Systems and Computers in Japan, Vol. 36, No. 1, 2005
Translated from Denshi Joho Tsushin Gakkai Ronbunshi, Vol. J85-D-II, No. 5, May 2002, pp. 758–765

Information Geometry for Turbo Decoding

Shiro Ikeda,[1] Toshiyuki Tanaka,[2] and Shun-ichi Amari[3]

[1]Institute of Statistical Mathematics, Minato, 106-8569 Japan

[2]Tokyo Metropolitan University, Hachioji, 192-0397 Japan

[3]RIKEN BSI, Wako, 351-0198 Japan

SUMMARY

Turbo codes are known as a class of error-correcting codes which have high error-correcting performance with efficient decoding algorithm. Characteristics of the iterative decoding algorithm have been studied in detail through a variety of numerical experiments, but theoretical results are still insufficient. In this paper, this issue is addressed from the information geometrical viewpoint. As a result, a mathematical framework for analyzing turbo codes is obtained, and some of the fundamental properties of turbo decoding are elucidated based on this framework. Recently, it has been pointed out that the turbo decoding algorithm is related to the decoding algorithm of low-density parity check codes, the computation method of Bethe approximation in statistical physics, and the belief propagation algorithm of Bayesian networks. The mathematical framework given in the present paper can also be used to analyze these wide classes of iterative computation methods, and hence represent a new analysis tool. © 2004 Wiley Periodicals, Inc. Syst Comp Jpn, 36(1): 79–87, 2005; Published online in Wiley InterScience (www.interscience.wiley.com). DOI 10.1002/scj.10359

Contract grant sponsor: Supported in part by MEXT, Japan under a Grant-in-Aid for Scientific Research (16700227, 14084208).

Key words: turbo code; MPM decoding; information geometry.

1. Introduction

Turbo codes are a class of error-correcting codes in which an iterative algorithm is used for decoding. Since their appearance in 1993 [1], a variety of numerical experiments have shown the high performance of the practical codes. However, theoretical results [2] reported are not enough, and we need further results to understand the fundamental properties of the codes.

Similarities between turbo decoding and other methods have also been noted. Common features were found [4] to exist between the iterative algorithm of the turbo decoding and the iterative algorithm used in the decoding of low-density parity check codes [3]. It was also shown that these decoding problems could be formularized as an inference problem of a Bayesian network, and that the iterative decoding algorithms were equivalent to belief propagation for this Bayesian network [5]. Common features were also shown to exist in the computation methods of Bethe approximation in statistical physics [6]. Of course, there are still many theoretically unresolved problems regarding these methods. Therefore, the framework of these iterative methods will become clear once the mathematical structure of turbo codes is clarified.

In this paper, we first describe the turbo decoding algorithm from an information geometrical [7, 8] viewpoint, and build a mathematical framework to analyze it. The basic mathematical properties of the turbo decoding algorithm are also clarified based on this framework. The geometrical structure of turbo decoding results is elucidated, and local stability conditions of the algorithm are demonstrated. A cost function, which is important for turbo decoding, is also shown, and the decoding errors of turbo codes are discussed. The result of the present paper is also useful for the iterative algorithms of the previously described classes.

2. Information Geometry of Turbo Codes

2.1. Turbo codes

Let us consider a case in which an information data block $\boldsymbol{x} = (x_1, \ldots, x_N)^T$, $x_i \in \{-1, +1\}$ is sent over a memoryless binary symmetric channel (BSC). In this paper, a BSC is assumed for simplicity, but the framework of the information geometry obtained in the present paper can be expanded to an additive white Gaussian noise channel and other memoryless channels. Turbo codes (Fig. 1) are implemented as convolutional codes in order to increase the code lengths, but we treat them as block codes in the present paper [2, 5]. Turbo codes use two encoders to create two parity check words for a single code word. The corresponding results are $\boldsymbol{y}_1 = (y_{11}, \ldots, y_{1L})^T$, $\boldsymbol{y}_2 = (y_{21}, \ldots, y_{2L})^T$, $y_{1j}, y_{2j} \in \{-1, +1\}$. When $(\boldsymbol{x}, \boldsymbol{y}_1, \boldsymbol{y}_2)$ are transmitted over a communications channel, they are received as $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$, where $\tilde{x}_i, \tilde{y}_{1j}, \tilde{y}_{2j} \in \{-1, +1\}$. Also, $\boldsymbol{y}_r$, $r = 1, 2$, is a function of $\boldsymbol{x}$, and is expressed as $\boldsymbol{y}_r(\boldsymbol{x})$ when needed. Based on the received words, the original information word is inferred as $\hat{\boldsymbol{x}}$.

A turbo-decoding algorithm will first be described. Turbo decoding is a type of decoding in which two decoders are used in an alternating manner. The probability distributions $p(\tilde{\boldsymbol{x}}|\boldsymbol{x})$, $p(\tilde{\boldsymbol{y}}_r|\boldsymbol{x})$, $r = 1, 2$ and the variables shown below are defined, as is function F.

$$
\begin{aligned}
lx_i &\stackrel{\text{def}}{=} \ln \frac{\sum_{\{\boldsymbol{x}:x_i=+1\}} p(\tilde{\boldsymbol{x}}|\boldsymbol{x})}{\sum_{\{\boldsymbol{x}:x_i=-1\}} p(\tilde{\boldsymbol{x}}|\boldsymbol{x})} \\
&= \ln \frac{p(\tilde{x}_i|x_i=+1)}{p(\tilde{x}_i|x_i=-1)}, \\
ly_{rj} &\stackrel{\text{def}}{=} \ln \frac{\sum_{\{\boldsymbol{x}:y_{rj}=+1\}} p(\tilde{\boldsymbol{y}}_r|\boldsymbol{x})}{\sum_{\{\boldsymbol{x}:y_{rj}=-1\}} p(\tilde{\boldsymbol{y}}_r|\boldsymbol{x})} \\
&= \ln \frac{p(\tilde{y}_{rj}|y_{rj}=+1)}{p(\tilde{y}_{rj}|y_{rj}=-1)},
\end{aligned}
$$

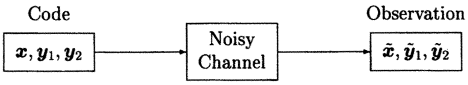

Fig. 1. Turbo codes.

$$
\begin{aligned}
L_r\boldsymbol{x} &\stackrel{\text{def}}{=} F(l\boldsymbol{x}, l\boldsymbol{y}_r) \\
&= \left\{ \ln \frac{\sum_{\{\boldsymbol{x}:x_i=+1\}} p(\tilde{\boldsymbol{x}}|\boldsymbol{x}) p(\tilde{\boldsymbol{y}}_r|\boldsymbol{x})}{\sum_{\{\boldsymbol{x}:x_i=-1\}} p(\tilde{\boldsymbol{x}}|\boldsymbol{x}) p(\tilde{\boldsymbol{y}}_r|\boldsymbol{x})} \right\}
\end{aligned}
$$

These are used to define the turbo decoding algorithm in the following manner (Fig. 2).

