A REFINED INFORMATION GEOMETRIC INTERPRETATION OF TURBO DECODING

*John M. Walsh, C. Richard Johnson, Jr.**

Cornell University
School of Electrical and Computer Engineering
Ithaca, NY 14853

Phillip A. Regalia †

The Catholic University of America
Department of Electrical Engineering
Washington, DC 20064

ABSTRACT

Many previous attempts at analyzing the convergence behavior of turbo and iterative decoding, such as EXIT style analysis [1] and density evolution [2], ultimately appeal to results which become valid only when the block length grows rather large, while still other attempts, such as connections to factor graphs [3] and belief propagation [4], have been largely unsuccessful at showing convergence due to loops in the turbo coding graph. The information geometric interpretation presented in this paper, which builds upon the results of [5], [6], and [7], allows us to relate the quantities of interest in the turbo decoder. Using it, we point out a measure which will be key in studying convergence.

1. INTRODUCTION

Along with being one of the most prominent communications inventions of the past decade, the introduction of turbo codes in [8] began a new era in communications systems which brought them closer than ever to theoretical performance limits. The creation of turbo codes introduced a new method of decoding these codes which brought the decoding of complex codes within the reach of computationally practical algorithms. The iterative decoding algorithm, while being suboptimal, performs well enough to bring turbo codes very close to theoretically attainable limits. Yet, an accurate justification for why the decoding strategy performs as well as it does is still lacking. Significant progress has been made with EXIT style analysis [1] and density evolution [2], but these techniques appeal to approximations which are only valid in the case of very large block sizes. Connections were shown to the sum product and belief propagation algorithms in [3] and [4], but these algorithms are only known to converge when the code graph has no loops, which is rarely true for turbo codes. Indeed, many questions remain concerning where the fixed points of the iterative decoding algorithm lie and under which conditions the algorithm converges.

In order to gain some insight into the iterative decoding algorithm, we will follow in the footsteps of [5], [6], and [7] and develop a information geometric interpretation of turbo decoding. Our approach differs in several ways from [6] and [7], because we will consider the algorithm as four projections rather than two, and we will avoid working in the log domain. We will also be able to provide a couple of new results complementing [9] as to the form of the solutions to the projections and the properties which they satisfy in our context.

In order to make the exposition as clear as possible, we will review the relevant information geometric concepts in Section 2, giving examples along the way, before studying information projections in Section 3, and finally stating the information projection form of the turbo decoding algorithm in Section 4. At that point we state the equations we know relating the quantities of interest to the turbo decoding algorithm, and mention a quantity that is key to convergence in this formulation.

2. INFORMATION GEOMETRY OF PMFS

In our investigation we will be interested in the set $\mathscr{F}$ of all probability mass functions (PMFs) on a binary word, $\mathbf{x} \in \mathcal{B}_N$, of length N, where $\mathbf{x}$ is a length N vector whose entries are 0 or 1 (bits) and $\mathcal{B}_N$ is the set of all such $\mathbf{x}$s. We will consider coordinate systems for $\mathscr{F}$, which will be a bijective mapping between a subset, $\mathcal{D}$, of the extended Euclidean space $\mathbb{R}^N \bigcup \{\pm\infty\}$ and the set of all possible PMFs $\mathscr{F}$. It turns out (because discrete probability densities can be viewed as exponential densities [6]) that there are two natural coordinate systems for the set of all PMFs on $\mathcal{B}_N$ [10].

The first possible coordinate system for $\mathscr{F}$ can be formed by first enumerating all of the points in $\mathcal{B}_N$, so we have $\mathcal{A} = \{x_0, x_1, \ldots, x_{2^N-1}\}$. Then, to represent a PMF $\mathsf{P} \in \mathscr{F}$, just use

$$\eta_i = \mathsf{P}(x_i)$$

for $i \in \{1, \ldots, 2^N - 1\}$. We will call this coordinate system the η coordinate system, and affine manifolds in these coordinates will be called m-flat manifolds or m-flat submanifolds.

Ex. 1 (Posterior PMFs for a Code [5])**:** *The set of all posterior PMFs for a given code forms an m-flat submanifold of the manifold of all PMFs for a binary observation.*

A second way to represent a PMF $\mathsf{P} \in \mathscr{F}$, is to calculate for each $i \in \{1, \ldots, 2^N - 1\}$

$$\theta_i = \log\left(\frac{\mathsf{P}(x_i)}{\mathsf{P}(x_0)}\right)$$

We shall refer to this coordinate system as the θ coordinates, and affine manifolds in these coordinates will be referred to as e-flat manifolds (or submanifolds).

