ITERATIVE DECODING AS DYKSTRA'S ALGORITHM WITH ALTERNATE I-PROJECTION AND REVERSE I-PROJECTION

Florence Alberge

Laboratoire des Signaux et Systèmes
Univ. Paris-Sud, UMR 8506, Supelec
Orsay, F91405
phone: +33 1 69 85 17 57 - fax: +33 1 69 85 17 69 - email : alberge@lss.supelec.fr

ABSTRACT

Bit interleaved Coded Modulation with iterative decoding is known to provide excellent performance over both Gaussian and fading channels. However a complete analysis of the iterative demodulation is still lacking. In this paper, we complete the geometric interpretation of turbo-like iterative demodulation and we give an interpretation for the extrinsic propagation. We prove that iterative demodulation is the Dykstra's algorithm with I-projections and reverse I-projections.

1. INTRODUCTION

Bit-Interleaved Coded Modulation (BICM) was first suggested by Zehavi in [1] to improve the Trellis Coded Modulation performance over Rayleigh-fading channels. In BICM, the diversity order is increased by using bit-interleavers instead of symbol interleavers. This improvement is achieved at the expense of a reduced minimum Euclidean distance leading to a degradation over non-fading Gaussian channels [1], [2]. This drawback can be overcome by using iterative decoding (BICM-ID) at the receiver. BICM-ID is known to provide excellent performance for both Gaussian and fading channels.

The iterative decoding scheme used in BICM-ID is very similar to serially concatenated turbo-decoders. Indeed, the serial turbo-decoder makes use of an exchange of information between computationally efficient decoders for each of the component codes. In BICM-ID, the inner decoder is replaced by demapping which is less computationally demanding than a decoding step. Even if this paper focus on iterative decoding for BICM, the results can be applied to the large class of iterative decoders including serial or parallel concatenated turbo decoders as long as low-density parity-check (LDPC) decoders. The turbo-decoder and more generally iterative decoding was not originally introduced as the solution to an optimization problem rendering the analysis of its convergence and stability very difficult. Among the different attempts to provide an analysis of iterative decoding, the EXIT chart analysis and density evolution have permitted to make significant progress [3] but the results developed within this setting apply only in the case of large block length. Another tool of analysis is the connection of iterative decoding to factor graphs [4] and belief propagation [5]. Convergence results for belief propagation exists but are limited to the case where the corresponding graph is a tree which does not include turbo code or LDPC. A link between iterative decoding and classical optimization algorithms has been made recently in [6] where the turbo decoding is interpreted as a nonlinear block Gauss Seidel iteration for solving a constrained optimization problem. In parallel, a geometrical approach has been considered and provides and interesting interpretation in terms of projections. The particular case of BICM-decoding has been studied by Muquet in [7] where the decoding sub-block is interpreted as two successive projections. The interpretation of the demapping sub-block in terms of projection remains unachieved. In [8], the turbo-decoding is interpreted in a geometric setting as a dynamical system leading to new but incomplete results. The failure to obtain complete results is mainly due to the inability to efficiently describe extrinsic information passing in terms of information projection.

Here, we emphasize the connection between iterative decoding and the Dykstra's algorithm from the convex optimization literature [9]. The extrinsics are exactly the deflected versions of the previous outputs passed through the blocks in the Dykstra's algorithm.

2. TOOLS

2.1 BICM-ID with soft decision feedback

A conventional BICM system [2] is built from a serial concatenation of a convolutional encoder, a bit interleaver and an M-ary bits-to-symbol mapping (where $M = 2^m$) as shown in fig. 1. The sequence of information bits $\underline{\mathbf{b}}$ is first encoded by a convolutional encoder to produce the output encoded bit sequence $\underline{\mathbf{c}}$ of length L_c which is then scrambled by a bit interleaver (as opposed to the channel symbols in the symbol-interleaved coded sequence) operating on bit indexes. Let $\underline{\mathbf{d}}$ denote the interleaved sequence. Then, m consecutive bits of $\underline{\mathbf{d}}$ are grouped as a channel symbol $\mathbf{d}_k = (d_{km+1}, \ldots d_{(k+1)m})$. The complex transmitted signal $s_k = \varepsilon(\mathbf{d}_k)$, is then chosen from an M-ary constellation Ψ where ε denotes the mapping scheme. For simplicity, we consider transmission over the AWGN channel. The received signals can be written as :

$$y_k = s_k + n_k \quad 1 \le k \le L_c/m \tag{1}$$

where n_k is a complex white Gaussian noise with independent in-phase and quadrature components having two-sided power spectral density σ_c^2.