[Turbo Decoding]

1. Set $\xi_1 = 0$, $t = 1$.

2. Calculate $L_1\boldsymbol{x}^{(t)} = F((l\boldsymbol{x} + \xi_1), l\boldsymbol{y}_1)$, and ξ_2 is updated as

$$\boldsymbol{\xi}_2 = L_1\boldsymbol{x}^{(t)} - (l\boldsymbol{x} + \boldsymbol{\xi}_1) \tag{1}$$

3. Calculate $L_2\boldsymbol{x}^{(t)} = F((l\boldsymbol{x} + \xi_2), l\boldsymbol{y}_2)$, and ξ_1 is updated as

$$\boldsymbol{\xi}_1 = L_2\boldsymbol{x}^{(t)} - (l\boldsymbol{x} + \boldsymbol{\xi}_2) \tag{2}$$

4. 2 and 3 are repeated while t is increased by 1 until $L_1\boldsymbol{x}^{(t)} = L_2\boldsymbol{x}^{(t)} = L_1\boldsymbol{x}^{(t+1)} = L_2\boldsymbol{x}^{(t+1)}$ is satisfied.

The algorithm does not necessarily converge. The operations are repeated until it converges or a predetermined number of iterations is achieved. Usually this limit is from several to about ten.

2.2. MPM decoding

The goal of decoding turbo codes is to obtain the MPM (maximum posterior marginal) decoding. In MPM decoding, the distribution $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$ is marginalized down to each component of $\boldsymbol{x}_i$, and the code which maximizes the product of marginal distributions is taken to be

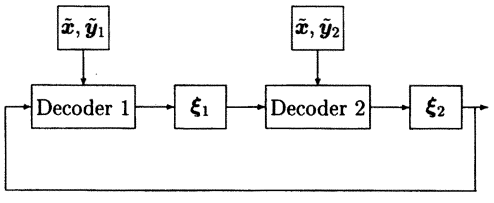

Fig. 2. Turbo decoding.

the inferred result. Let us first consider $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$. Since a memoryless BSC is assumed, the following is true:

$$p(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2|\boldsymbol{x}) = p(\tilde{\boldsymbol{x}}|\boldsymbol{x})p(\tilde{\boldsymbol{y}}_1|\boldsymbol{x})p(\tilde{\boldsymbol{y}}_2|\boldsymbol{x})$$

The right-hand side can be written as follows:

$$p(\tilde{\boldsymbol{x}}|\boldsymbol{x}) = \exp\left(\beta\tilde{\boldsymbol{x}}\cdot\boldsymbol{x} - N\psi(\beta)\right)$$
$$p(\tilde{\boldsymbol{y}}_r|\boldsymbol{x}) = \exp\left(\beta\tilde{\boldsymbol{y}}\cdot\boldsymbol{y}_r(\boldsymbol{x}) - L\psi(\beta)\right), \quad r = 1, 2$$
$$\psi(\beta) = \ln(e^{-\beta} + e^{\beta})$$

Here, β is a positive real number, and the bit error rate f_n of the BSC is expressed as $f_n = (1 - \tanh\beta)/2$. Assuming that $c_0(\boldsymbol{x}) = \beta\tilde{\boldsymbol{x}}\cdot\boldsymbol{x}$, $c_1(\boldsymbol{x}) = \beta\tilde{\boldsymbol{y}}_1\cdot\tilde{\boldsymbol{y}}_1$, $c_2(\boldsymbol{x}) = \beta\tilde{\boldsymbol{y}}_2\cdot\boldsymbol{y}_2$, $p(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2|\boldsymbol{x})$ is written as follows:

$$p(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2|\boldsymbol{x}) = \exp\left(c_0(\boldsymbol{x}) + c_1(\boldsymbol{x}) + c_2(\boldsymbol{x}) - (N + 2L)\psi(\beta)\right)$$

If we consider a uniform distribution $p(\boldsymbol{x}) = 1/2^N$ as the prior distribution of $\boldsymbol{x}$, the posterior distribution of $\boldsymbol{x}$ is given as follows:

$$p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2) = \frac{p(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2|\boldsymbol{x})}{\sum_{\boldsymbol{x}} p(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2|\boldsymbol{x})} = C\exp(c_0(\boldsymbol{x}) + c_1(\boldsymbol{x}) + c_2(\boldsymbol{x})) \quad (3)$$

Also, we define

$$\Pi\circ p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2) \stackrel{\text{def}}{=} \prod_{i=1}^{N} p_i(x_i|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$$

where Π is the operator of marginalization.

MPM decoding can be defined in the following manner:

$$\hat{\boldsymbol{x}} = \operatorname*{argmax}_{\boldsymbol{x}} \Pi\circ p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$$

In turbo decoding, the MPM decoding is the ultimate goal, but the calculation cost of marginalization grows exponentially with respect to N, and the marginalization is not tractable. The turbo decoding algorithm provides an approximation to MPM decoding.

2.3. Preliminaries of information geometry

In this section, preliminaries of information geometry are given. Let us consider a set S of probability distributions for $\boldsymbol{x}$. It is a set of multinomial distributions of 2^N elements. Its degree of freedom is $(2^N - 1)$, and it is an exponential family:

$$S = \left\{p(\boldsymbol{x}) \middle| p(\boldsymbol{x}) \geq 0, \boldsymbol{x} \in \{-1, +1\}^N, \sum p(\boldsymbol{x}) = 1\right\}$$

Let us now define the *e*-flat and *m*-flat submanifolds included in S.

e-flat: Manifold $M \in S$ is *e*-flat when the $r(\boldsymbol{x}; t)$ defined by the following equation is contained in M, where all $q(\boldsymbol{x})$, $p(\boldsymbol{x}) \in M$:

$$\ln r(\boldsymbol{x}; t) = (1 - t)\ln q(\boldsymbol{x}) + t\ln p(\boldsymbol{x}) + c, \quad t \in R$$

Here, c is the normalization constant.

m-flat: Manifold $M \in S$ is *m*-flat when the $r(\boldsymbol{x}; t)$ defined by the following equation is contained in M, where all $q(\boldsymbol{x})$, $p(\boldsymbol{x}) \in M$:

$$r(\boldsymbol{x}; t) = (1 - t)q(\boldsymbol{x}) + tp(\boldsymbol{x}), \quad t \in [0, 1]$$

An *m*-projection will next be defined.

[Definition 1] Let us assume that M is an *e*-flat submanifold of S. An *m*-projection from $q(\boldsymbol{x}) \in S$ to M will be a point on M in which the *Kullback–Leibler* (*KL*) divergence from $q(\boldsymbol{x})$ to M is at a minimum, and it can be defined in the following manner:

$$\Pi_M\circ q(\boldsymbol{x}) = \operatorname*{argmin}_{p(\boldsymbol{x})\in M} D[q(\boldsymbol{x}); p(\boldsymbol{x})]$$

[Theorem 1] The *m*-projection $\Pi_M\circ q(\boldsymbol{x})$ of S from $q(\boldsymbol{x}) \in S$ to an *e*-flat submanifold M is unique.

