# A Convergence Proof for the Turbo Decoder as an Instance of the Gauss-Seidel Iteration


John MacLaren Walsh
School of Elec. and Comp. Eng.
Cornell University
Ithaca, NY 14850 USA
Email: jmw56@cornell.edu

Phillip Regalia
Dept. of Elec. Eng. and Comp. Sci.
Catholic University of America
Washington DC 20064, USA
Email: regalia@cua.edu

C. Richard Johnson, Jr.
School of Elec. and Comp. Eng.
Cornell University
Ithaca, NY 14850 USA
Email: johnson@ece.cornell.edu



***Abstract*— Many previous attempts at analyzing the convergence behavior of turbo and iterative decoding, such as EXIT style analysis [2] and density evolution [3], 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 [4] and belief propagation [5], have been largely unsuccessful at showing convergence due to loops in the turbo coding graph. The information geometric attempts [6], [7], [8], [9], and [10], in turn have been inhibited by inability to efficiently describe intrinsic information extraction as an information projection. This paper recognizes turbo decoding as an instance of a Gauss-Seidel iteration on a particular nonlinear system of equations. This interpretation holds regardless of block length, and allows a connection to existing convergence results for nonlinear block Gauss Seidel iterations. We thus adapt existing convergence theory for the Gauss Seidel iteration to give sufficient conditions for the convergence of the turbo decoder that hold regardless of the block length.**


## I. Introduction

Along with being one of the most prominent communications inventions of the past decade, the introduction of turbo codes in [11] 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 [2] and density evolution [3], 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 [4] and [5], but these algorithms are only known to converge when the code graph has no loops, which is rarely true for turbo codes. The information geometric interpretation [7], [8], [6], [9], [10], while it provided an interesting way of describing certain portions of turbo decoding as information projections, seems to have been unable to completely describe turbo decoder convergence as well. This was largely due to the inability to describe the intrinsic information removal in the product densities as an information projection on an invariant set.

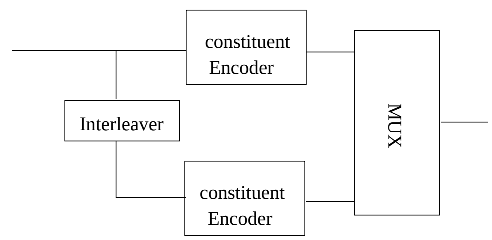


Fig. 1. A parallel concatenated turbo code. The MUX selects both the systematic and parity check bits from one of the component codes and just the parity check bits of the other. If puncturing is used, some of the parity check bits are never transmitted.

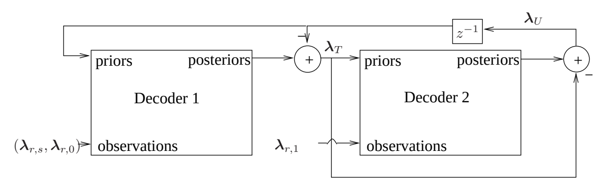


Fig. 2. The turbo decoder.

Indeed, despite a variety of these interpretations of the iterative decoding algorithm, many questions remain concerning where its fixed points lie and under which conditions it converges.

In this paper, we wish to somewhat demystify the iterative decoding algorithm by noting that it may be interpreted as a nonlinear block Gauss Seidel iteration on a particular system of nonlinear equations. While connection with the numerical analysis via fixed point iterations [12] have been noted before, the relation to the nonlinear block Gauss Seidel, and the convergence proof it can provide, seems to have been overlooked. This interpretation allows us to adapt existing convergence results for nonlinear block Gauss Seidel iterations to the situation encountered in turbo decoding. From it, we can gain sufficient conditions which are independent of block length which give a region of convergence for the turbo decoder.

## II. Notation and Review of the Turbo Decoder

In this section, we introduce our notation, which is heavily influenced by the information geometric analysis of turbo decoding [8] [7] [9] [10]. We also give a brief review of a description of the operation of the turbo decoder. For simplicity, we will consider the (parallel concatenated) turbo code structure as shown in Fig. 1 [11] and the iterative decoder structure as shown in Fig. 2. Let $N$ be the number of systematic bits in a block, and let $\mathbf{B}$ be the $2^N \times N$ matrix whose $i$th row is binary representation of the integer $i$.