Due to the presence of the random bit interleaver, the true

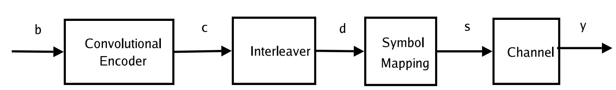

Figure 1: Transmission model

maximum likelihood decoding of BICM is too complicated to implement in practice. Figure 2 shows the block diagram of the receiver for a BICM-ID system with soft-decision feedback. In the first iteration, the encoded bits are assumed

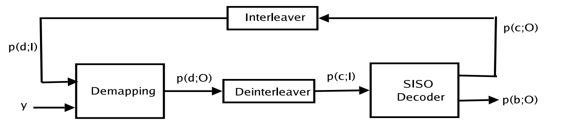

Figure 2: Receiver for a BICM-ID with soft-decision feedback

equally likely. The demapping consists in evaluating *a posteriori* probabilities (APP) for the encoded bits without accounting for the code structure, namely:

$$p_{APP}(d_{km+i}=b)=p(d_{km+i}=b\mid y_k)\sim\sum_{s_k\in\Psi_b^i}p(y_k\mid s_k)p(s_k) \quad (2)$$

where Ψ_b^i, $b\in\{0,1\}$, denotes the subset of Ψ that contains all symbols whose labels have the value b in the i^{th} position. In the turbo decoding process, the quantities exchanged through the blocks are not *a posteriori* probabilities (APP) but extrinsic information [10]. The extrinsic information at the output of the demapping $p(d_{km+i};O)$ is computed as $p_{APP}(d_{km+i})/p(d_{km+i};I)$ where $p(d_{km+i};I)$ is the *a priori* information for the demapping sub-block. Since the bit interleaver makes the bits independent, the extrinsic information $p(d_{km+i};O)$ reads:

$$p(d_{km+i}=b;O)=K_m\sum_{s_k\in\Psi_b^i}p(y_k\mid s_k)\prod\nolimits_{j\neq i}p(d_{km+j};I) \quad (3)$$

and the corresponding APP reads:

$$p_{APP}(d_{km+i}=b)=K_m'\sum_{s_k\in\Psi_b^i}p(y_k\mid s_k)\prod_{1\leq j\leq m}p(d_{km+j};I) \quad (4)$$

where K_m and K_m' are normalization factors and O and I refer to the output and the input. Note that $p(d_{km+i};O)$ is computed from the *a priori* probabilities $p(d_{km+j};I)$ of the others bits on the same channel symbol. However, we can also write the APP using the whole sequence as:

$$p_{APP}(d_{km+i}=b)=p(d_{km+i}=b\mid y_k)\sim\sum_{\mathbf{s}:s_k\in\Psi_b^i}p(\mathbf{y}\mid\mathbf{s})p(\mathbf{s}) \quad (5)$$

The extrinsic information $p(d_{km+i};O)$ is de-interleaved and delivered to the SISO decoder [11] as an *a priori* information on the encoded bits. Let $c_l=d_{\sigma^{-1}(km+j)}$ where σ^{-1} is for the permutation on the indexes due to the deinterleaver; $p(c_l;I)$ is the updated input of the Soft Input Soft Output (SISO) decoder. The extrinsic information at the output of the SISO decoder is obtained through:

$$p(c_l=b;O)=K_c\sum_{\mathbf{c}\in\mathscr{R}_b^l}\mathbf{I}_{\mathscr{C}}(\mathbf{c})\prod_{j\neq l}p(c_j;I) \quad (6)$$

and the corresponding APP is:

$$p_{APP}(c_l=b)=K_c'\sum_{\mathbf{c}\in\mathscr{R}_b^l}\mathbf{I}_{\mathscr{C}}(\mathbf{c})\prod_{1\leq j\leq L_c}p(c_j;I) \quad (7)$$

where $\mathbf{I}_{\mathscr{C}}(\mathbf{c})$ stands for the indicator function of the code, i.e. $\mathbf{I}_{\mathscr{C}}(\mathbf{c})=1$ if $\mathbf{c}$ is a codeword and 0 otherwise and $\mathscr{R}_b^l$ denotes the set of binary words of length L_c with value b in the l^{th} position. K_c and K_c^l are normalization factors. The extrinsic information $p(c_l;O)$ is interleaved and delivered to the demapping sub-block as a regenerated *a priori* information. The process is continued until the APP at the output of the two sub-blocks are the same or until the maximal iteration number is reached.
In the next section, we give an interpretation of the demodulation process via information geometry.