The KL divergence $D[\cdot;\cdot]$ is defined as follows:

$$D\left[q(\boldsymbol{x}); p(\boldsymbol{x})\right] = \sum_{\boldsymbol{x}} q(\boldsymbol{x})\ln\frac{q(\boldsymbol{x})}{p(\boldsymbol{x})}$$

The KL divergence satisfies the inequality $D[q(\boldsymbol{x}); p(\boldsymbol{x})] \geqq 0$ and becomes 0 if $q(\boldsymbol{x}) = p(\boldsymbol{x})$ holds for any $\boldsymbol{x}$.

To understand the turbo decoding algorithm, let us consider a submanifold M_D consisting of factorizable distribution. The definition will be as follows:

$$M_D = \left\{p(\boldsymbol{x}; \boldsymbol{\theta}) = \exp\left(\boldsymbol{\theta}\cdot\boldsymbol{x} - \psi(\boldsymbol{\theta})\right) \middle| \boldsymbol{\theta} \in \mathcal{R}^N\right\}$$

where $\psi(\boldsymbol{\theta})$ is the normalizing factor defined in the following manner:

$$\psi(\boldsymbol{\theta}) = \ln\sum_{\boldsymbol{x}}\exp(\boldsymbol{\theta}\cdot\boldsymbol{x}) = \sum_i \ln(e^{-\theta_i} + e^{\theta_i})$$

By its definition, M_D is an exponential family, and since an exponential family is *e*-flat, M_D is an *e*-flat submanifold [8]. Parameter $\boldsymbol{\theta}$ gives the coordinate system of the manifold M_D and is called the natural parameter. Another coordinate system, $\boldsymbol{\eta}$, which is called the expectation parameter, is defined as follows:

$$\boldsymbol{\eta} = \sum_{\boldsymbol{x}} p(\boldsymbol{x}; \boldsymbol{\theta})\boldsymbol{x}$$

The following one-to-one relation exists between $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$:

$$\boldsymbol{\eta} = \partial_{\boldsymbol{\theta}} \psi(\boldsymbol{\theta}) \tag{4}$$

[Theorem 2] The marginalized probability distribution $\Pi \circ q(\boldsymbol{x})$ of $q(\boldsymbol{x})$ is an m-projection of $q(\boldsymbol{x})$ on M_D.

(Proof) Let us consider an m-projection of $q(\boldsymbol{x})$ to M_D. From Theorem 1, we can differentiate $D[q(\boldsymbol{x}); p(\boldsymbol{x}; \boldsymbol{\theta})]$ with respect to $\boldsymbol{\theta}$. Using the results of Eq. (4), the following is obtained:

$$\partial_{\boldsymbol{\theta}} D[q(\boldsymbol{x}); p(\boldsymbol{x}; \boldsymbol{\theta})] = \boldsymbol{\eta} - \sum_{\boldsymbol{x}} q(\boldsymbol{x}) \boldsymbol{x}$$

Consequently, $\boldsymbol{\eta}^* = \Sigma_{\infty}\, q(\boldsymbol{x})\boldsymbol{x}$, where $\boldsymbol{\eta}^*$ is the projected $\boldsymbol{\eta}$ coordinate, gives the m-projection. This indicates that an m-projection is expressed by the independent expected value of each component of $\boldsymbol{x}$ based on $q(\boldsymbol{x})$, which proves the theorem.

Furthermore, MPM decoding can be written as follows.

$$\hat{\boldsymbol{x}} = \mathrm{sgn}(\boldsymbol{\eta}^*)$$

where $\mathrm{sgn}(\cdot)$ is applied independently to each bit.

2.4. Information geometrical view of turbo decoding

Turbo decoding does not use two observed parity check words simultaneously as $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$, but uses $p(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1|\boldsymbol{x})$, $p(\boldsymbol{x}, \tilde{\boldsymbol{y}}_2|\boldsymbol{x})$ in which $\tilde{\boldsymbol{x}}$ and only one of the observed parity check words is taken into account. The following gives the form of $p(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_r|\boldsymbol{x})$, $r = 1, 2$:

$$p(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_r|\boldsymbol{x}) = \exp\left(c_0(\boldsymbol{x}) + c_r(\boldsymbol{x}) - (N+L)\psi(\beta)\right)$$

The following is obtained for these distributions when $\omega(\boldsymbol{x}; \boldsymbol{\xi}) \in M_D$ is taken to be the prior distribution of $\boldsymbol{x}$, and a posterior distribution is given as follows:

$$\begin{aligned} p_r(\boldsymbol{x}; \boldsymbol{\xi}) &= p_r(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_r; \boldsymbol{\xi}) \\ &= \frac{p_r(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_r|\boldsymbol{x})\omega(\boldsymbol{x}; \boldsymbol{\xi})}{\sum_{\boldsymbol{x}} p_r(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_r|\boldsymbol{x})\omega(\boldsymbol{x}; \boldsymbol{\xi})} \\ &= \exp(c_0(\boldsymbol{x}) + c_r(\boldsymbol{x}) + \boldsymbol{\xi} \cdot \boldsymbol{x} - \varphi_r(\boldsymbol{\xi})) \end{aligned}$$

$\varphi_r(\xi)$ is the normalization factor. Here, we assume that the m-projection from $p_r(\boldsymbol{x}; \xi)$, $r = 1, 2$, to M_D is tractable in polynomial time. The turbo decoding varies iteratively, and the m-projection of $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$ is approximated.

In this section, the information geometrical view of turbo decoding is given. First, we define three important manifolds:

$$\begin{aligned} &M_0 \\ &= \left\{ p_0(\boldsymbol{x}; \boldsymbol{\xi}) = \exp\left(c_0(\boldsymbol{x}) + \boldsymbol{\xi} \cdot \boldsymbol{x} - \varphi_0(\boldsymbol{\xi})\right) \middle| \boldsymbol{\xi} \in \mathcal{R}^N \right\} \\ &M_1 = \left\{ p_1(\boldsymbol{x}; \boldsymbol{\xi}) \middle| \boldsymbol{\xi} \in \mathcal{R}^N \right\} \\ &M_2 = \left\{ p_2(\boldsymbol{x}; \boldsymbol{\xi}) \middle| \boldsymbol{\xi} \in \mathcal{R}^N \right\} \end{aligned}$$

Here, ξ defines the coordinate system of each manifold. Since $c_0(\boldsymbol{x}) = \beta\tilde{\boldsymbol{x}} \cdot \boldsymbol{x}$, the condition $p(\boldsymbol{x}; \boldsymbol{\theta}') = p_0(\boldsymbol{x}; \boldsymbol{\theta})$ holds for $p(\boldsymbol{x}; \boldsymbol{\theta}') \in M_D$, if $\boldsymbol{\theta}' = \boldsymbol{\theta} + \beta\tilde{\boldsymbol{x}}$ is assumed. Consequently, M_0 is equivalent to M_D. Let $\pi_{M_0} \circ q(\boldsymbol{x})$ be a coordinate ξ defined by the m-projection of $q(\boldsymbol{x})$ on M_0:

$$\pi_{M_0} \circ q(\boldsymbol{x}) = \operatorname*{argmin}_{\boldsymbol{\xi} \in \mathcal{R}^N} D[q(\boldsymbol{x}); p_0(\boldsymbol{x}; \boldsymbol{\xi})]$$

The condition $\tau_{M_0} \circ q(\boldsymbol{x}) = \pi_{M_D} \circ q(\boldsymbol{x}) + \beta\tilde{\boldsymbol{x}}$ holds. The turbo decoding is written in the following manner with π_{M_0} (Fig. 3).