---

*John Walsh and C. Richard Johnson, Jr. were supported in part by Applied Signal Technology, Texas Instruments, and NSF Grants CCF-0310023 and INT-0233127. Contact jmw56@cornell.edu for any inquiries regarding this paper.

†P. A. Regalia was supported in part by the Network of Excellence in Wireless Communications (NEWCOM), E. C. Contract no. 507325 while at the Groupe des Ecoles des Télécommunications, INT, 91011 Evry France.

Ex. 2 ($\mathcal{P}$: The Space of Product PMFs [6])**:** *The space of PMFs which can be written as a product of their marginals is an e-flat manifold. In other words, the set of factorizable PMFs is an affine manifold in the θ coordinates.*

Proof: Define the matrix $\mathbf{B} = \left[\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_i, \cdots, \mathbf{b}_{2^N-1}\right]^T$ where $\mathbf{b}_i$ is the vector binary representation of i. For any PMF z on $\mathcal{B}_N$ define θ_z to be the vector of θ coordinates of z. Then,

$$\mathcal{P} = \left\{ \mathsf{z} | \mathbf{F} \left[\begin{array}{c} 0 \\ \theta_\mathsf{z} \end{array} \right] = 0 \right\} \tag{1}$$

where $\mathbf{F}$'s columns are a basis for null$(\mathbf{B}^T)$. ■

3. INFORMATION PROJECTIONS

In this section we will consider two types of information projections onto sets of PMFs. Here, because we are dealing with discrete densities, the relations are less complex that those encountered in [9]. Also, we will not concern ourselves about questions of existence and uniqueness of solutions here, since these are addressed in [9], which proved them for the context we are working with. In general, a projection will be finding a point in some set of PMFs which minimizes the Kullback Leibler distance between itself and another given PMF. Because the Kullback Leibler distance is not symmetric, there are two types of projections, m-projections and e-projections. We discuss these below.

3.1. m-Projections

Consider the following optimization problem. Given a PMF $\mathsf{r}(\mathbf{x})$, find the PMF, $\mathsf{p}(\mathbf{x})$, which minimizes the relative entropy among the set of PMFs satisfying the constraints[1]

$$\mathsf{p} \in \mathcal{H} = \left\{ \mathsf{q} | \sum_{\mathbf{x}} \mathsf{q}(\mathbf{x}) = 1, \sum_{\mathbf{x}} \mathsf{h}_i(\mathbf{x}) \log(\mathsf{q}(\mathbf{x})) = \beta_i \right\} \tag{2}$$

that is

$$\mathsf{p} = \arg\min_{\mathsf{q} \in \mathcal{H}} \mathsf{D}(\mathsf{r}||\mathsf{q}) \tag{3}$$

The existence and uniqueness of a solution to such an optimization problem is discussed in [9]. We have independently developed the following result

Prop. 1: *The solution p to the optimization problem above takes the form*

$$\mathsf{p}(\mathbf{x}) = \mathsf{r}(\mathbf{x}) + \sum_i e_i \mathsf{h}_i(\mathbf{x}) \tag{4}$$

where the constants e_i are chosen to satisfy the constraints in (2). Also

$$\mathsf{D}(\mathsf{r}||\mathsf{p}) = \mathsf{H}(\mathsf{p}) - \mathsf{H}(\mathsf{r}) + \sum_i e_i \beta_i \tag{5}$$

where $\mathsf{H}(\mathsf{p}) = \sum_\mathbf{x} \mathsf{p}(\mathbf{x}) \log(\mathsf{p}(\mathbf{x}))$ is the entropy of p. Furthermore, for any other PMF q in the constraint set $\mathcal{H}$ we have

$$\mathsf{D}(\mathsf{r}||\mathsf{q}) = \mathsf{D}(\mathsf{r}||\mathsf{p}) + \mathsf{D}(\mathsf{p}||\mathsf{q}) \tag{6}$$

[1] Note that, in the language of [6], [10], these constraints form an e-flat submanifold of the manifold of all PMFs.

