance $\sigma_i^2 = p^2\sigma_a^2 + q^2\sigma_b^2$. These update rules for Gaussian messages are shown in Fig. 3, case (5), where we note that the last update is only an example for the specific choice $f(v_a, v_b, v_i) = I(v_a + v_b = v_i)$.

In a general factor graph, the variable nodes may contain vectors of \mathbf{v} (subsets \mathbf{v}_i with more than one variable), such that the messages $m(\mathbf{v}_i)$ are multi-dimensional Gaussian PDFs specified with a mean vector and a covariance

matrix. The update rules in Fig. 3, case (5), are in this case matrix/vector manipulations on these statistics, which have been derived in [23]. The global function (5) is the Gaussian PDF $\mathcal{N}_{\mathbb{R}}(\mu_{g,i}, \sigma_{g,i}^2)$ if all factors of $f(\mathbf{v})$ satisfy the conditions for Gaussian message passing defined above. Both $\mu_{g,i}$ and $\sigma_{g,i}^2$ are computed correctly with Gaussian message passing as long as the factor graph is cycle-free. Surprisingly, the analysis in [24] revealed that even if the graph contains cycles, the means and variances propagated in the graph converge and the available result in the variable nodes yields the correct means $\mu_{g,i}$, but too optimistic variances $\hat{\sigma}_{g,i}^2 < \sigma_{g,i}^2$.

Gaussian message passing can be extended to complex-valued variables. With the decomposition $v_i = v_{i,R} + v_{i,I}j$ of v_i into the real and imaginary part, where $\bar{v}_i = (v_{i,R} \ v_{i,I})^T$, the communicated messages are 2-dimensional PDFs of type $\mathcal{N}_{\mathbb{R}}(\mathbf{m}_i, \mathbf{\Sigma}_i)$, $\mathbf{m}_i = \mathbb{E}(\bar{v}_i)$, $\mathbf{\Sigma}_i = \text{Cov}(\bar{v}_i, \bar{v}_i)$, over the real-valued argument $\bar{v}_i \in \mathbb{R}^2$. The constraints on the factors of $f(\mathbf{v})$ such the messages are Gaussian are straightforward extensions of the real-valued case. We refer to [23] for the update rules of the vector-matrix pairs $(\mathbf{m}_i, \mathbf{\Sigma}_i)$. An alternative way to propagate the parameters of a two-dimensional Gaussian PDF is to use the triple $(\mu_i, \sigma_i^2, \psi_i)$, i.e., the mean $\mu_i = \mathbb{E}(v_i)$, the variance $\sigma_i^2 = \text{Cov}(v_i, v_i)$, and the pseudo-variance $\psi_i = \text{Cov}(v_i, v_i^*)$, which contains the same information as the pair $(\mathbf{m}_i, \mathbf{\Sigma}_i)$ [18]. In case of circular symmetry, where $\psi_i = 0$, the pair (μ_i, σ_i^2) is sufficient and the update rules in Fig. 3, case (5), can be applied. Because of space limitation, we cannot include update rules for Gaussian messages of type $(\mathbf{m}_i, \mathbf{\Sigma}_i)$ (or $(\mu_i, \sigma_i^2, \psi_i)$).

Least-squares problem: Using Gaussian message passing for real-valued variables v_i , the parameters $\mu_{g,i}$ and $\sigma_{g,i}^2$ of the global function $g(v_i)$ can be interpreted as the solution of a least-square problem. We show in Sec. V that $f(\mathbf{v})$ is often a PDF of type $f(\mathbf{v}|ob)$ conditioned on an observation ob . In this case the statistics $\mu_{g,i}$ and $\sigma_{g,i}^2$ are identical to $\mathbb{E}(v_i|ob)$ and $\text{Cov}(v_i, v_i|ob)$, respectively, which is the estimate $\hat{v}_i = \mu_{g,i}$ and the estimation error variance $\text{Cov}(e_i, e_i) = \sigma_{g,i}^2$ of a minimum-mean-square-error (MMSE) estimator aiming to minimize the square of the error $e_i = \hat{v}_i - v_i$ [25, 26]. Thus, using the update rules in Fig. 3, case (5), for an arbitrary alphabet \mathcal{V} , we perform linear MMSE estimation using message passing of the statistics μ_i (being an MMSE estimate) and σ_i^2 (being an estimation error variance). The update rules in Fig. 3, case (5), must only be extended with rules to convert a message $m(v_i)$ being an arbitrary PMF (or PDF) into the pair (μ_i, σ_i^2) and vice-versa. This is done in Fig. 4 for real-valued variables v_i and we note that there are similar conversion rules for complex-valued variables (being circularly symmetric or not). The conversion of $m(v_i)$ into (μ_i, σ_i^2) is simply the calculation of the mean and the variance of the PMF or PDF $m(v_i)$. The conversion of (μ_i, σ_i^2) into $m(v_i)$ uses the standard assumption that the estimate μ_i is Gaussian distributed with variance σ_i^2 [7, 27].