[Expression of Turbo Decoding as Information Geometry]

1. Set $\xi_1^t = 0$ for $t = 0$, and $t = 1$.

2. Calculate $\pi_{M_0} \circ p_2(\boldsymbol{x}; \xi_1^t)$, which is the projection of $p_2(\boldsymbol{x}; \xi_1^t)$ to M_0, and update ξ_2^{t+1} as follows:

$$\boldsymbol{\xi}_2^{t+1} = \pi_{M_0} \circ p_2(\boldsymbol{x}; \boldsymbol{\xi}_1^t) - \boldsymbol{\xi}_1^t \tag{5}$$

3. Calculate $\pi_{M_0} \circ p_1(\boldsymbol{x}; \xi_2^{t+1})$, which is the projection of $p_1(\boldsymbol{x}; \xi_2^{t+1})$ to M_0, and update ξ_1^{t+1} as follows:

$$\boldsymbol{\xi}_1^{t+1} = \pi_{M_0} \circ p_1(\boldsymbol{x}; \boldsymbol{\xi}_2^{t+1}) - \boldsymbol{\xi}_2^{t+1} \tag{6}$$

4. Return to step 2 if $\pi_{M_0} \circ p_1(\boldsymbol{x}; \xi_2^{t+1}) \neq \tau_{M_0} \circ p_2(\boldsymbol{x}; \xi_1^{t+1})$.

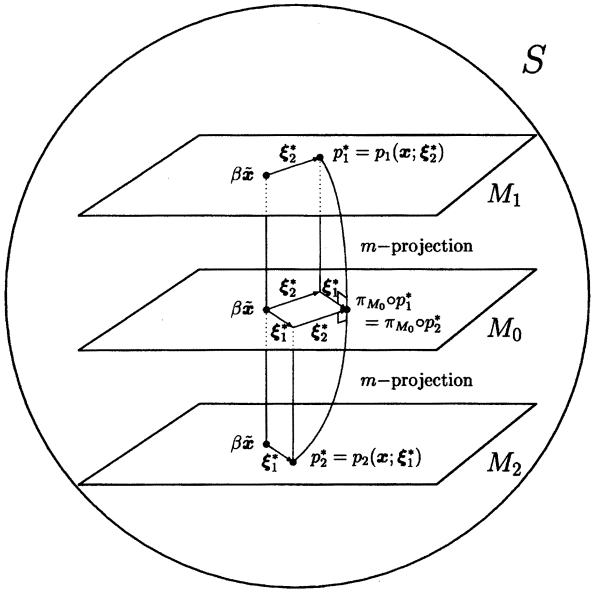

Fig. 3. Information geometry of turbo decoding.

$L_1\boldsymbol{x}^{(t)}$, $L_2\boldsymbol{x}^{(t)}$ in Eqs. (1) and (2), that is, the expressions $F((l\boldsymbol{x}+\xi_1), l\boldsymbol{y}_1)$, $F((l\boldsymbol{x}+\xi_2), l\boldsymbol{y}_2)$ correspond to $\tau_{M_0}\circ p_2(\boldsymbol{x}; \xi_1^t)$, $\pi_{M_0}\circ p_1(\boldsymbol{x}; \xi_2^{t+1})$ in Eqs. (5) and (6). In addition, ξ_1, ξ_2 in Eqs. (1) and (2) correspond to ξ_1^t, ξ_2^{t+1} in Eqs. (5) and (6).

3. Properties of Turbo Decoding

3.1. Properties of stationary points

Let us assume that ξ_1^*, ξ_2^* are the convergence points of turbo decoding. The final result is the M_0 coordinate at which $\tau_{M_0}\circ p_1(\boldsymbol{x}; \xi_2^*) = \pi_{M_0}\circ p_2(\boldsymbol{x}; \xi_1^*)$. Let us define the point as $\boldsymbol{\theta}^*$. First, the following is obtained based on convergence conditions.

$$\Pi\circ p_1(\boldsymbol{x}; \boldsymbol{\xi}_2^*) = \Pi\circ p_2(\boldsymbol{x}; \boldsymbol{\xi}_1^*) = p_0(\boldsymbol{x}; \boldsymbol{\theta}^*)$$

Also, the following is obtained based on $\tau_{M_0}\circ p_1(\boldsymbol{x}; \xi_2^*) = \pi_{M_0}\circ p_2(\boldsymbol{x}; \xi_1^*)$ and on steps 2 and 3 of the algorithm:

$$\boldsymbol{\theta}^* = \boldsymbol{\xi}_1^* + \boldsymbol{\xi}_2^*$$

In turbo decoding, the results of true MPM decoding are approximated as $\boldsymbol{\theta}^* = \xi_1^* + \xi_2^*$:

$$p_0(\boldsymbol{x}; \boldsymbol{\theta}^*) = \exp(c_0(\boldsymbol{x}) + \boldsymbol{\xi}_1^* \cdot \boldsymbol{x} + \boldsymbol{\xi}_2^* \cdot \boldsymbol{x} - \varphi_0(\boldsymbol{\theta}^*)) \qquad (7)$$

Intuitively, ξ_2 in Eq. (7) is substituted by $c_2(\boldsymbol{x})$ in step 2, ξ_1 in Eq. (7) is substituted by $c_1(\boldsymbol{x})$ in step 3 of turbo decoding, and ξ_1^* is determined. Consequently, the influences of $c_1(\boldsymbol{x})$ and $c_2(\boldsymbol{x})$ are expressed by ξ_1^* and ξ_2^*, respectively, but their influences usually cannot be linearly separated on M_0.