Proof: To see this, consider

$$\mathsf{D}(\mathsf{r}||\mathsf{p}) = \sum_{\mathbf{x}} \mathsf{r}(\mathbf{x}) \log(\mathsf{r}(\mathbf{x})) - \sum_{\mathbf{x}} \mathsf{r}(\mathbf{x}) \log(\mathsf{p}(\mathbf{x})) \tag{7}$$

and

$$\begin{aligned} \mathsf{D}(\mathsf{r}||\mathsf{q}) &= \sum_{\mathbf{x}} \mathsf{r}(\mathbf{x}) \log\left(\frac{\mathsf{r}(\mathbf{x})\mathsf{p}(\mathbf{x})}{\mathsf{q}(\mathbf{x})\mathsf{p}(\mathbf{x})}\right) \\ &= \sum_{\mathbf{x}} \mathsf{r}(\mathbf{x}) \log\left(\frac{\mathsf{r}(\mathbf{x})}{\mathsf{p}(\mathbf{x})}\right) + \sum_{\mathbf{x}} \mathsf{r}(\mathbf{x}) \log\left(\frac{\mathsf{p}(\mathbf{x})}{\mathsf{q}(\mathbf{x})}\right) \\ &= \mathsf{D}(\mathsf{r}||\mathsf{p}) + \sum_{\mathbf{x}} \mathsf{r}(\mathbf{x}) \log\left(\frac{\mathsf{p}(\mathbf{x})}{\mathsf{q}(\mathbf{x})}\right) \end{aligned} \tag{8}$$

Substituting in $\mathsf{r}(\mathbf{x}) = \mathsf{p}(\mathbf{x}) - \sum_i e_i \mathsf{h}_i(\mathbf{x})$ in (7) and (8), we have

$$\mathsf{D}(\mathsf{r}||\mathsf{p}) = \mathsf{H}(\mathsf{p}) - \mathsf{H}(\mathsf{r}) + \sum_i e_i \sum_{\mathbf{x}} \mathsf{h}_i(\mathbf{x}) \log(\mathsf{p}(\mathbf{x}))$$

which gives (5), and

$$\begin{aligned} \mathsf{D}(\mathsf{r}||\mathsf{q}) &= \mathsf{D}(\mathsf{r}||\mathsf{p}) + \sum_{\mathbf{x}} \mathsf{p}(\mathbf{x}) \log\left(\frac{\mathsf{p}(\mathbf{x})}{\mathsf{q}(\mathbf{x})}\right) \\ &\quad + \sum_i \sum_{\mathbf{x}} e_i \mathsf{h}_i(\mathbf{x}) \log(\frac{\mathsf{p}(\mathbf{x})}{\mathsf{q}(\mathbf{x})}) \\ &= \mathsf{D}(\mathsf{r}||\mathsf{p}) + \mathsf{D}(\mathsf{p}||\mathsf{q}) + \sum_i (e_i\beta_i - e_i\beta_i) \\ &= \mathsf{D}(\mathsf{r}||\mathsf{p}) + \mathsf{D}(\mathsf{p}||\mathsf{q}) \end{aligned}$$

This shows that if there is a density of the form (4) which satisfies the constraints in (2) then it solves the minimization problem (3) and satisfies (6). ■

Ex. 3 (Maginalization)**:** *Finding the nearest factorizable PMF in terms of relative entropy to a given PMF is an m-projection. Furthermore the density that results has the same bitwise marginal probabilities as the original density. [6]*

Proof: Defining, as before, $\mathbf{B}$ and a basis for the null space of $\mathbf{B}$, to be $\mathbf{F} = \text{null}(\mathbf{B}^T)$ let $\mathsf{f}_i(\mathbf{x}_j) = \mathbf{F}[i,j] \forall i, j$. If we define $\mathsf{h}_i(\mathbf{x}_0) = \mathsf{f}_i(\mathbf{x}_0) - \sum_x \mathsf{f}_i(\mathbf{x})$ and $\mathsf{h}_i(\mathbf{x}) = \mathsf{f}_i(\mathbf{x}) \forall \mathbf{x} \neq \mathbf{x}_0$, then, using (1) our constraints are

$$\mathcal{P} = \{\mathsf{z}| \sum_{\mathbf{x}} \mathsf{h}_i(\mathbf{x}) \log(\mathsf{z}(\mathbf{x})) = 0 \; \forall i \in \{1, \ldots, N-1\}\}$$