Maximization: Besides the marginalization in (4), (5), it may be of interest to compute the maximum of $f(\mathbf{v})$,

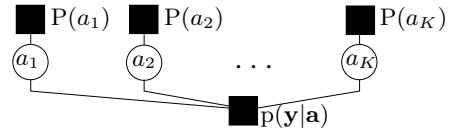


Fig. 5. Factor graph of coded data transmission over an ISI channel.

$$g = \max_{\mathbf{v} \in \mathcal{V}^L} f(\mathbf{v}), \quad (6)$$

using message passing with the update rules in Fig. 3, case (4). The argument \mathbf{v} maximizing $f(\mathbf{v})$ is often of interest, too, and should be propagated as well.

V. GRAPHICAL MODELS FOR CODED DATA TRANSMISSION OVER AN ISI CHANNEL

We say that an optimal receiver for the system in Fig. 1 computes data bit estimates \hat{a}_n minimizing the bit-error probability (BER) $P(a_n \neq \hat{a}_n)$. This is achieved by the choice $\hat{a}_n = \text{argmax}_{a \in \{0,1\}} P(a_n = a|\mathbf{y})$ [4], where $P(a_n = a|\mathbf{y}) = \sum_{\mathbf{a}: a_n = a} P(\mathbf{a}|\mathbf{y})$ is the a-posteriori probability (APP) of a_n given \mathbf{y} . We may also compute the estimate $\hat{\mathbf{a}} = \text{argmax}_{\mathbf{a} \in \{0,1\}^K} P(\mathbf{a}|\mathbf{y})$ minimizing the sequence-error probability $P(\hat{\mathbf{a}} \neq \mathbf{a})$. Using Bayes' rule, we can split $P(\mathbf{a}|\mathbf{y})$ into $p(\mathbf{y}|\mathbf{a}) \cdot P(\mathbf{a})/p(\mathbf{y})$, where the constant $1/p(\mathbf{y})$ not depending on \mathbf{a} can be neglected. Assuming independence of the data bits a_n yields $P(\mathbf{a}) = \prod_{n=1}^K P(a_n)$, where $P(a_n)$ is the a-priori probability that a_n takes on a value from from $\{0, 1\}$. It is convenient to use log-likelihood ratios [28] rather than probabilities when binary variables are concerned such as a_n , but we continue to use probabilities in the factor graph frame work. Using the function

$$f(\mathbf{a}) = p(\mathbf{y}|\mathbf{a}) \cdot P(a_1) \cdot \dots \cdot P(a_K)$$

of the K variables a_n , the optimal receiver algorithm can be written as $\hat{a}_n = \text{argmax}_{a \in \{0,1\}} \sum_{\mathbf{a}: a_n = a} f(\mathbf{a})$ (or $\hat{\mathbf{a}} = \text{argmax}_{\mathbf{a} \in \{0,1\}^K} f(\mathbf{a})$). Performing message passing on the factor graph of $f(\mathbf{a})$ shown in Fig. 5 yields the correct results for $P(a_n = a|\mathbf{y})$ (or $\hat{\mathbf{a}}$) in the variable nodes for a_1, \dots, a_K , since the graph is a tree. However, the complexity of calculating the output messages in the function node $p(\mathbf{y}|\mathbf{a})$ is proportional to $K \cdot 2^{K-1}$, which is prohibitively large for large K . Considering our two strategies *factorization* and *adding states*, we may start with a factorization of $p(\mathbf{y}|\mathbf{a})$ to reduce this complexity. This turns out to be a hard task because of the interleaver in the system. However, after introducing the states \mathbf{c} and \mathbf{x} , we can factorize the resulting function $f(\mathbf{a}, \mathbf{c}, \mathbf{x})$, the joint PDF $p(\mathbf{y}, \mathbf{c}, \mathbf{x}|\mathbf{a})$:

$$f(\mathbf{a}, \mathbf{c}, \mathbf{x}) = p(\mathbf{y}|\mathbf{x}) \cdot P(\mathbf{x}|\mathbf{c}) \cdot P(\mathbf{c}|\mathbf{a}) \cdot P(a_1) \cdot \dots \cdot P(a_K).$$

The PMF $P(\mathbf{x}|\mathbf{c})$ factors into $\prod_{k=1}^L P(x_k|\mathbf{c}_k)$, since a symbol x_k depends only on q code bits c_n . The PMF $P(\mathbf{c}|\mathbf{a})$ merely indicates whether the sequence \mathbf{b} interleaved to \mathbf{c} is the valid codeword encoded from \mathbf{a} .