Let us define the ξ_1 and ξ_2 that satisfy the equation below as $\xi_1(\boldsymbol{\theta})$ and $\xi_2(\boldsymbol{\theta})$, respectively:

$$\Pi\circ p_1(\boldsymbol{x}; \boldsymbol{\xi}_2) = \Pi\circ p_2(\boldsymbol{x}; \boldsymbol{\xi}_1) = p_0(\boldsymbol{x}; \boldsymbol{\theta})$$

Also, let us define the *m*- and *e*-flat manifolds that connect $p_0(\boldsymbol{x}; \boldsymbol{\theta})$, $p_1(\boldsymbol{x}; \xi_2(\boldsymbol{\theta}))$, $p_2(\boldsymbol{x}; \xi_1(\boldsymbol{\theta}))$ as $M(\theta)$ and $E(\theta)$, respectively:

$$M(\boldsymbol{\theta}) = \left\{ p(\boldsymbol{x}) \Big| \sum_{\boldsymbol{x}} p(\boldsymbol{x})\boldsymbol{x} = \sum_{\boldsymbol{x}} p_0(\boldsymbol{x}; \boldsymbol{\theta})\boldsymbol{x} \right\}$$

$$E(\boldsymbol{\theta}) = \left\{ p = C p_0^{t_0} p_1^{t_1} p_2^{t_2} \Big| \sum_{r=0}^{2} t_r = 1 \right\}$$

For every $p(\boldsymbol{x}) \in M(\boldsymbol{\theta})$, its *m*-projection to M_0 is $p_0(\boldsymbol{x}; \boldsymbol{\theta})$.

[Theorem 3] At a stationary point, the three distributions p_0^*, p_1^*, p_2^* are included in $M(\boldsymbol{\theta}^*)$, and the four distributions p_0^*, p_1^*, p_2^*, $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \boldsymbol{y}_2)$ are included in $E(\boldsymbol{\theta}^*)$ (Fig. 4).

(Proof) Based on the definition, the fact that $p_0^*, p_1^*, p_2^* \in (M(\boldsymbol{\theta}^*), E(\boldsymbol{\theta}^*))$ is proven. The fact that the four distributions $p_0^*, p_1^*, p_2^*, p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2) \in E(\boldsymbol{\theta}^*)$ is proven from the following result:

$$\begin{aligned} & C\frac{p_1(\boldsymbol{x}; \boldsymbol{\xi}_2^*) p_2(\boldsymbol{x}; \boldsymbol{\xi}_1^*)}{p_0(\boldsymbol{x}; \boldsymbol{\theta}^*)} \\ &= C\exp(c_0(\boldsymbol{x}) + c_1(\boldsymbol{x}) + c_2(\boldsymbol{x})) \\ &= p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2) \end{aligned}$$

by setting $t_0 = -1$, $t_1 = t_2 = 1$ and using $\boldsymbol{\theta}^* = \xi_1^* + \xi_2^*$.

$M(\boldsymbol{\theta}_{MPM}^*)$ includes $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$ if a true solution for MPM decoding is $\boldsymbol{\theta}_{MPM}^*$. Generally, however, the solution $\boldsymbol{\theta}^*$ for turbo decoding and $\boldsymbol{\theta}_{MPM}^*$ do not agree with each other. Consequently, $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2) \notin M(\boldsymbol{\theta}^*)$, but is included in $E(\boldsymbol{\theta}^*)$. Since *e*-flatness and *m*-flatness generally do not agree with each other, there is a discrepancy between the manifolds $E(\boldsymbol{\theta}^*)$ and $M(\boldsymbol{\theta}^*)$. Turbo decoding obtains the approximation by substituting $M(\boldsymbol{\theta}^*)$ for $E(\boldsymbol{\theta}^*)$. Similar structures exist in other statistical physics techniques [9–12].

3.2. Stability of stationary point

Let us assume that the convergence points of turbo decoding are ξ_1^*, ξ_2^*, $\boldsymbol{\theta}^* = \xi_1^* + \xi_2^*$.

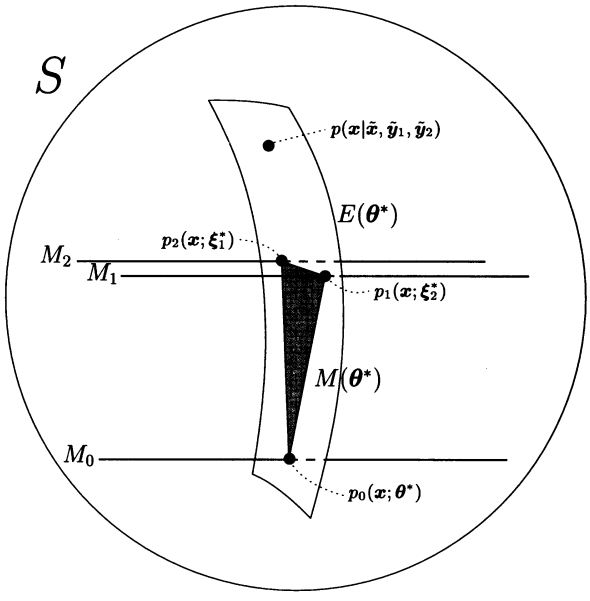

Fig. 4. Information geometrical view of turbo codes.

Now we define $G_0(\boldsymbol{\theta})$, $G_1(\xi)$, $G_2(\xi)$ as the Fisher information matrices of $p_0(\boldsymbol{x};\boldsymbol{\theta})$, $p_1(\boldsymbol{x};\xi)$, $p_2(\boldsymbol{x};\xi)$, respectively; I_N is a unit matrix, and $\boldsymbol{\eta}_0(\boldsymbol{\theta})$, $\boldsymbol{\eta}_1(\xi)$, $\boldsymbol{\eta}_2(\xi)$ are the expectation parameters of the corresponding distributions. The following condition holds for the convergence points $\boldsymbol{\eta}_0(\boldsymbol{\theta}^*)$, $\boldsymbol{\eta}_1(\xi_2^*) = \boldsymbol{\eta}_2(\xi_1^*)$. The following also holds for each of these ($r = 0, 1, 2$):

$$G_r(\boldsymbol{\theta}) = \partial_{\boldsymbol{\theta}\boldsymbol{\theta}'}\varphi_r(\boldsymbol{\theta}) = \partial_{\boldsymbol{\theta}}\boldsymbol{\eta}_r(\boldsymbol{\theta})$$

To see the stability, we add a sufficiently small vector $\boldsymbol{\delta}$ to ξ_1^* as $\xi_1 = \xi_1^* + \boldsymbol{\delta}$, and set it as the initial value of the algorithm, then the turbo decoding algorithm is applied once. Setting $\xi_1' = \xi_1^* + \boldsymbol{\delta}'$ to be the parameter after a single cycle of turbo decoding, a linear stability analysis is carried out. From step 2, the following is obtained:

$$\boldsymbol{\eta}_0(\boldsymbol{\theta}^* + \Delta\boldsymbol{\theta}) = \boldsymbol{\eta}_2(\boldsymbol{\xi}_1^* + \boldsymbol{\delta})$$

When this expression is expanded, the result is

$$\begin{aligned}&\boldsymbol{\eta}_0(\boldsymbol{\theta}^*) + G_0(\boldsymbol{\theta}^*)\Delta\boldsymbol{\theta} = \boldsymbol{\eta}_1(\boldsymbol{\xi}_2^*) + G_2(\boldsymbol{\xi}_1^*)\boldsymbol{\delta}\\ &\Delta\boldsymbol{\theta} = G_0(\boldsymbol{\theta}^*)^{-1}G_2(\boldsymbol{\xi}_1^*)\boldsymbol{\delta}\end{aligned}$$

Also, ξ_2 in step 2 becomes

$$\boldsymbol{\xi}_2 = \boldsymbol{\xi}_2^* + \left(G_0(\boldsymbol{\theta}^*)^{-1}G_2(\boldsymbol{\xi}_1^*) - I_N\right)\boldsymbol{\delta}$$

Step 3 is treated in the same manner. As a result, $\boldsymbol{\delta}$ is updated as

$$\begin{aligned}&\boldsymbol{\delta}' = \mathcal{T}\boldsymbol{\delta}\\ &\mathcal{T} = \left(G_0(\boldsymbol{\theta}^*)^{-1}G_1(\boldsymbol{\xi}_2^*) - I_N\right)\left(G_0(\boldsymbol{\theta}^*)^{-1}G_2(\boldsymbol{\xi}_1^*) - I_N\right)\end{aligned}$$

which gives the linearized approximation of turbo decoding.