which is the form desired. To see second half of the statement, recall that the relative entropy between a point with θ coordinates θ_r and a point in $\mathcal{P}$ which will have θ coordinates $\mathbf{B}\lambda$ [7]

$$\begin{aligned} \mathsf{D}_\theta(\theta_r||\mathbf{B}\lambda) &= < \frac{\exp(\theta_r)}{\|\exp(\theta_r)\|_1}, \theta_r - \mathbf{1}\log(\|\exp(\theta_r)\|_1) \\ &\quad -\mathbf{B}\lambda + \mathbf{1}\log(\|\exp(\mathbf{B}\lambda)\|_1) > \end{aligned}$$

where we added a zero on top of θ_r so that it would have the appropriate dimension. Taking the derivative with respect to λ and setting equal to zero yields

$$\mathbf{B}^T \frac{\exp(\mathbf{B}\lambda)}{\|\exp(\mathbf{B}\lambda)\|_1} - \mathbf{B}^T \frac{\exp(\theta_r)}{\|\exp(\theta_r)\|_1} = 0$$

and since multiplying the η coordinates by $\mathbf{B}^T$ yields the marginals [7], we see that at the minimum relative entropy, the marginals of the densities θ_r and $\mathbf{B}\lambda$ must be equal.■

3.2. e-Projections

Consider the following optimization problem, whose relevance, formulation, and solution were noted in [5]. Given a PMF $\mathsf{r}(\mathbf{x})$, find the PMF, $\mathsf{p}(\mathbf{x})$, which minimizes the relative entropy among the set of PMFs satisfying the following constraints

$$\mathsf{p} \in \mathcal{H} = \left\{ \mathsf{q} \middle| \sum_{\mathbf{x}} \mathsf{f}_i(\mathbf{x})\mathsf{q}(\mathbf{x}) = \mu_i \quad i \in \{0, 1, \ldots, N-1\} \right\} \tag{9}$$

That is, find

$$\mathsf{p} = \arg\min_{\mathsf{q}\in\mathcal{H}} \mathsf{D}(\mathsf{q}||\mathsf{r}) \tag{10}$$

The existence and uniqueness of a solution to such an optimization problem is discussed in [9]. We have the following result

Prop. 2: *The solution* p *to the optimization problem above takes the form*

$$\mathsf{p}(\mathbf{x}) = \mathsf{r}(\mathbf{x}) \exp\left(c_0 + \sum_i c_i \mathsf{f}_i(\mathbf{x})\right) \tag{11}$$

where the constants c_i *are chosen to satisfy the constraints. Furthermore, any other PMF* q *in the constraint set satisfies*

$$\mathsf{D}(\mathsf{q}||\mathsf{r}) = \mathsf{D}(\mathsf{p}||\mathsf{r}) + \mathsf{D}(\mathsf{q}||\mathsf{p}) \tag{12}$$

Proof: First of all, note that

$$\begin{aligned} \mathsf{D}(\mathsf{p}||\mathsf{r}) &= \sum_{\mathbf{x}} \mathsf{p}(\mathbf{x}) \log\left(\frac{\mathsf{r}(\mathbf{x})\exp\left(c_0 + \sum_i c_i \mathsf{f}_i(\mathbf{x})\right)}{\mathsf{r}(\mathbf{x})}\right) \\ &= \sum_{\mathbf{x}} \mathsf{p}(\mathbf{x}) \left(c_0 + \sum_i c_i \mathsf{f}_i(\mathbf{x})\right) = c_0 + \sum_i c_i \mu_i \end{aligned}$$

$$\begin{aligned} \mathsf{D}(\mathsf{q}||\mathsf{r}) &= \sum_{\mathbf{x}} \mathsf{q}(\mathbf{x}) \log\left(\frac{\mathsf{q}(\mathbf{x})\mathsf{p}(\mathbf{x})}{\mathsf{r}(\mathbf{x})\mathsf{p}(\mathbf{x})}\right) \\ &= \mathsf{D}(\mathsf{q}||\mathsf{p}) + \sum_{\mathbf{x}} \mathsf{q}(\mathbf{x}) \log(\exp(c_0 + \sum_i c_i \mathsf{f}_i(\mathbf{x}))) \\ &= \mathsf{D}(\mathsf{q}||\mathsf{p}) + c_0 + \sum_i c_i\mu_i = \mathsf{D}(\mathsf{q}||\mathsf{p}) + \mathsf{D}(\mathsf{p}||\mathsf{r}) \end{aligned}$$