The factor graph of $f(\mathbf{a}, \mathbf{c}, \mathbf{x})$ is shown in Fig. 6 for the choice $q = 3$. It reveals that the corresponding receiver algorithm is iterative by nature because of the cycles in the graph. In fact, this graph leads to Turbo equalization [6, 7] for a particular choice of scheduling the messages, which is to compute output messages for each vertex of node $p(\mathbf{y}|\mathbf{x})$

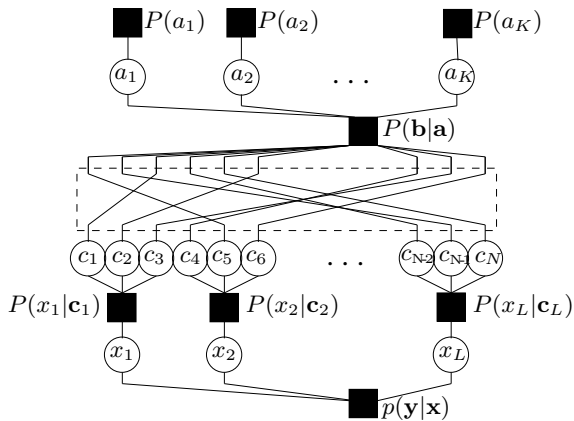


Fig. 6. Factor graph of coded data transmission over an ISI channel leading to Turbo equalization. The modulation parameter q is 3.

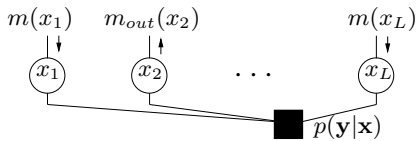


Fig. 7. Local graph of the factor $p(\mathbf{y}|\mathbf{x})$ of the function $f(\mathbf{a}, \mathbf{c}, \mathbf{x})$ with incoming messages $m(x_k)$ and outgoing messages $m_{out}(x_k)$.

(detection), then in the nodes $P(x_k|\mathbf{c}_k)$ (demapping), finally in the node $P(\mathbf{b}|\mathbf{a})$ (decoding) and vice-versa (iterations!). The graph structure and the way the messages are generated exactly predict the ad-hoc guidelines of the Turbo principle [5, 29], which is to use BER-optimal techniques for detection and decoding as well as the concept of using extrinsic probabilities. The message update in the node $P(\mathbf{b}|\mathbf{a})$ is very complex because of the $N+K$ connected vertices. A suitable factorization of this factor may use the parity check matrix of the used ECC yielding a local factor graph containing cycles [11]. The LDPC codes [8, 9] are one class of ECCs optimizing the performance of the corresponding iterative decoding algorithm (message passing on this local graph). Most popular in the Turbo equalization setup are convolutional ECCs, for which efficient trellis factorizations yielding a cycle-free local graph exist (after adding suitable state variables) [11, 12].

Our main focus is on the factor $p(\mathbf{y}|\mathbf{x})$ of the function $f(\mathbf{a}, \mathbf{c}, \mathbf{x})$, whose local factor graph is shown in Fig. 7. Performing message passing on this graph can be regarded as detection part of the receiver algorithm. Indeed, for the BER-optimal receiver, where we apply the global function (4), the incoming messages $m(x_k)$ traveling through the variables nodes for x_k are the PMFs $P(x_k=s)$, which can be thought of as a-priori probabilities that x_k takes on a value s from \mathcal{S} . They are usually set to $1/|\mathcal{S}|$ for all x when the node $p(\mathbf{y}|\mathbf{x})$ computes the outgoing messages $m_{out}(x_k)$ the first time. Using the update rules in Fig. 3, case (2), we find that $m_{out}(x_k=s)$ is given by $\gamma \cdot \sum_{\mathbf{x}:x_k=s} p(\mathbf{y}|\mathbf{x}) \prod_{k=i:k \neq i}^L m(x_k)$, which is the extrinsic APP in the Turbo equalization setup [30]. Thus, message passing on this local graph is identical to APP detection.

Recall that the channel matrix \mathbf{H} is known to the receiver and, thus, merely a parameter of the PDF $p(\mathbf{y}|\mathbf{x})$ given by $\mathcal{N}_{\mathbb{C}}(\mathbf{H}\mathbf{x}, \nu\mathbf{I}_L)$ or $\exp(-\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2/\nu)/(\pi\nu)^L$, respectively. Using the graph in Fig. 7 to perform APP detection is

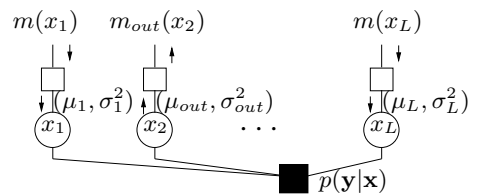


Fig. 8. Local graph of the factor $p(\mathbf{y}|\mathbf{x})$ of the function $f(\mathbf{a}, \mathbf{c}, \mathbf{x})$ used to solve a MMSE estimation problem.

unfeasible because the update complexity in node $p(\mathbf{y}|\mathbf{x})$ is proportional to $L \cdot |\mathcal{S}|^{L-1}$. Before we attempt to find alternative graphical descriptions with reduced complexity, we recall Sec. IV, which states that a factor graph can be used to efficiently solve an MMSE estimation problem.