2.2 Simple facts from information geometry

Suppose that p and q are probability measures defined on subsets of $\mathscr{H}$ where $\mathscr{H}$ is the set of the first 2^{L_c} integers. The I-divergence of p with respect to q also called Kullback-Leibler divergence is given by:

$$D(p\parallel q)=\sum_{k=1}^{k=2^{L_c}}p(k)ln(p(k)/q(k)) \quad (8)$$

The minimum of $D(p\parallel q)$ for p in a subset $\mathscr{S}$ of $\mathscr{H}$ is denoted $D(\mathscr{S}\parallel q)$. If a unique minimizer exists it is called the I-projection of q to $\mathscr{S}$. Similarly, the minimum of $D(p\parallel q)$ for q in a subset $\mathscr{S}$ of $\mathscr{H}$ is denoted $D(p\parallel\mathscr{S})$. If a unique minimizer exists it is called the reverse I-projection of p to $\mathscr{S}$ [12]. These projections can also be termed respectively as backward and forward Bregman projection based on the Bregman distance D_f with $f(\mathbf{x})=xln(x)-x$. Now, we examine the projections onto linear and exponential families. They are of particular interest in our context and they give rise to a Pythagorean theorem.

Definition 1 ([13]) *For any given function $f_1,f_2,...,f_r$ on $\mathscr{H}$ and numbers $\alpha_1,\alpha_2,...,\alpha_r$, the set*

$$\mathscr{L}=\{p:\sum_{\mathbf{x}}p(\mathbf{x})f_i(\mathbf{x})=\alpha_i,1\leq i\leq k\}$$

if non empty, will be called a linear family of probability distributions. Moreover the set $\mathscr{E}$ of all p such that

$$p(\mathbf{x})=cq(\mathbf{x})exp(\sum_{i=1}^{r}\theta_i f_i(\mathbf{x})),\ \ for\ \ some\ \ \theta_1,...,\theta_r$$

will be called an exponential family of probability distributions; here q is a given distribution and c is a normalization factor.

The linear family $\mathscr{L}$ is completely defined by the functions $f_1,f_2,...,f_r$ and the scalars $\alpha_1,\alpha_2,...,\alpha_r$. The exponential family $\mathscr{E}$ is completely defined by the distribution q (which belongs to $\mathscr{E}$) and the functions $f_1,f_2,...,f_r$.
Let $\mathscr{E}_{\mathscr{P}}$ denote the product manifold *ie* the set of all L_c-variate distribution with independent components. This family will reveal to be of great importance for our study. It is clear that $\mathscr{E}_{\mathscr{P}}$ is an exponential family [7]. The Pythagorean theorems are stated below:

Theorem 1 ([13]) *The I-projection p^* of q onto a linear family $\mathscr{L}$ is unique and satisfies the* Pythagorean identity

$$D(p\parallel q)=D(p\parallel p^*)+D(p^*\parallel q)\quad\forall p\in\mathscr{L}$$

Similarly, it exists a Pythagorean identity for reverse I-projections on exponential families.

Theorem 2 ([12]) *The reverse I-projection q^* of p onto an exponential family $\mathscr{E}$ is unique and satisfies the* Pythagorean identity

$$D(p \parallel q) = D(p \parallel q^*) + D(q^* \parallel q) \quad \forall q \in \mathscr{E}$$

For I-projections onto $\mathscr{L}$, analytical results exist. The I-projection p^* of q onto a linear family (using notations in definition 1) reads [14]:

$$p^*(\mathbf{x}) = q(\mathbf{x})exp(-\sum_{1}^{r} \mu_j (f_j(\mathbf{x}) - \alpha_j)) \tag{9}$$

where the $\{\mu_j\}$ are Lagrange multipliers determined from the constraints. In the particular case where the functions $\{f_i\}$ in the definition of the linear family are the parity-check equations of a code and if $\{\alpha_i\} = 0$, then the I-projection in (9) reads [14]:

$$p^*(\mathbf{x}) = q(\mathbf{x})exp(-\mu_0)I_1(\mathbf{x})I_2(\mathbf{x})...I_r(\mathbf{x}) \tag{10}$$

where $I_i(\mathbf{x})$ is the indicator function for all vectors $\mathbf{x}$ which satisfy constraint f_i and $exp(-\mu_0)$ is a normalization constant. The indicator function for a codeword is $I_{\mathscr{C}}(\mathbf{x}) = I_1(\mathbf{x})I_2(\mathbf{x})...I_r(\mathbf{x})$ since a codeword must satisfy all of the parity constraint simultaneously. Thus, the I-projection p^* of q onto the linear family $\mathscr{L}_{\mathscr{C}}$ formed with the parity-check equations of the code $\mathscr{C}$ reads:

$$p^*(\mathbf{x}) = \frac{q(\mathbf{x})I_{\mathscr{C}}(\mathbf{x})}{\sum_{\mathbf{x}} q(\mathbf{x})I_{\mathscr{C}}(\mathbf{x})} \tag{11}$$

The reverse I-projection q^* of p onto $\mathscr{E}_{\mathscr{P}}$ also admits a closed-form expression:

$$q^*(\mathbf{x}) = q^*(x_1, x_2, ...x_s) = \prod_i p_i(x_i) \tag{12}$$

where $p_i(x_i)$ is the marginal distribution on x_i of the probability measure p. In the next section, the link between I-projections, reverse I-projections and iterative decoding is emphasized.