[Theorem 4] Let λ_i be the eigenvalue of $\mathcal{T}$. The stationary point will be stable if $|\lambda_i| < 1$ holds for every i.

This result agrees with that of Ref. 2.

3.3. Cost function and stationary points

Let us consider the following function, assuming $\boldsymbol{\theta} = \xi_1 + \xi_2$:

$$\mathcal{F}(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2) = \varphi_0(\boldsymbol{\theta}) - \varphi_1(\boldsymbol{\xi}_2) - \varphi_2(\boldsymbol{\xi}_1)$$

[Theorem 5] The stationary points ξ_1^*, ξ_2^* of turbo decoding are the critical points of $\mathcal{F}$.

(Proof) With direct differentiation, the following is obtained:

$$\begin{aligned}&\partial_{\boldsymbol{\xi}_1}\mathcal{F} = \partial_{\boldsymbol{\theta}}\varphi_0(\boldsymbol{\theta}) - \partial_{\boldsymbol{\xi}_1}\varphi_2(\boldsymbol{\xi}_1) = \boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_2(\boldsymbol{\xi}_1)\\ &\partial_{\boldsymbol{\xi}_2}\mathcal{F} = \partial_{\boldsymbol{\theta}}\varphi_0(\boldsymbol{\theta}) - \partial_{\boldsymbol{\xi}_2}\varphi_1(\boldsymbol{\xi}_2) = \boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_1(\boldsymbol{\xi}_2)\end{aligned}$$

The above differential is 0 because $\boldsymbol{\eta}_0(\boldsymbol{\theta}^*) = \boldsymbol{\eta}_2(\xi_1^*) = \boldsymbol{\eta}_1(\xi_2^*)$ at the equilibrium.

The turbo decoding algorithm can be approximated with

$$\begin{pmatrix}\boldsymbol{\xi}_1^{t+1}\\ \boldsymbol{\xi}_2^{t+1}\end{pmatrix} - \begin{pmatrix}\boldsymbol{\xi}_1^{t}\\ \boldsymbol{\xi}_2^{t}\end{pmatrix} \simeq \begin{pmatrix}O & G_0(\boldsymbol{\theta})^{-1}\\ G_0(\boldsymbol{\theta})^{-1} & O\end{pmatrix}\begin{pmatrix}\partial_{\boldsymbol{\xi}_1}\mathcal{F}\\ \partial_{\boldsymbol{\xi}_2}\mathcal{F}\end{pmatrix}$$

when the parameter changes by a very small value. The Hessian of $\mathcal{F}$ is calculated as

$$\mathcal{H} = \begin{pmatrix}\partial_{\boldsymbol{\xi}_1\boldsymbol{\xi}_1}\mathcal{F} & \partial_{\boldsymbol{\xi}_1\boldsymbol{\xi}_2}\mathcal{F}\\ \partial_{\boldsymbol{\xi}_2\boldsymbol{\xi}_1}\mathcal{F} & \partial_{\boldsymbol{\xi}_2\boldsymbol{\xi}_2}\mathcal{F}\end{pmatrix} = \begin{pmatrix}G_0 - G_1 & G_0\\ G_0 & G_0 - G_2\end{pmatrix}$$

Here, the variables are changed to $\boldsymbol{\theta} = \xi_1 + \xi_2$, $\boldsymbol{\nu} = \xi_1 - \xi_2$, yielding

$$\begin{aligned}&\begin{pmatrix}\partial_{\boldsymbol{\theta}\boldsymbol{\theta}}\mathcal{F} & \partial_{\boldsymbol{\theta}\boldsymbol{\nu}}\mathcal{F}\\ \partial_{\boldsymbol{\nu}\boldsymbol{\theta}}\mathcal{F} & \partial_{\boldsymbol{\nu}\boldsymbol{\nu}}\mathcal{F}\end{pmatrix}\\ &\quad = \frac{1}{4}\begin{pmatrix}4G_0(\boldsymbol{\theta}) - (G_1 + G_2) & (G_1 - G_2)\\ (G_1 - G_2) & -(G_1 + G_2)\end{pmatrix}\end{aligned}$$

from which we can see that $\partial_{\boldsymbol{\theta}\boldsymbol{\theta}}\mathcal{F}$ is probably positive definite and that $\partial_{\boldsymbol{\nu}\boldsymbol{\nu}}\mathcal{F}$ is always negative. Consequently, the stationary points of turbo decoding are often believed to be a saddle point.

3.4. Difference between turbo decoding results and MPM solution

Theorem 3 shows that the difference between MPM decoding and turbo decoding is the difference between $M(\boldsymbol{\theta})$ and $E(\boldsymbol{\theta})$. Based on this result, we evaluate the difference between the true MPM solution and the turbo decoding solution by the perturbation analysis. First, $\boldsymbol{\theta} = (\theta^1, \ldots, \theta^N)^T$, $\boldsymbol{\nu} = (\nu^1, \nu^2)^T$, and $\boldsymbol{c}(\boldsymbol{x}) \stackrel{\text{def}}{=} (c_1(\boldsymbol{x}), c_2(\boldsymbol{x}))^T$ are used to define $p(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\nu})$ in the following manner:

$$\begin{aligned}&p(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\nu}) = \exp(c_0(\boldsymbol{x}) + \boldsymbol{\theta}\cdot\boldsymbol{x} + \boldsymbol{\nu}\cdot\boldsymbol{c}(\boldsymbol{x}) - \varphi(\boldsymbol{\theta},\boldsymbol{\nu}))\\ &\varphi(\boldsymbol{\theta},\boldsymbol{\nu}) = \ln\sum_{\boldsymbol{x}}\exp(c_0(\boldsymbol{x}) + \boldsymbol{\theta}\cdot\boldsymbol{x} + \boldsymbol{\nu}\cdot\boldsymbol{c}(\boldsymbol{x}))\end{aligned}$$