As shown in Fig. 8, we simply convert the PMFs $m(x_k)$ into the (mean, variance) pair (μ_k, σ_k^2) (for real-valued alphabets \mathcal{S}) and vice-versa for the outgoing message $m_{out}(x_k)$ using the rules in Fig. 4. Now we use the pairs (μ_k, σ_k^2) as parameters of Gaussian distributions $\mathcal{N}_{\mathbb{R}}(\mu_k, \sigma_k^2)$ to find the (mean, variance) pair of the outgoing Gaussian distributions of node $p(\mathbf{y}|\mathbf{x})$ such as $\mathcal{N}_{\mathbb{R}}(\mu_{out}, \sigma_{out}^2)$ of variable x_2 depicted in Fig. 8. From the update rules in Fig. 3, case (3), follows that we have to integrate the following product over all x_k except x_2 :

$$\begin{aligned} & K_0 \cdot \exp(-\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2/\nu) \cdot \prod_{k=1:k \neq 2}^L \exp(-(x_k-\mu_k)^2/(2\sigma_k^2)) \\ &= K_1 \cdot \exp(-\mathbf{x}^H \mathbf{A} \mathbf{x} + 2\text{Re}(\mathbf{b}^H \mathbf{x})) \\ &= K_2 \cdot \exp(-(\mathbf{x}-\mathbf{A}^{-1}\mathbf{b})^H \mathbf{A} (\mathbf{x}-\mathbf{A}^{-1}\mathbf{b})) \end{aligned}$$

where $\mathbf{A} = \mathbf{H}^H \mathbf{H} / \nu + \mathbf{D}$, $\mathbf{D} = \text{Diag}(\sigma_1^{-2} 0 \sigma_3^{-2} \dots \sigma_L^{-2})/2$, $\mathbf{b} = \mathbf{H}^H \mathbf{y} / \nu + [\frac{\mu_1}{\sigma_1^2} 0 \frac{\mu_3}{\sigma_3^2} \dots \frac{\mu_L}{\sigma_L^2}]^T$, and K_0, K_1, K_2 are real-valued constants. Here we arrive at an inconsistency of the factor graph frame work. The incoming message $\mathcal{N}_{\mathbb{R}}(\mu_2, \sigma_2^2)$ of variable x_2 is not a part of the product, because we want to compute the outgoing message for x_2 . Thus, the PDF $\mathcal{N}_{\mathbb{R}}(\mu_2, \sigma_2^2)$ is correctly replaced by the "uniform" distribution $\mathcal{N}_{\mathbb{R}}(0, \infty)$ over \mathbb{R} yielding the zero entry in the matrix \mathbf{D} . A similar purpose has the message send from a leaf variable node to a function node as shown in Fig. 3, case (5), second update. However, we know that x_2 cannot take any value from \mathbb{R} equally likely, since it is drawn from the zero-mean, unit average power alphabet \mathcal{S} . Thus, the appropriate "initial" distribution $\mathcal{N}_{\mathbb{R}}(\mu_{init}, \sigma_{init}^2)$ for the second update rule in Fig. 3, case (5), as well as for the product above is $\mathcal{N}_{\mathbb{R}}(0, 1)$. It follows that the matrix $\mathbf{D}' = \text{Diag}(\sigma_1^{-2} 1 \sigma_3^{-2} \dots \sigma_L^{-2})/2$, should be used instead of \mathbf{D} . The desired parameters μ_{out} and σ_{out}^2 are simply the 2-nd and (2,2)-st entry of $\mathbf{A}^{-1}\mathbf{b}$ and \mathbf{A}^{-1} , respectively, i.e., $\mu_{out} = \text{Re}(\mathbf{u}_2^H \mathbf{H}^H \Sigma^{-1} (\mathbf{y} - \mathbf{H}[\mu_1 0 \mu_3 \dots \mu_L]^T))$ and $\sigma_{out}^2 = 1 - \mathbf{u}_2^H \mathbf{H}^H \Sigma^{-1} \mathbf{H} \mathbf{u}_2$, where $\Sigma = \nu/2 \cdot \mathbf{I}_L + \mathbf{H} \mathbf{D}'^{-1} \mathbf{H}^H$ and \mathbf{u}_2 is a length- L unit column vector with a single one at the 2-nd position. Surprisingly, this solution was already derived in [7, 27] as soft-in soft-out MMSE equalization algorithm in a Turbo equalization setup and the factor graph framework shows that these solutions are optimal among all solutions implementing linear techniques [31–33].