2.3 Interpretation of iterative decoding

Let $p_{APP}(\mathbf{d})$ respectively $p_{APP}(\mathbf{c})$ denote the probability measure belonging to the product space $\mathscr{E}_{\mathscr{P}}$ with marginal distributions $p_{APP}(d_{km+i})_{i=1,..,m;k=1,..L_c/m}$ respectively $p_{APP}(c_l)_{1 \le l \le L_c}$. From (5) and (12), we can conclude that $p_{APP}(\mathbf{d})$ is the reverse I-projection of $p(\mathbf{y} \mid \mathbf{s})p(\mathbf{d};I)$ onto $\mathscr{E}_{\mathscr{P}}$. With an AWGN channel, $p(\mathbf{y} \mid \mathbf{s})$ is a Gaussian distribution, namely

$$p(\mathbf{y} \mid \mathbf{s}) = Kexp(-\frac{\parallel \mathbf{y} - \mathbf{s} \parallel^2}{2\sigma_c^2})$$

where K is a normalization factor. Thus, from (9), $p(\mathbf{y} \mid \mathbf{s})p(\mathbf{d};I)$ is the I-projection of $p(\mathbf{d};I)$ onto the linear family $\mathscr{L}_{\mathscr{M}} = \{p : \sum_{\mathbf{x}} p(\mathbf{x}) = 1 \quad \sum_{\mathbf{x}} p(\mathbf{x}) \parallel \mathbf{y} - \mathbf{x} \parallel^2 = \alpha(\sigma_c^2)\}$. Let $\Pi_{\mathscr{L}}$ respectively $\Pi^r_{\mathscr{E}}$ denote the I-projection onto the linear family $\mathscr{L}$ respectively the reverse I-projection onto the exponential family $\mathscr{E}$. Thus $p_{APP}(\mathbf{d}) = D_{\mathscr{M}}(p(\mathbf{d};I))$ where:

$$D_{\mathscr{M}} : \mathscr{E}_{\mathscr{P}} \to \mathscr{E}_{\mathscr{P}} : \mathbf{q} \mapsto \Pi^r_{\mathscr{E}_{\mathscr{P}}}[\Pi_{\mathscr{L}_{\mathscr{M}}}(\mathbf{q})]$$

Note that the linear family involved in the demapping is changing from one iteration to another. However, at each iteration, it exists a linear family such that $p(y \mid s)p(\mathbf{d};I)$ is the I-projection of $p(\mathbf{d};I)$ onto this linear family.
From (7), (11) and (12), we can also conclude that $p_{APP}(\mathbf{c}) = D_{\mathscr{C}}(p(\mathbf{c};I))$ where:

$$D_{\mathscr{C}} : \mathscr{E}_{\mathscr{P}} \to \mathscr{E}_{\mathscr{P}} : \mathbf{q} \mapsto \Pi^r_{\mathscr{E}_{\mathscr{P}}}[\Pi_{\mathscr{L}_{\mathscr{C}}}(\mathbf{q})]$$

The extrinsic information is obtained by the point-wise division of the APP at the output of each block with the input of the same block. The geometric interpretation of the iterative decoding is summarized in fig. 3. Without extrinsic propaga-

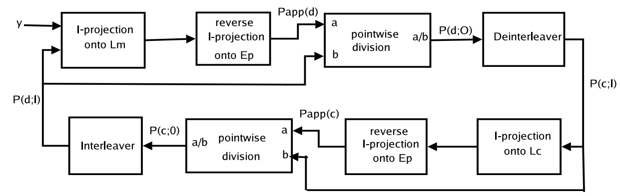

Figure 3: Geometric interpretation of the iterative decoding

tion (*ie* with APP propagation), the iterative decoding would be the alternate projection algorithm of Csiszár [15] or more generally the method of Bregman retractions. The convergence of these algorithms is well established for projection onto closed convex sets. The linear families are closed convex sets. The exponential families are log-convex and are not closed [13]. Thus, the classical results on the convergence of alternate minimization can not be applied in the setting of iterative decoding so there is no guarantee that the APP provided by the demapping and decoding block will converge toward the same limit. Dykstra's algorithm is a well-known algorithm with stronger properties than the alternate projection procedure. In the following section, we prove that the propagation of extrinsic rather than APP is Dykstra's algorithm with I-projections and reverse I-projections.