Let $\mathbf{B}_i$ denote the $i$-th row of $\mathbf{B}$. If $p(\mathbf{B}_i)$ is a probability mass function over the outcomes $\{\mathbf{B}_i\}_{i=0}^{2^N-1}$, we will find it convenient to work with its logarithmic coordinates [7], [8], [10]

$$\theta(\mathbf{B}_i) = \log p(\mathbf{B}_i) - \log p(\mathbf{B}_0)$$

which, when listed in a vector $\boldsymbol{\theta}$ for $i$ from 0 to $2^N - 1$, are called the $\theta$ coordinates. This amounts to expressing $p(\mathbf{B}_i)$ as

$$p(\mathbf{B}_i) = \exp(\theta(\mathbf{B}_i) - \psi)$$

in which $\psi = \sum_i \exp[\theta(\mathbf{B}_i)]$ is a normalization constant to ensure that outcomes $p(\mathbf{B}_i)$ sum to one. One may show [7], [8] that a probability mass function $p(\mathbf{B}_i)$ is a product density (i.e., coincides with the product of its bitwise marginal functions) if and only if its $\theta$ coordinates become

$$\boldsymbol{\theta} = [\theta(\mathbf{B}_0), \theta(\mathbf{B}_1), \cdots, \theta(\mathbf{B}_{2^N-1})]^T = \mathbf{B}\boldsymbol{\lambda}$$

for some vector $\boldsymbol{\lambda}$, which may be identified as the log marginal ratios of the distribution:

$$\boldsymbol{\lambda}_i = \log[\mathsf{Pr}(\xi_i = 1)/\mathsf{Pr}(\xi_i = 0)]$$

Let $N_1$ be the number of parity check bits generated from the first component code (including punctured bits) and let $N_2$ be the number of parity check bits generated by the second component code (including punctured bits). Suppose that the vector of bitwise log likelihood ratios (LLRs) at the output of the channel (including zeros at the locations of the punctured bits) are $\boldsymbol{\lambda}_r \in \mathbb{R}^{N+N_1+N_2}$. Reorder these into a new vector $\boldsymbol{\lambda}'_r = (\boldsymbol{\lambda}_{r,s}, \boldsymbol{\lambda}_{r,0}, \boldsymbol{\lambda}_{r,1})$ so that all of the LLRs associated with the systematic bits are at the beginning of the vector $\boldsymbol{\lambda}_{r,s}$, followed by all of the LLRs associated with the parity check bits of the first code $\boldsymbol{\lambda}_{r,0}$, followed by all of the LLRs associated with the parity check bits of the second code $\boldsymbol{\lambda}_{r,1}$. Now, consider the codebook we are using. In particular, if we were to consider every possible value of the systematic bits, encoding each possibility and reordering it into the (systematic,parity check 1, parity check 2) order described above, and then stack these reordered codewords on top of each other, we would get a binary matrix of all the codewords, $\mathbf{C} \in \{0,1\}^{2^N \times (N+N_1+N_2)}$. This matrix would have the form

$$\mathbf{C} = [\mathbf{B}|\mathbf{C}_0|\mathbf{C}_1]$$

where now the $i$ row is the (systematic, parity 1, parity 2)-reordered codeword if the systematic block that was encoded was the binary representation of the integer $i$. We will assume that the all-zero codeword is contained in the code-books of each of the component decoders so that the first row of $\mathbf{C}$ is all zeros. This assumption is guaranteed, for example, if the two component codes are linear. Each of the component decoders could use some of the observed channel LLRs and its own codebook to generate a word-wise likelihood function. A component decoder may then be regarded as bitwise marginalizing its likelihood function weighted with the pseudo prior wordwise pmf obtained by multiplying the marginal bitwise prior probabilities that it was given as an input. The likelihood function that the first decoder uses, for instance, has $\theta$ coordinates [8] [7] [10]

$$\boldsymbol{\theta}_0 = [\mathbf{B}|\mathbf{C}_0] \begin{bmatrix} \boldsymbol{\lambda}_{r,s} \\ \boldsymbol{\lambda}_{r,0} \end{bmatrix}$$