Row-by-row factorization of $p(\mathbf{y}|\mathbf{x})$:

We saw that message passing on the local graph shown in Fig. 7 and 8 is feasible only if linear MMSE equalization

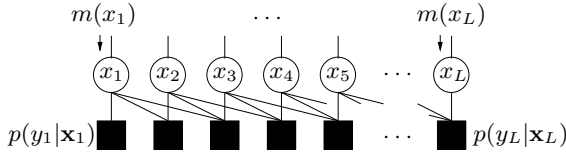


Fig. 9. Factor graph of a row-by-row factorization of $p(\mathbf{y}|\mathbf{x})$ for a one-dimensional length- $L_h = 3$ ISI channel.

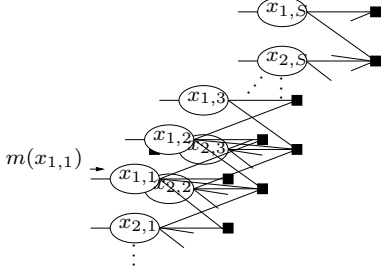


Fig. 10. Factor graph of a row-by-row factorization of $p(\mathbf{y}|\mathbf{x})$ for a two-dimensional ISI channel with $L_h = L_v = 2$.

if performed. However, we can as well apply our two basic rules to reduce the complexity of message passing on this local graph. First, we consider a row-by-row factorization of \mathbf{H} yielding L factors of $p(\mathbf{y}|\mathbf{x})$ depending on y_1, \dots, y_L .

Example I: A one-dimensional length-3 ISI channel

Here, the CIR \mathbf{h}_k is given by $(h_{k,0} \ h_{k,1} \ h_{k,2})^T$ and $p(\mathbf{y}|\mathbf{x})$ factors into $\prod_{k=1}^L p(y_k|\mathbf{x}_k)$, where $p(y_k|\mathbf{x}_k)$ is given by $\mathcal{N}_{\mathbb{C}}(\mathbf{h}_k^H \mathbf{x}_k, \nu)$. The corresponding factor graph is shown in Fig. 9. The update complexity in the L nodes $p(y_k|\mathbf{x}_k)$ is proportional to $L \cdot N_z \cdot |\mathcal{S}|^{N_z-1}$, where $N_z \leq L_h$ is the number of non-zero coefficients in the CIR \mathbf{h}_k . This complexity is significantly smaller than that of the unfactored node $p(\mathbf{y}|\mathbf{x})$, which is $L \cdot |\mathcal{S}|^{L-1}$, because N_z is (usually) much smaller than L . The factor graph contains cycles for $N_z > 2$, i.e., performing message passing using the update rules in Fig. 3, case (3), is iterative, which should be taken into account for the complexity considerations. The same algorithm was derived in [34] without the factor graph concept for sparse but very long CIRs \mathbf{h}_k .

Example II: A 2-dimensional length- 2×2 ISI channel

Here, the CIR at position (i, j) is given by $\mathbf{h}_{i,j} = (h_{i,j,0,0} \ h_{i,j,0,1} \ h_{i,j,1,0} \ h_{i,j,1,1})^T$ and $p(\mathbf{y}|\mathbf{x})$ factors into $\prod_{i=1}^S \prod_{j=1}^S p(y_{i,j}|\mathbf{x}_{i,j})$, $\mathbf{x}_{i,j} = (x_{i,j} \ x_{i,j-1} \ x_{i-1,j} \ x_{i-1,j-1})^T$, where $p(y_{i,j}|\mathbf{x}_{i,j})$ is given by $\mathcal{N}_{\mathbb{C}}(\mathbf{h}_{i,j}^H \mathbf{x}_{i,j}, \nu)$. The corresponding factor graph is shown in Fig. 10. The update complexity in the $L = S^2$ nodes $p(y_{i,j}|\mathbf{x}_{i,j})$ is proportional to $L \cdot N_z \cdot |\mathcal{S}|^{N_z-1}$, where $N_z \leq L_h L_v$ is the number of non-zero coefficients of $\mathbf{h}_{i,j}$. This graph and message passing with the update rules in Fig. 3, case (2), was used in [17].

If the complexity of the update in both examples is still too large, e.g., if L_h , L_v , or $|\mathcal{S}|$ are large, we can perform linear MMSE equalization on the graphs in Fig. 9 and 10. Since the graphs contains cycles, we will not perform exact MMSE estimation, but we know that the estimates μ_{out} arriving in the variable nodes after convergence are correct and the estimation error variances σ_{out}^2 are too optimistic [24]. The update rules are given in Fig. 3, case (5) for real-valued symbols x_k . Of interest is again the function node update depicted in Fig. 11 for *Example I*, where we have

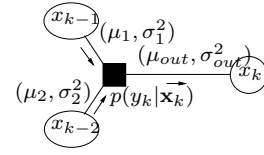


Fig. 11. Function node update performing (local) MMSE equalization.