3. MAIN RESULT

3.1 Dykstra's algorithm with I-projections

Dykstra's algorithm respectively the method of cyclic projections are often employed to solve best approximation respectively convex feasibility problems. The Dykstra's algorithm is an iterative procedure which (asymptotically) finds the nearest point of any given point onto the intersection of a family of closed convex sets (best approximation problem). In the method of cyclic projections, the output of the previous projection is delivered to the next projection whereas, in the Dykstra's algorithm, only a deflected version of the previous output is given as an input to the next projection. The algorithm was first proposed and analysed by Dykstra in 1983 for orthogonal projection onto closed convex sets. This work was extended to I-projections in 1985 [9]. Here, we focus on the Dykstra's algorithm with I-projections [9].
Let $\Pi_{\mathscr{C}_i}$ stand for the I-projection onto $\mathscr{C}_i$. Here $i \in 1,2,3,4$ (as in the iterative decoding) however the procedure is valid for t projections with t finite. In the following, the interleaver/deinterleaver is omitted. Actually, these operators realize permutations of the bit indexes. As far as Kullback-Leibler minimizations are concerned, these

permutations have no insights on the result of the projection. All the products and divisions are point-wise operators: $\mathbf{u} = (\mathbf{pq})/\mathbf{r}$ stands for $\mathbf{u}(k) = (\mathbf{p}(k)\mathbf{q}(k))/\mathbf{r}(k) \forall k$.

Dykstra's algorithm

- *Initialization*
  Let $s_{1,1} = r$ and let $p_{1,1} = \Pi_{\mathscr{C}_1}(s_{1,1})$.
  Let $s_{1,2} = p_{1,1} = r(p_{1,1}/s_{1,1})$. We note that if $s_{1,1}(k) = 0$, then so is $p_{1,1}(k)$. We take $0/0$ to be 1. Set $p_{1,2} = \Pi_{\mathscr{C}_2}(s_{1,2})$.
  Let $s_{1,3} = p_{1,2}$ and set $p_{1,3} = \Pi_{\mathscr{C}_3}(s_{1,3})$.
  Let $s_{1,4} = p_{1,3}$ and set $p_{1,4} = \Pi_{\mathscr{C}_4}(s_{1,4})$.

- *Iteration n*
  Let $s_{n,1} = p_{n-1,4}/(p_{n-1,1}/s_{n-1,1})$ and let $p_{n,1} = \Pi_{\mathscr{C}_1}(s_{n,1})$.
  For i=2,3,4 Let $s_{n,i} = p_{n,i-1}/(p_{n-1,i}/s_{n-1,i})$ and let $p_{n,i} = \Pi_{\mathscr{C}_i}(s_{n,i})$.

For closed convex set, this procedure converges towards $u \in \mathscr{C} = \cap_1^4 \mathscr{C}_i$ and $D(u \parallel r) = \arg\min_{p \in \mathscr{C}} D(p \parallel r)$ *ie* the procedure converges towards the closest point to r in $\mathscr{C}$. Note that, for closed convex sets, the classical alternating minimization procedure will also converge to a point in $\cap_1^4 \mathscr{C}_i$, not necessarily the closest point to r in $\cap_1^4 \mathscr{C}_i$. For closed convex sets, Dykstra's algorithm solves the best approximation problem whereas the alternating minimization procedure solves convex feasibility problems. In that sense, Dykstra's algorithm exhibits stronger properties than the alternating minimization procedure.

3.2 Linking Dykstra's algorithm and iterative decoding

For comparison, we provide below the iterative procedure commonly used in iterative decoding.

Iterative decoding

- *Initialization*
  Let $v_{1,1} = (1/2^{L_c}...1/2^{L_c})$ and set $p_{1,1} = \Pi_{\mathscr{L}_\mathscr{M}}(v_{1,1})$.
  Let $v_{1,2} = p_{1,1}$ and set $p_{1,2} = \Pi^r_{\mathscr{E}_\mathscr{P}}(p_{1,1})$.
  Let $v_{1,3} = p_{1,2}/v_{1,1}$ and set $p_{1,3} = \Pi_{\mathscr{L}_\mathscr{C}}(v_{1,3})$.
  Let $v_{1,4} = p_{1,3}$ and set $p_{1,4} = \Pi^r_{\mathscr{E}_\mathscr{P}}(p_{1,3})$.