This distribution includes $p_0(\boldsymbol{x};\boldsymbol{\theta})(\boldsymbol{\nu} = 0)$, $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$ ($\boldsymbol{\theta} = \boldsymbol{0}$, $\boldsymbol{\nu} = (1, 1)^T$), $p_r(\boldsymbol{x};\xi)$ ($\boldsymbol{\theta} = \xi$, $\boldsymbol{\nu} = \boldsymbol{e}_r$) as special cases. Here, $\boldsymbol{e}_r$ is

$$\boldsymbol{e}_1 = (1, 0)^T, \quad \boldsymbol{e}_2 = (0, 1)^T$$

The expectation parameter $\boldsymbol{\eta}(\boldsymbol{\theta},\boldsymbol{\nu})$ is defined as follows:

$$\boldsymbol{\eta}(\boldsymbol{\theta},\boldsymbol{\nu}) = \partial_{\boldsymbol{\theta}}\varphi(\boldsymbol{\theta},\boldsymbol{\nu}) = \sum_{\boldsymbol{x}} p(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\nu})\boldsymbol{x}$$

This defines a submanifold $M(\boldsymbol{\theta}^*)$ in which the expectation parameters of all distributions are equal to $\boldsymbol{\eta}(\boldsymbol{\theta}^*)$. In other words, we have a set of distributions for which the condition $\boldsymbol{\eta}(\boldsymbol{\theta}, \boldsymbol{\nu}) = \boldsymbol{\eta}(\boldsymbol{\theta}^*)$ is satisfied. Hereafter, we use the following notations $\boldsymbol{\eta}(\boldsymbol{\theta}) \stackrel{\text{def}}{=} \boldsymbol{\eta}(\boldsymbol{\theta}, 0)$ and $\boldsymbol{\eta}^* \stackrel{\text{def}}{=} \boldsymbol{\eta}(\boldsymbol{\theta}^*)$.

The dependency of $\boldsymbol{\theta}$ on $\boldsymbol{\nu}$ will now be considered using the perturbation analysis under the restriction that $p(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\nu}) \in M(\boldsymbol{\theta}^*)$. In the perturbation analysis, the contribution of $\{v^r c_r(\boldsymbol{x})\}$ to $p(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\nu})$ is assumed to be small, we take the Taylor series of $p(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\nu})$ with respect to $\{v^r\}$ up to $O(\|\boldsymbol{\nu}\|^2)$, and the effect of $\{v^r c_r(\boldsymbol{x})\}$ is evaluated. The Taylor expansion gives

$$\eta_i(\boldsymbol{\theta}, \boldsymbol{v}) = \eta_i^* + \sum_j \partial_j \eta_i^* \Delta\theta^j + \sum_r \partial_r \eta_i^* v^r + \frac{1}{2}\sum_{r,s} \partial_r \partial_s \eta_i^* v^r v^s + \sum_{j,r} \partial_r \partial_j \eta_i^* v^r \Delta\theta^j + \frac{1}{2}\sum_{k,l} \partial_k \partial_l \eta_i^* \Delta\theta^k \Delta\theta^l + O(\|\boldsymbol{v}\|^3) + O(\|\Delta\boldsymbol{\theta}\|^3)$$

where $\{i, j, k, l\}$ are the subscripts of $\boldsymbol{\theta}$, $\{r, s\}$ are the subscripts of $\boldsymbol{\nu}$, and $\Delta\boldsymbol{\theta} \stackrel{\text{def}}{=} \boldsymbol{\theta} - \boldsymbol{\theta}^*$. From the condition that the distribution must be on $M(\boldsymbol{\theta}^*)$, $\eta_i(\boldsymbol{\theta}, \boldsymbol{\nu}) = \eta_i^*$ holds. Also, $\{g_{ij}\}$, which is the Fisher information matrix of $p(\boldsymbol{x}; \boldsymbol{\theta}^*, \boldsymbol{0})$, is a diagonal matrix. Using these results, the following equation is derived:

$$\Delta\theta^i = -g^{ii}\Big[\sum_r \partial_r \eta_i^* v^r - \frac{1}{2}\sum_{r,s} \partial_r \partial_s \eta_i^* v^r v^s - \frac{1}{2}\sum_{k,l} \partial_k \partial_l \eta_i^* \Delta\theta^k \Delta\theta^l - \sum_{k,r} \partial_r \partial_k \eta_i^* v^r \Delta\theta^k\Big] + O(\|\boldsymbol{v}\|^3) + O(\|\Delta\boldsymbol{\theta}\|^3)$$

where $g^{ii} = 1/g_{ii}$. By ignoring third- and higher-order terms of $v^r \Delta\theta^i$ is rewritten as follows:

$$\Delta\theta^i \simeq -g^{ii}\sum_r A_{ir} v^r - \frac{g^{ii}}{2} \times \sum_{r,s}\Big(\partial_r - \sum_k g^{kk} A_{kr}\partial_k\Big)\Big(\partial_s - \sum_j g^{jj} A_{js}\partial_j\Big)\eta_i^* v^r v^s \tag{8}$$

where $A_{ir} = \partial_r \eta_i^*$.

When $\boldsymbol{\nu} = \boldsymbol{e}_1$, $p(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\nu})$ is restricted to $M(\boldsymbol{\theta}^*)$, so $\boldsymbol{\theta} = \boldsymbol{\xi}_2^*$ and $\Delta\boldsymbol{\theta} = \boldsymbol{\xi}_2^* - \boldsymbol{\theta}^* = -\boldsymbol{\xi}_1^*$. Based on the same argument, $\Delta\boldsymbol{\theta} = -\boldsymbol{\xi}_2^*$ when $\boldsymbol{\nu} = \boldsymbol{e}_2$. The following expression is derived from Eq. (8):

$$-\xi_r^{i,*} \simeq -g^{ii} A_{ir} - \frac{g^{ii}}{2}\Big(\partial_r - \sum_k g^{kk} A_{kr}\partial_k\Big)\Big(\partial_r - \sum_j g^{jj} A_{jr}\partial_j\Big)\eta_i^* \tag{9}$$

On the other hand, if we assume that $\boldsymbol{\nu} = \Sigma_r \boldsymbol{e}_r$ and define $\bar{\boldsymbol{\theta}}$ as the parameter that satisfies the condition, generally, $\bar{\boldsymbol{\theta}} \neq \boldsymbol{\theta}$. This indicates that $p(\boldsymbol{x}|\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2)$ is not necessarily included in $M(\boldsymbol{\theta}^*)$. It follows from Eq. (8) that

$$\bar{\theta}^i - \theta^{i,*} \simeq -g^{ii}\sum_r A_{ir} - \frac{g^{ii}}{2}\sum_r \Big(\partial_r - \sum_k g^{kk} A_{kr}\partial_k\Big)\Big(\partial_r - \sum_j g^{jj} A_{jr}\partial_j\Big)\eta_i^*$$

From $\theta^{i,*} = \xi_1^{i,*} + \xi_2^{i,*}$, and based on the results of Eq. (9), we can conclude that

$$\bar{\theta}^i \simeq -\frac{g^{ii}}{2}\sum_{r\neq s}\Big(\partial_r - \sum_k g^{kk} A_{kr}\partial_k\Big)\Big(\partial_s - \sum_j g^{jj} A_{js}\partial_j\Big)\eta_i^*$$

The difference between a turbo solution and an MPM solution is evaluated if the difference between the parameters on M_0 is evaluated. If the MPM solution can be written as $\boldsymbol{\eta}_{MPM}^*$, the result is

$$\eta_{i,MPM}^* \simeq \eta_i^* + \frac{1}{2}\sum_{r\neq s}\Big(\partial_r - \sum_k g^{kk} A_{kr}\partial_k\Big)\Big(\partial_s - \sum_j g^{jj} A_{js}\partial_j\Big)\eta_i^*$$

The following theorem is derived from this result.