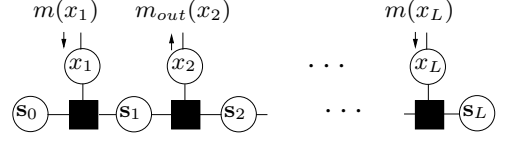


Fig. 12. Factor graph of a state-based factorization of $p(\mathbf{y}|\mathbf{x})$ for the one-dimensional ISI channel.

to integrate the following product over x_{k-1} and x_{k-2} :

$$K_0 \cdot \exp(-|y_k - \mathbf{h}_k^H \mathbf{x}_k|^2 / \nu) \cdot \prod_{i=1}^2 \exp(-(x_{k-i} - \mu_i)^2 / (2\sigma_i^2)) \\ = K_1 \cdot \exp(-(\mathbf{x}_k - \mathbf{A}^{-1} \mathbf{b})^H \mathbf{A} (\mathbf{x}_k - \mathbf{A}^{-1} \mathbf{b})),$$

where $\mathbf{A} = \mathbf{h}_k \mathbf{h}_k^H / \nu + \text{Diag}(0 \ \sigma_1^{-2} \ \sigma_2^{-2}) / 2$, $\mathbf{b} = \mathbf{h}_k y_k / \nu + [0 \ \frac{\mu_1}{\sigma_1^2} \ \frac{\mu_2}{\sigma_2^2}]^T$. Again, we replace \mathbf{A} by $\mathbf{A}' = \mathbf{h}_k \mathbf{h}_k^H / \nu + \text{Diag}(1 \ \sigma_1^{-2} \ \sigma_2^{-2}) / 2$ to solve the inconsistency regarding the initial distribution of x_k (assumed to be Gaussian), which should be $\mathcal{N}_{\mathbb{R}}(0, 1)$ rather than $\mathcal{N}_{\mathbb{R}}(0, \infty)$. The desired parameters μ_{out} and σ_{out}^2 are simply the 1-st and (1,1)-st entry of $\mathbf{A}'^{-1} \mathbf{b}$ and \mathbf{A}'^{-1} , respectively, i.e., $\mu_{out} = \text{Re}(h_{k,0}(y_k - \mathbf{h}_k^H [0 \ \mu_1 \ \mu_2]^T) / S)$ and $\sigma_{out}^2 = 1 - |h_{k,0}|^2 / S$, where $S = \nu / 2 + \mathbf{h}_k^H \text{Diag}(1 \ \sigma_1^2 \ \sigma_2^2) \mathbf{h}_k$.

State factorization of $p(\mathbf{y}|\mathbf{x})$:

The row-by-row factorization of $p(\mathbf{y}|\mathbf{x})$ in Figs. 9 and 10 produces graphs with cycles. Alternative factorizations are possible by augmenting $p(\mathbf{y}|\mathbf{x})$ with suitable states:

Example I: A one-dimensional length-3 ISI channel

The channel law $y_k = n_k + \mathbf{h}_k^H \mathbf{x}_k$ in (2) is described by the state equations $\mathbf{s}_k = \mathbf{A} \mathbf{s}_{k-1} + \mathbf{B} x_k$ and $y_k = \mathbf{C} \mathbf{s}_{k-1} + \mathbf{D} x_k + n_k$, where $\mathbf{s}_k = [x_k \ x_{k-1}]^T$ with the initial state $\mathbf{s}_0 = [0 \ 0]^T$ and

$$\mathbf{A} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{C} = [h_{k,1} \ h_{k,2}], \quad \mathbf{D} = h_{k,0}.$$

Augmenting $p(\mathbf{y}|\mathbf{x})$ with all states to $p(\mathbf{s}_0, \dots, \mathbf{s}_L, \mathbf{y}|\mathbf{x})$ and applying the chain rule yields the factorization $\prod_{k=1}^L p(y_k|x_k, \mathbf{s}_{k-1}) \cdot P(\mathbf{s}_k|x_k, \mathbf{s}_{k-1})$, where $p(y_k|x_k, \mathbf{s}_{k-1})$ is given by $\mathcal{N}_{\mathbb{R}}(\mathbf{h}_k^H \mathbf{x}_k, \nu)$ and $P(\mathbf{s}_k|x_k, \mathbf{s}_{k-1})$ is the indicator function $\gamma \cdot I(\mathbf{s}_k = \mathbf{A} \mathbf{s}_{k-1} + \mathbf{B} x_k)$ (a PMF) with γ being a normalizing constant. The corresponding factor graph is depicted in Fig. 12. This graph is obviously cycle-free and performing message passing on this subgraph to solve the global function (4) yields the correct result, the extrinsic APPs in the Turbo equalization setup [30] as shown in Sec. V. Computing the messages (from left to right, right to left, and towards the nodes for x_k) corresponds to the forward and backward recursion of the BCJR algorithm [35] applied to APP detection.