- *Iteration n*
  Let $v_{n,1} = p_{n-1,4}/v_{n-1,3}$ and set $p_{n,1} = \Pi_{\mathscr{L}_\mathscr{M}}(v_{n,1})$.
  Let $v_{n,2} = p_{n,1}$ and set $p_{n,2} = \Pi^r_{\mathscr{E}_\mathscr{P}}(p_{n,1})$.
  Let $v_{n,3} = p_{n,2}/v_{n,1}$ and let $p_{n,3} = \Pi_{\mathscr{L}_\mathscr{C}}(v_{n,3})$.
  Let $v_{n,4} = p_{n,3}$ and set $p_{n,4} = \Pi^r_{\mathscr{E}_\mathscr{P}}(p_{n,3})$.

Even if, at first glance, the two procedure seems slightly different, they produce exactly the **same sequence of projected distributions onto** $\mathscr{E}_\mathscr{P}$ *ie*, they produce the same sequences $\{p_{n,2}\}$ and $\{p_{n,4}\}$ as stated in theorem 3. Note that, in the iterative decoding, $\{p_{n,2}\}$ and $\{p_{n,4}\}$ are the APP at the output of each sub-block which are intended to converge towards the same solution p^*. The hard decisions rely on p^*. Thus $\{p_{n,2}\}$ and $\{p_{n,4}\}$ are of particular importance in our setting.

Theorem 3 *Iterative decoding and the Dykstra's algorithm with* $r = (1/2^{L_c}...1/2^{L_c})$, $\Pi_{\mathscr{C}_1} = \Pi_{\mathscr{L}_\mathscr{M}}$, $\Pi_{\mathscr{C}_3} = \Pi_{\mathscr{L}_\mathscr{C}}$ *and* $\Pi_{\mathscr{C}_2} = \Pi_{\mathscr{C}_4} = \Pi^r_{\mathscr{E}_\mathscr{P}}$ *lead to the same sequence of projected distributions* $\{p_{n,2}\}$ *and* $\{p_{n,4}\}$.

Proof:

- *Initialization*. By definition, $p_{1,1}$, $p_{1,2}$ and $p_{1,4}$ are the same in the two procedures. In the iterative decoding, $p_{1,3} = \Pi_{\mathscr{L}_\mathscr{C}}(v_{1,3}) = \Pi_{\mathscr{L}_\mathscr{C}}(p_{1,2}/v_{1,1})$. Since $v_{1,1}(k) = 1/2^{L_c}, \forall k \in \{1,...,2^{L_c}\}$ then $p_{1,3} = \Pi_{\mathscr{L}_\mathscr{C}}(p_{1,2})$ which is the definition of $p_{1,3}$ in the Dykstra's algorithm.
- *Iteration n*. We prove here that $s_{n,2}$ and $v_{n,2}$ are proportionals to:

$$\frac{p_{n-1,4}p_{n-2,4}\cdots p_{1,4}}{p_{n-1,2}p_{n-2,2}\cdots p_{1,2}} exp(-\frac{\parallel \mathbf{y} - \mathbf{s}(\mathbf{d}) \parallel^2}{2\sigma_c^2}) \tag{13}$$

and that $s_{n,4}$ and $v_{n,4}$ are proportional to:

$$\frac{p_{n,2}p_{n-1,2}\cdots p_{1,2}}{p_{n-1,4}p_{n-2,4}\cdots p_{1,4}} I_\mathscr{C}(\mathbf{c}) \tag{14}$$

For $n = 2$ case,

$$\begin{aligned} s_{2,2} &= \frac{p_{2,1}p_{1,1}}{p_{1,2}} \\ &= r\frac{p_{1,1}}{p_{1,2}}\frac{p_{1,4}}{p_{1,1}} exp(-\frac{\|\mathbf{y}-\mathbf{s}(\mathbf{d})\|^2}{2\sigma_c^2}) \\ &= r\frac{p_{1,4}}{p_{1,2}} exp(-\frac{\|\mathbf{y}-\mathbf{s}(\mathbf{d})\|^2}{2\sigma_c^2}) \end{aligned}$$