[Theorem 6] Assuming that $\boldsymbol{\eta}^*$ is the expectation of x_i based on a turbo decoding solution, and $\boldsymbol{\eta}_{MPM}^*$ is the expectation based on an MPM decoding solution, the difference between these is approximated as follows:

$$\boldsymbol{\eta}_{MPM}^* - \boldsymbol{\eta}^* \simeq \frac{1}{2}\sum_{r\neq s}\Big(\partial_r - \sum_k g^{kk} A_{kr}\partial_k\Big)\Big(\partial_s - \sum_j g^{jj} A_{js}\partial_j\Big)\boldsymbol{\eta}^*$$

The decoding error given by the above formula is related to the embedded *e*-curvature of the manifold $M(\boldsymbol{\theta})$.

4. Conclusions

In this paper, we have elucidated the mathematical structure of turbo codes from an information geometrical viewpoint. We believe that our results provide a mathematical framework for analyzing turbo codes and that more properties will be elucidated based on this framework.

This problem of turbo decoding can be regarded in a more general sense, as a problem of approximating the marginalization of a probability distribution which has a form shown in Eq. (3). The solution is approximated by dividing the whole problem into partial problems and partial information is integrated through iterative algorithm.

The same structure is found in low-density parity-check codes, the Bethe approximation of statistical physics, and the belief propagation of loopy Bayesian networks. Some differences exist in relation to the details of the algorithm, and completely identical results have not been obtained concerning solution stability and other properties, but since the basic structures are very similar, we believe it is not difficult to build the information geometrical framework for related problems. We have already extended the framework to the case of low-density parity-check codes [13]. Extensions for the Bethe approximation and belief propagation techniques are our future works.

Acknowledgments. We gratefully acknowledge Yoshiyuki Kabashima and Motohiko Isaka for their advice regarding this study. This work was supported in part by MEXT, Japan under a Grant-in-Aid for Scientific Research (16700227, 14084208).

REFERENCES

1. Berrou C, Glavieux A, Thitimajshima P. Near Shannon limit error-correcting coding and decoding: Turbo-codes. Proc IEEE Int Conf on Communications, p 1064–1070, Geneva, 1993.
2. Richardson T. The geometry of turbo-decoding dynamics. IEEE Trans Inf Theory 2000;46:9–23.
3. Gallager RG. Low density parity check codes. IRE Trans Inf Theory 1962;8:21–28.
4. MacKay DJC. Good error-correcting codes based on very sparse matrices. IEEE Trans Inf Theory 1999;45:399–431.
5. McEliece RJ, MacKay DJC, Cheng JF. Turbo decoding as an instance of Pearl's "belief propagation" algorithm. IEEE J Sel Areas Commun 1998;16:140–152.
6. Kabashima Y, Saad D. The TAP approach to intensive and extensive connectivity systems. In: Opper M, Saad D (editors). Advanced mean field methods—Theory and practice. MIT Press; 2001. p 65–84.
7. Amari S. Differential-geometrical methods in statistics. Lecture Notes in Statistics Vol. 28. Springer-Verlag; 1985.
8. Amari S, Nagaoka H. Methods of information geometry. AMS and Oxford University Press; 2000.
9. Kappen HJ, Wiegerinck WJ. Mean field theory graphical models. In: Opper M, Saad D (editors). Advanced mean field methods—Theory and practice. MIT Press; 2001. p 37–49.
10. Amari S, Ikeda S, Shimokawa H. Information geometry and mean field approximation: The α-projection approach. In: Opper M, Saad D (editors). Advanced mean field methods—Theory and practice. MIT Press; 2001. p 241–257.
11. Tanaka T. Information geometry of mean-field approximation. Neural Computation 2000;12:1951–1968.
12. Tanaka T. Information geometry of mean-field approximation. In: Opper M, Saad D (editors). Advanced mean field methods—Theory and practice. MIT Press; 2001. p 259–273.
13. Ikeda S, Tanaka T, Amari S. Information geometry of turbo codes and low-density parity-check codes. IEEE Trans Inf Theory 2004;50:1097–1114.

AUTHORS (from left to right)



Shiro Ikeda (member) received his B.E., M.E., and D.Eng. degrees in information physics from the University of Tokyo in 1991, 1993, and 1996. From 1996 to 2001, he was with RIKEN BSI, first as a special postdoctoral researcher of RIKEN and later as a researcher of JST. He was an associate professor at Kyushu Institute of Technology from 2001 to 2003, and has been an associate professor at the Institute of Statistical Mathematics since 2003. His research interests are in the areas of statistical signal processing, learning theory, and information geometry. He received the Best Research Award and Best Paper Award from the Japan Neural Network Society in 1999 and 2001, respectively.

Toshiyuki Tanaka (member) received his B.E., M.E., and D.Eng. degrees in electronics engineering from the University of Tokyo in 1988, 1990, and 1993. In 1993, he joined the Department of Electronics and Information Engineering, Tokyo Metropolitan University, and is currently an associate professor there. His research interests are in the interdisciplinary areas of information and communication theory, learning theory, information geometry, and statistical mechanics. He received the DoCoMo Mobile Science Award in 2002.

Shun-ichi Amari (fellow) graduated from the University of Tokyo in 1958, majoring in mathematical engineering, and received a D.Eng. degree from the University of Tokyo in 1963. He was an associate professor at Kyushu University, an associate professor and professor in the Department of Mathematical Engineering and Information Physics, University of Tokyo, and is now professor emeritus at the University of Tokyo. He is the Director of RIKEN BSI. He has been engaged in research in wide areas of mathematical engineering and applied mathematics, such as topological network theory, differential geometry of continuum mechanics, pattern recognition, mathematical foundations of neural networks, and information geometry. He served as President of the International Neural Network Society, Council member of the Bernoulli Society for Mathematical Statistics and Probability Theory, and is President-Elect of IEICE. He was founding Co-Editor-in-Chief of *Neural Networks*. He has been awarded the Japan Academy Award, IEEE Neural Networks Pioneer Award, IEEE Emanuel R. Poire Award, Neurocomputing best paper award, IEEE Signal Processing Society best paper award, and NEC C&C Prize, among many others.