Example II: A 2-dimensional length- 2×2 ISI channel

The channel law $y_k = n_{i,j} + \mathbf{h}_{i,j}^H \mathbf{x}_{i,j}$ in (3) is described with the state equations

$$\begin{bmatrix} \mathbf{s}_{i,j} \\ \mathbf{t}_{i,j} \end{bmatrix} = \mathbf{A} + \mathbf{B} x_{i,j}, \quad y_{i,j} = \mathbf{C} \begin{bmatrix} \mathbf{s}_{i-1,j} \\ \mathbf{t}_{i,j-1} \end{bmatrix} + \mathbf{D} x_{i,j} + n_{i,j}.$$

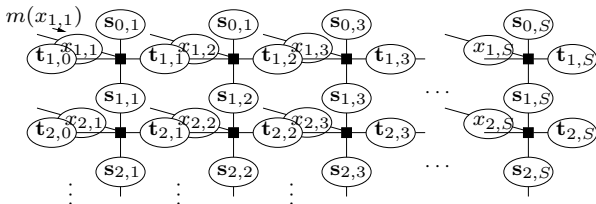


Fig. 13. Factor graph of a state-based factorization of $p(\mathbf{y}|\mathbf{x})$ for the two-dimensional ISI channel.

where $\mathbf{s}_{i,j} = [x_{i,j} \ x_{i,j-1}]^T$ and $\mathbf{t}_{i,j} = x_{i,j}$ with the initial states $\mathbf{s}_{0,j} = [0 \ 0]^T$ for all j , $\mathbf{t}_{i,0} = 0$ for all i , and

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} h_{i,j,1,0} \\ h_{i,j,1,1} \\ h_{i,j,0,1} \end{bmatrix}^T, \quad \mathbf{D} = h_{i,j,0,0}.$$

Augmenting $p(\mathbf{y}|\mathbf{x})$ with all state variables and applying the chain rule yields the following factorization:

$$p(\mathbf{s}_{0,1}, \dots, \mathbf{s}_{S,S}, \mathbf{t}_{1,0}, \dots, \mathbf{t}_{S,S} | \mathbf{y}, \mathbf{x}) = \prod_{i=1}^S \prod_{j=1}^S p(y_{i,j} | x_{i,j}, \mathbf{s}_{i-1,j}, \mathbf{t}_{i,j-1}) \cdot P(\mathbf{s}_{i,j} | x_{i,j}, \mathbf{t}_{i,j-1}) \cdot P(\mathbf{t}_{i,j} | x_{i,j}),$$

where $p(y_{i,j} | x_{i,j}, \mathbf{s}_{i-1,j}, \mathbf{t}_{i,j-1})$ is given by $\mathcal{N}_{\mathbf{C}}(\mathbf{h}_{i,j}^H \mathbf{x}_{i,j}, \nu)$, $P(\mathbf{s}_{i,j} | x_{i,j}, \mathbf{t}_{i,j-1})$ is the indicator function $\gamma_0 \cdot I(\mathbf{s}_{i,j} = \mathbf{A}_{st} \mathbf{t}_{i,j-1} + \mathbf{B}_s x_{i,j})$, and $P(\mathbf{t}_{i,j} | x_{i,j})$ is the indicator function $\gamma_1 \cdot I(\mathbf{t}_{i,j} = \mathbf{B}_s x_{i,j})$. The corresponding factor graph is depicted in Fig. 13. This graph is not cycle-free, but the connectivity compared to the graph in Fig. 10 is greatly reduced. Even though there are many other state factorizations of $p(\mathbf{y}|\mathbf{x})$, it turns out to be impossible to find a cycle-free graph for a general 2-dimensional ISI channel, i.e., no efficient symbol- (global function (4)) or sequence-based detection algorithm (global function (6)) exists [17].

VI. CONCLUSIONS

We showed why iterative receiver algorithms such as Turbo equalization work well for coded data transmission over ISI channels, because they are constructed to solve an optimal global function iteratively using message passing on a graph with cycles. This holds as well for "unusual" algorithms such as linear soft-in soft-out MMSE equalization [7]. We presented factor graph examples for different linear models (1), which yield efficient receiver algorithms, in particular for detection. Special attention was paid to linear algorithms (iterative or not).