This shows that if there exists a density of the form in (11) which satisfies the constraints in (9) then it solves the optimization problem in (10) and satisfies (12). ■

4. INFORMATION PROJECTION INTERPRETATION OF TURBO DECODING

We describe the turbo decoding algorithm in terms of its projections and its intrinsic information extractions. Here, we will be considering turbo codes created via parallel concatenation, as in Fig. 1. A typical turbo decoding implementation is shown in Fig. 2. Inside the box labelled *BCJR*, the bitwise MAP soft decoding is done for one of the component decoders using the probabilities associated with the systematic bits and the parity check bits for that decoder. The box labelled *Extract* extracts the intrinsic information for the systematic bits, and the *Interleave* block interleaves the likelihood information for the bits.

In what follows, $\mathcal{C}_0$ are the constraints for the first component code, $\mathcal{C}_1$ are the constraints for the second component code, and $\mathcal{P}$

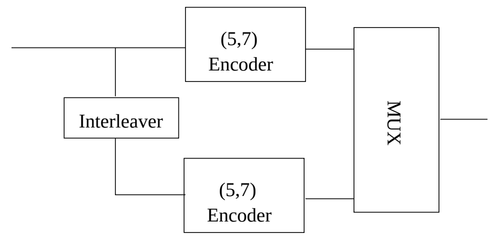

Fig. 1. A parallel concatenated turbo code. The MUX selects both the systematic and parity check bits from one of the component codes and just the parity check bits of the other. If puncturing is used, some of the parity check bits are never transmitted.

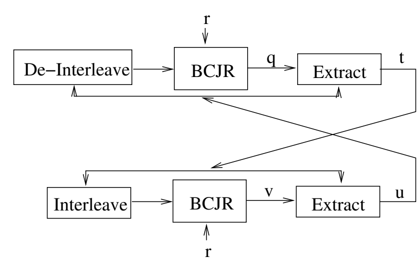

Fig. 2. The turbo decoder.

is the space of all product densities. We have already shown that $\mathcal{P}$ can be written as an affine manifold in the θ coordinates (i.e. an e-flat submanifold). $\mathcal{C}_0$ and $\mathcal{C}_1$ are developed from the parity check equations for the two component codes, [5], and can both be written as affine manifolds in the η coordinates (i.e. as m-flat submanifolds) by way of example 1.

To be specific, for any possible input message $\mathbf{m}$, denote the parity bits created by the first code by $\mathsf{c}_0(\mathbf{m})$ of length N_0, and denote the parity generated by the second code by $\mathsf{c}_1(\mathbf{m})$ of length N_1. This way, with attention to the output of both encoders, we can denote the codebook for the first code by $\mathcal{B}_0 = \{(\mathbf{m}, \mathsf{c}_0(\mathbf{m}), \mathbf{b}_0)\}$ and the codebook for the second code by $\mathcal{B}_1 = \{(\mathbf{m}, \mathbf{b}_1, \mathsf{c}_1(\mathbf{m}))\}$ where we were considering all $\mathbf{m} \in \{0,1\}^N$, $\mathbf{b}_i \in \{0,1\}^{N_i}$ so that the overall code book for the parallel concatenated code can be written as $\mathcal{B}_0 \bigcap \mathcal{B}_1$. Furthermore, if we define $\mathsf{f}(\mathbf{x})$ to be 0 if $\mathbf{x} \in \mathcal{B}_0$ and 1 otherwise, and $\mathsf{g}(\mathbf{x})$ to be 0 if $\mathbf{x} \in \mathcal{B}_1$ and 1 otherwise, this allows us to write the constraint sets $\mathcal{C}_0$ and $\mathcal{C}_1$ as

$$\mathcal{C}_0 = \{\mathsf{z}| \sum_{\mathbf{x}} \mathsf{f}(\mathbf{x})\mathsf{z}(\mathbf{x}) = 0\}$$

$$\mathcal{C}_1 = \{\mathsf{z}| \sum_{\mathbf{x}} \mathsf{g}(\mathbf{x})\mathsf{z}(\mathbf{x}) = 0\}$$

Let r be the bitwise conditional probabilities (irrespective of the codes) that we observe at the AWGN channel output given the input was ± 1, and let r_s be just the bitwise conditional probabilites of the systematic bits. Finally, we let $\mathcal{P}$ and h_i be defined as in examples 2 and 3 with the additional restriction that they the pmfs they contain are functions of the systematic bits only.