For the iterative decoding, $v_{2,2} = \frac{p_{1,4}.r.}{p_{1,2}} exp(-\frac{\|\mathbf{y}-\mathbf{s}(\mathbf{d})\|^2}{2\sigma_c^2})$. Thus the property is true for the $n = 2$ case. Moreover $\Pi^r_{\mathscr{E}_\mathscr{P}}(v_{2,2}) = \Pi^r_{\mathscr{E}_\mathscr{P}}(s_{2,2}) = p_{2,2}$. In the same way, we have $v_{2,4} = \frac{p_{2,2}p_{1,2}}{p_{1,4}.r} I_\mathscr{C}(\mathbf{c})$ and $s_{2,4} = \frac{p_{2,3}s_{1,4}}{p_{1,4}} = \frac{p_{2,2}p_{1,2}}{p_{1,3}}\frac{s_{1,4}}{p_{1,4}} I_\mathscr{C}(\mathbf{c}) = \frac{p_{2,2}p_{1,2}}{p_{1,4}} I_\mathscr{C}(\mathbf{c})$. So, $\Pi^r_{\mathscr{E}_\mathscr{P}}(v_{2,4}) = \Pi^r_{\mathscr{E}_\mathscr{P}}(s_{2,4}) = p_{2,4}$.
We suppose now that the proof is true at iteration $n-1$ and we prove it at iteration n. We have $s_{n,2} = \frac{p_{n,1}s_{n-1,2}}{p_{n-1,2}}$. Since $p_{n,1}$ is the projection of $s_{n,1}$ onto $\mathscr{L}_\mathscr{M}$, $p_{n,1} \propto s_{n,1} exp(-\frac{\|\mathbf{y}-\mathbf{s}(\mathbf{d})\|^2}{2\sigma_c^2})$. We also have $s_{n,1} = \frac{p_{n-1,4}s_{n-1,1}}{p_{n-1,1}}$. So $s_{n,2} \propto \frac{s_{n-1,2}p_{n-1,4}s_{n-1,1}}{p_{n-1,2}p_{n-1,1}} exp(-\frac{\|\mathbf{y}-\mathbf{s}(\mathbf{d})\|^2}{2\sigma_c^2})$ which can be simplified as $s_{n,2} \propto \frac{s_{n-1,2}p_{n-1,4}}{p_{n-1,2}}$ since $p_{n-1,1}$ is the projection of $s_{n-1,1}$ onto $\mathscr{L}_\mathscr{M}$. This proves that $s_{n,2}$ is proportional to the expression in (13).
For the iterative decoding, $v_{n,2} \propto v_{n,1} exp(-\mu f_{\mathscr{L}_\mathscr{M}}(\mathbf{d}))$ (where μ is a normalization constant and $f_{\mathscr{L}_\mathscr{M}}$ is the function related with $\mathscr{L}_\mathscr{M}$) or equivalently $v_{n,2} \propto \frac{p_{n-1,4}}{v_{n-1,3}} exp(-\frac{\|\mathbf{y}-\mathbf{s}(\mathbf{d})\|^2}{2\sigma_c^2})$ thus $v_{n,2} \propto \frac{p_{n-1,4}v_{n-1,1}}{p_{n-1,2}} exp(-\frac{\|\mathbf{y}-\mathbf{s}(\mathbf{d})\|^2}{2\sigma_c^2}) = \frac{p_{n-1,4}v_{n-1,2}}{p_{n-1,2}}$ which proves that $s_{n,2}$ is proportional to the expression in (13).
So we have, $\Pi^r_{\mathscr{E}_\mathscr{P}}(v_{n,2}) = \Pi^r_{\mathscr{E}_\mathscr{P}}(s_{n,2}) = p_{n,2}$.

In the same way, we have $s_{n,4} = \frac{p_{n,3}s_{n-1,4}}{p_{n-1,4}}$. Since $p_{n,3}$ is the projection of $s_{n,3}$ onto $\mathscr{L}_\mathscr{C}$, we get $s_{n,4} \propto \frac{s_{n,3}s_{n-1,4}}{p_{n-1,4}} I_\mathscr{C}(\mathbf{c})$. Using the definition of $s_{n,3}$ we obtain $s_{n,4} \propto \frac{p_{n,2}s_{n-1,3}s_{n-1,4}}{p_{n-1,3}p_{n-1,4}} I_\mathscr{C}(\mathbf{c})$ which is equivalent to $s_{n,4} \propto \frac{p_{n,2}s_{n-1,4}}{p_{n-1,4}} I_\mathscr{C}(\mathbf{c})$. So, $s_{n,4}$ is proportional to the expression in (14).

For the iterative decoding, $v_{n,4} \propto v_{n,3} I_{\mathscr{C}}(\mathbf{c})$. The definitions of $v_{n,3}$ and $v_{n,1}$ leads to $v_{n,4} \propto \frac{p_{n,2}}{v_{n,1}} I_{\mathscr{C}}(\mathbf{c}) = \frac{p_{n,2} v_{n-1,3}}{p_{n-1,4}} I_{\mathscr{C}}(\mathbf{c})$. Since $v_{n-1,4} = p_{n-1,3}$ is the projection of $v_{n-1,3}$ onto $\mathscr{L}_{\mathscr{C}}$, we finally obtain $v_{n,4} \propto \frac{p_{n,2} v_{n-1,4}}{p_{n-1,4}}$ which proves that $v_{n,4}$ is proportional to the expression in (14). So the projections of $v_{n,4}$ and $s_{n,4}$ onto $\mathscr{E}_{\mathscr{P}}$ are the same.