REFERENCES

- [1] S. Benedetto et al., "Serial concatenation of interleaved codes: performance analysis design, and iterative decoding," *IEEE Trans. on Information Theory*, vol. 44, pp. 909–926, May 1998.
- [2] K. Chugg, A. Anastasopoulos, and X. Chen, *Iterative Detection*. Boston: Kluwer Academic Press, 2001.
- [3] M. Tüchler, "Design of serially concatenated systems depending on the block length," in *Proc. ICC 2003, Anchorage*, May 2003.
- [4] J. Proakis, *Digital Communications, 3rd Ed.* Singapore: McGraw-Hill, 1995.
- [5] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. IEEE Intern. Conf. on Comm., Geneva*, May 1993.
- [6] C. Douillard et al., "Iterative correction of intersymbol interference: Turbo equalization," *European Trans. on Telecomm.*, vol. 6, pp. 507–511, Sep-Oct 1995.
- [7] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: principles and new results," *IEEE Trans. on Comm.*, vol. 50, pp. 754–767, May 2002.
- [8] R. Gallager, "Low density parity check codes," *IRE Trans. on Information Theory*, vol. 8, pp. 21–28, January 1962.
- [9] D. J. C. MacKay, "Good error correcting codes based on very sparse matrices," *IEEE Trans. on Information Theory*, vol. 45, pp. 399–431, March 1999.
- [10] N. Wiberg, *Codes and decoding on general graphs*. PhD thesis, Linköping University, 1996.
- [11] F. Kschischang, B. Frey, and A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. on Information Theory*, vol. 47, pp. 498–519, Feb 2001.
- [12] G. Forney, "Codes on graphs: Normal realizations," *IEEE Trans. on Information Theory*, vol. 47, pp. 520–548, Feb 2001.
- [13] G. Forney, *preprint: Codes on graphs: Generalized state realizations*. 1999.
- [14] A. Ji and R. McEliece, "The generalized distributive law," *IEEE Trans. on Information Theory*, vol. 46, pp. 325–343, March 2000.
- [15] H. Niu, M. Shen, J. Ritsey, H. Liu, "Iterative channel estimation and LDPC decoding over flat-fading channels: a factor graph approach," in *Proc. Conf. on Information Sciences and Systems (CISS)*, Baltimore, March 2003.
- [16] A. Worthen and W. Stark, "Unified design of iterative receivers using factor graphs," *IEEE Trans. on Information Theory*, vol. 47, pp. 843–849, Feb 2001.
- [17] N. Singla, J. O'Sullivan, R. Indeck, and Y. Wu, "Iterative decoding and equalization for 2-D recording channels," *IEEE Transactions on Magnetics*, vol. 38, pp. 2328–2330, Sep 2002.
- [18] F. Neeser and J. Massey, "Proper complex random processes with applications to information theory," *IEEE Trans. on Information Theory*, vol. 39, pp. 1293–1302, July 1993.
- [19] B. Picinbono and C. Chevalier, "Widely linear estimation with complex data," *IEEE Trans. on Signal Processing.*, vol. 43, pp. 2030–2033, Aug 1995.
- [20] K. Imminck, P. Siegel, and J. Wolf, "Codes for digital recorders," *IEEE Trans. on Inf. Theory*, vol. 44, pp. 2260–2299, Oct 1998.
- [21] S. ten Brink, "Convergence behaviour of iteratively decoded parallel concatenated codes," *IEEE Trans. on Comm.*, vol. 49, pp. 1727–1737, Oct 2001.
- [22] T. Richardson and R. Urbanke, "The capacity of low density parity-check codes under message passing decoding," *IEEE Trans. on Information Theory*, vol. 47, pp. 599–618, Feb 2001.
- [23] H. Loeliger, *Least Squares and Kalman Filtering on Forney Graphs*. Codes, Graphs, and Systems, R.E. Blahut and R. Koetter, eds., Kluwer, 2002.
- [24] Y. Weiss and W. Freeman, "On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs," *IEEE Trans. on Inf. Theory*, vol. 47, pp. 736–744, Feb 2001.
- [25] S. Haykin, *Adaptive Filter Theory, 3rd Ed.* Upper Saddle River, New Jersey: Prentice Hall, 1996.
- [26] H. Poor, *An Introduction to Signal Detection and Estimation, 2nd Ed.* New York: Springer Verlag, 1994.
- [27] X. Wang and H. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. on Comm.*, vol. 47, no. 7, pp. 1046–1061, 1999.
- [28] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. on Information Theory*, pp. 429–445, March 1996.
- [29] J. Hagenauer, "The turbo principle: Tutorial introduction and state of the art," in *Proc. Intern. Symp. on Turbo Codes, Brest, France*, pp. 1–11, Sep 1997.
- [30] G. Bauch and V. Franz, "A comparison of soft-in/soft-out algorithms for 'Turbo detection'," in *Proc. Intern. Conf. on Telecomm.*, pp. 259–263, June 1998.
- [31] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. of the Intern. Symposium on Turbo codes, Brest, France*, pp. 96–102, September 1997.
- [32] Z. Wu and J. Cioffi, "Turbo decision aided equalization for magnetic recording channels," in *Proc. Global Telecomm. Conf.*, pp. 733–738, Dec 1999.
- [33] D. Raphaeli and A. Saguy, "Linear equalizers for Turbo equalization: A new optimization criterion for determining the equalizer taps," in *Proc. 2nd Intern. Symp. on Turbo codes, Brest, France*, pp. 371–374, Sep 2000.
- [34] J. Yu, Y. Li, and S. Yoshida, "Split soft-decision equalization for wireless channels with large delay spread," in *Proc. PIMRC, Lissabon, Portugal*, Sep 1999.
- [35] L.R. Bahl et al., "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Information Theory*, vol. 20, pp. 284–287, March 1974.