Given these definitions, the turbo decoding algorithm admits an exact interpretation as the iteration of the following six steps, where e_i, b_i, c_i and d_i are the constants which satisfy the constraints for the minimization. There is an implicit understanding that whenever two PMFs are multiplied or divided, there will be a constant multiplicative scale factor included (which is not shown) which ensures that the new PMF sums to 1. We also have written the equations we know for the form of the solution to each projection and the minimum distance attained at each step.

1. **e-Projection to $\mathcal{C}_0$**

$$\mathsf{p}_k = \arg\min_{\mathsf{z}\in\mathcal{C}_0} \mathsf{D}(\mathsf{z}||\mathsf{r}\mathsf{u}_k) = \mathsf{r}\mathsf{u}_k \exp\{c_0 + c_1 \mathsf{f}(\mathbf{x})\}$$

$$\mathsf{D}(\mathsf{z}||\mathsf{r}\mathsf{u}_k) = \mathsf{D}(\mathsf{p}_k||\mathsf{r}\mathsf{u}_k) + \mathsf{D}(\mathsf{z}||\mathsf{p}_k) \quad \forall \mathsf{z}\in\mathcal{C}_0$$
$$\mathsf{D}(\mathsf{p}_k||\mathsf{r}\mathsf{u}_k) = c_0$$

2. **m-Projection to $\mathcal{P}$**

$$\mathsf{q}_k = \arg\min_{\mathsf{z}\in\mathcal{P}} \mathsf{D}(\mathsf{p}_k||\mathsf{z}) = \mathsf{p}_k(\mathbf{x}) + \sum_i e_i \mathsf{h}_i(\mathbf{x})$$

$$\mathsf{D}(\mathsf{p}_k||\mathsf{z}) = \mathsf{D}(\mathsf{p}_k||\mathsf{q}_k) + \mathsf{D}(\mathsf{q}_k||\mathsf{z}) \quad \forall \mathsf{z}\in\mathcal{P}$$
$$\mathsf{D}(\mathsf{p}_k||\mathsf{q}_k) = \mathsf{H}(\mathsf{q}_k) - \mathsf{H}(\mathsf{p}_k)$$

3. **Extract Extrinsic Information**

$$\mathsf{t}_k = \mathsf{q}_k/\mathsf{r}_s\mathsf{u}_k \in \mathcal{P}$$

$$\mathsf{t}_k(\mathbf{x}) = \frac{\mathsf{r}}{\mathsf{r}_s}\exp\{c_0 + c_1\mathsf{f}(\mathbf{x})\} + \frac{\sum_i e_i \mathsf{h}_i(\mathbf{x})}{\mathsf{r}_s\mathsf{u}_k(\mathbf{x})}$$

4. **e-Projection to $\mathcal{C}_1$**

$$\mathsf{v}_k = \arg\min_{\mathsf{z}\in\mathcal{C}_1} \mathsf{D}(\mathsf{z}||\mathsf{r}\mathsf{t}_k) = \mathsf{r}\mathsf{t}_k(\mathbf{x}) \exp(d_0 + d_1 \mathsf{g}(\mathbf{x}))$$

$$\mathsf{D}(\mathsf{z}||\mathsf{r}\mathsf{t}_k) = \mathsf{D}(\mathsf{v}_k||\mathsf{r}\mathsf{t}_k) + \mathsf{D}(\mathsf{z}||\mathsf{v}_k) \quad \forall \mathsf{z}\in\mathcal{C}_1$$
$$\mathsf{D}(\mathsf{v}_k||\mathsf{r}\mathsf{t}_k) = d_0$$

5. **m-Projection to $\mathcal{P}$**

$$\mathsf{s}_k = \arg\min_{\mathsf{z}\in\mathcal{P}} \mathsf{D}(\mathsf{v}_k||\mathsf{z}) = \mathsf{v}_k + \sum_i b_i \mathsf{h}_i(\mathbf{x})$$

$$\mathsf{D}(\mathsf{v}_k||\mathsf{z}) = \mathsf{D}(\mathsf{v}_k||\mathsf{s}_k) + \mathsf{D}(\mathsf{s}_k||\mathsf{z}) \quad \forall \mathsf{z}\in\mathcal{P}$$
$$\mathsf{D}(\mathsf{v}_k||\mathsf{s}_k) = \mathsf{H}(\mathsf{s}_k) - \mathsf{H}(\mathsf{v}_k)$$