□

In the original version of the Dykstra's algorithm, all the projections are I-projections onto (non-varying) closed convex sets. In the iterative decoding, I-projections onto closed convex sets are involved as long as reverse I projections onto log-convex sets. Thus, the convergence results can not be extended straightforwardly to iterative decoding. In particular, there is no guarantee that the iterative decoding converges towards the closest point to $r = (1/2^{L_c} \dots 1/2^{L_c})$ in the set $\mathscr{L}_{\mathscr{C}} \cap \mathscr{L}_{\mathscr{M}} \cap \mathscr{E}_{\mathscr{P}}$. However, based on the duality between projections onto linear and exponential families, the reverse I-projection onto the set of separable densities is equivalent to an I-projection onto a particular (varying) linear family (see [16]). Thus, iterative decoding can also be written with I-projections onto closed convex sets. The difference with the classical Dykstra's algorithm is limited to the "varying" nature of the linear family involved in the minimization process. A recent work gives some elements to tackle this problem. Indeed, Niesen and *al.* proposed in [17] a generalization of the alternating minimization procedure of Csiszar to the case of projections onto time-varying sets. This proof combined with the convergence results in [9] seems a promising direction of investigation for the derivation of new convergence results for iterative decoding.

4. CONCLUSION

In this paper, we have presented some tools and concepts of information geometry that apply for the description of iterative receivers. The extrinsic propagation is very similar to the deflected output propagation used in Dystra's algorithm. This similarity suggests that convergence results might be developed via this analogy.

REFERENCES

[1] E. Zehavi. 8-PSK trellis codes for a Rayleigh fading channel. *IEEE Trans. Commun.*, 40:873–883, May 1992.

[2] G. Caire, G.Taricco, and E. Biglieri. Bit-interleaved coded modulation. *IEEE Trans. on Inf. Theory*, 4:927–946, May 1998.

[3] S. ten Brink. Convergence behavior of iteratively decoded parallel concatenated codes. *IEEE trans Commun.*, 49:1727–1737, Oct 2001.

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

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

[6] J. M. Walsh, P.A. Regalia, and C. R. Johnson. Turbo decoding as Iterative Constrained Maximum-Likelihood Sequence Detection. *IEEE Trans. on Inf. Theory*, 52:5426–5437, Dec. 2006.

[7] B. Muquet, P. Duhamel, and M. de Courville. A geometrical interpretation of iterative turbo decoding. In *Proc. Int. Symposium on Information Theory*, Lausanne, Switzerland, May 2002.

[8] T. Richardson. The geometry of turbo-decoding dynamics. *IEEE Trans. on Information Theory*, 46(1):9–23, 2000.

[9] R.L. Dykstra. An iterative procedure for obtaining I-projections onto the intersection of convex sets. *The annals of Probability*, 13(3):975–984, 1985.

[10] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon limit error-correcting coding and decoding: Turbo codes. In *Proc. IEEE Int. Conf. Commun*, pages 1064–1070, 1993.

[11] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara. A soft-input soft-output APP module for iterative decoding of concatenated codes. *IEEE Commun. Letters*, 1:22–24, Jan 1997.

[12] I. Csiszár and F. Matús. Information projections revisited. *IEEE Trans. on Inform Theory*, 49(6):1474–1489, June 2003.

[13] I. Csiszár and P. Schields. Information theory and statistics: A tutorial. In *Foundations and Trends in Communications and Information Theory*. Now Publishers Inc., 2004.

[14] M. Moher and T.A. Gulliver. Cross-entropy and iterative decoding. *IEEE Trans. On Inform. Theory*, 44(7):3097–3104, Nov. 1998.

[15] I. Csiszár and G. Tusnády. Information geometry and alternating minimization procedure. *Statistics & Decisions*, supplement issue 1:205–237, 1984.

[16] J.N. Darroch and D. Ratcliff. Generalized Iterative Scaling for Log-Linear Models. *The annals of Mathematical Statistics*, 43(5):1470–1480, 1972.

[17] U. Niesen, D. Shah, and G. Wornell. Adaptive Alternating Minimization Algorithms. *submitted to IEEE Trans. on Inform. Theory*, 2007.