6. **Extract Extrinsic Information**

$$\mathsf{u}_{k+1} = \mathsf{s}_k/\mathsf{r}_s\mathsf{t}_k \in \mathcal{P}$$

$$\mathsf{u}_{k+1} = \frac{\mathsf{r}}{\mathsf{r}_s}\exp(d_0 + d_1\mathsf{g}(\mathbf{x})) + \frac{\sum_i b_i \mathsf{h}_i(\mathbf{x})}{\mathsf{r}_s\mathsf{t}_k(\mathbf{x})}$$

Here, steps 1 and 2 combined describe the behavior of the first component decoder, which takes the probabilistic channel observations r along with the pseudo-priors u_k, and determines the pseudo-posteriors q_k, from which the extrinsic information t_k is extracted in step 3. Similarly, steps 4 and 5 combined describe the behavior of the second component decoder, which takes the extrinsic information from the output of the first component decoder as pseudo priors and determines the bitwise pseudo posteriors s_k, from which the extrinsic information u_{k+1} is extracted, and the structure iterates. The pseudo-posteriors which the turbo decoder makes its final hard decisions based on are q_k and s_k.

Keeping in mind that the turbo decoder is said to converge when the two decoder's pseudo posteriors, q_k and s_k, agree, one possible quantity to track over time to determine the convergence of iterative decoding would be $\mathsf{D}(\mathsf{q}_k||\mathsf{s}_k)$. This information geometric interpretation ought to aid in tracking such a quantity. In fact, as shown in Fig. 3, simulations using two (5,7) encoders, a random interleaver, and $E_b/N_0 = 3.5$dB suggested that this quantity decreases montonically with time.

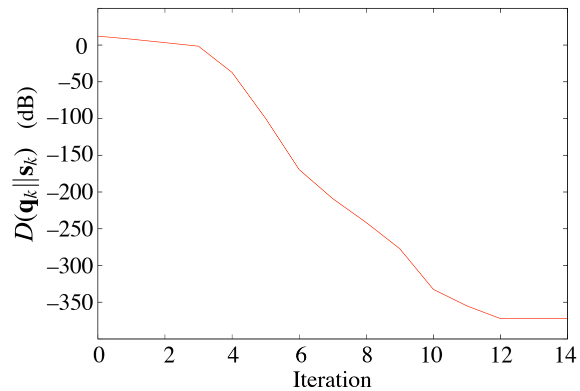

Fig. 3. Apparent monotone convergence of $\mathsf{D}(\mathsf{q}_k||\mathsf{s}_k)$ for two $(5,7)$ encoders, a random interleaver, and $E_b/N_o = 3.5$dB.

5. REFERENCES

[1] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes.," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, Oct. 2001.

[2] H. El Gamal and A. R. Hammons, Jr., "Analyzing the turbo decoder using the gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 671–686, Feb. 2001.

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

[4] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo decoding as an instance of pearls belief propagation algorithm.," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 140–152, Feb. 1998.

[5] M. Moher and T. A. Gulliver, "Cross-entropy and interative decoding.," *IEEE Trans. Inform. Theory*, vol. 44, pp. 3097–3104, Nov. 1998.

[6] S. Ikeda, T. Tanaka, and S. Amari, "Information geometry of turbo and low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 50, pp. 1097 – 1114, June 2004.

[7] T. Richardson, "The geometry of turbo-decoding dynamics.," *IEEE Trans. Inform. Theory*, vol. 46, pp. 9–23, Jan. 2000.

[8] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes.," in *ICC 93*, Geneva, May 1993, vol. 2, pp. 1064–1070.

[9] I. Csiszár and F. Matúš, "Information projections revisited," *IEEE Trans. Inform. Theory*, vol. 49, pp. 1474–1490, June 2003.

[10] S. Amari, *Methods of Information Geometry*, vol. 191, AMS Translations of Mathematical Monographs, 2004.

[11] C. Berrou and A. Glavieux, "Near optimum error correction coding and decoding: Turbo codes.," *IEEE Trans. Commun.*, vol. 44, pp. 1262–1271, Oct. 1996.