while the likelihood function the second decoder uses has $\theta$ coordinates

$$\boldsymbol{\theta}_1 = [\mathbf{C}_1]\boldsymbol{\lambda}_{r,1}$$

Let the extrinsic information from the first component decoder be denoted by $\boldsymbol{\lambda}_T$ and let the extrinsic information from the second decoder be denoted by $\boldsymbol{\lambda}_U$. During the turbo decoder iterations, these extrinsic informations are obtained by marginalizing the word-wise densities whose $\theta$ coordinates are $\mathbf{B}\boldsymbol{\lambda}_U + \boldsymbol{\theta}_0$ and $\mathbf{B}\boldsymbol{\lambda}_T + \boldsymbol{\theta}_1$, respectively and then subtracting off intrinsic information. Thus, for any particular value of $\boldsymbol{\lambda}_T$ define a column vector $\mathbf{p}_2 \in [0,1]^N$ by

$$\mathbf{p}_2(\boldsymbol{\lambda}_T) = \mathbf{B}^T \frac{\exp\left(\mathbf{B}\boldsymbol{\lambda}_T + \boldsymbol{\theta}_1\right)}{\|\exp\left(\mathbf{B}\boldsymbol{\lambda}_T + \boldsymbol{\theta}_1\right)\|_1}$$

These are the bitwise marginal pseudo posterior probabilities that the second component decoder would output given pseudo prior log likelihood ratios $\boldsymbol{\lambda}_T$. Similarly, for any particular value of $\boldsymbol{\lambda}_U$ define a column vector $\mathbf{p}_1 \in [0,1]^N$ by

$$\mathbf{p}_1(\boldsymbol{\lambda}_U) = \mathbf{B}^T \frac{\exp\left(\mathbf{B}\boldsymbol{\lambda}_U + \boldsymbol{\theta}_0\right)}{\|\exp\left(\mathbf{B}\boldsymbol{\lambda}_U + \boldsymbol{\theta}_0\right)\|_1}$$

These are the bitwise marginal pseudo posterior probabilities that the first component decoder would output given pseudo prior log likelihood ratios $\boldsymbol{\lambda}_U$. In practice of course, $\mathbf{p}_1$ and $\mathbf{p}_2$ would be calculated by exploiting the Markovian structure of the channel observations restricted to each of the component codes using, for example, the forward backward algorithm [13]. However, for our purposes, this description suffices. Finally, define the column vector $\mathbf{p}_0 \in [0,1]^N$

$$\mathbf{p}_0(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) = \mathbf{B}^T \frac{\exp(\mathbf{B}(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T))}{\|\exp(\mathbf{B}(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T))\|_1}$$

These are the bitwise marginal probabilities associated with the extrinsic information values $\boldsymbol{\lambda}_U$ from the first component decoder together with the extrinsic information values $\boldsymbol{\lambda}_T$ from the second component decoder.

We shall also find it convenient to define the matrices $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2$ given by

$$\mathbf{P}_0 = \mathbf{B}^T \mathsf{Diag}\left[\frac{\exp(\mathbf{B}(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T))}{\|\exp(\mathbf{B}(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T))\|_1}\right]\mathbf{B}$$

$$\mathbf{P}_1 = \mathbf{B}^T \mathsf{Diag}\left[\frac{\exp(\mathbf{B}\boldsymbol{\lambda}_U + \boldsymbol{\theta}_0)}{\|\exp(\mathbf{B}\boldsymbol{\lambda}_U + \boldsymbol{\theta}_0)\|_1}\right]\mathbf{B}$$

$$\mathbf{P}_2 = \mathbf{B}^T \mathsf{Diag}\left[\frac{\exp(\mathbf{B}\boldsymbol{\lambda}_T + \boldsymbol{\theta}_1)}{\|\exp(\mathbf{B}\boldsymbol{\lambda}_T + \boldsymbol{\theta}_1)\|_1}\right]\mathbf{B}$$

Note then, that the entry at the $i$th row and $j$th column of $\mathbf{P}_0$, for example, is the probability that the $i$th and $j$th bits are one according to the measure whose $\theta$ coordinates are $\mathbf{B}(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T)$. The same holds true for $\mathbf{P}_1$ and $\mathbf{P}_2$ with $\mathbf{B}(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T)$ replaced with $\mathbf{B}\boldsymbol{\lambda}_U + \boldsymbol{\theta}_0$ and $\mathbf{B}\boldsymbol{\lambda}_T + \boldsymbol{\theta}_1$, respectively. Finally, we define $\mathbf{Q}_i = \mathbf{P}_i - \mathbf{p}_i\mathbf{p}_i^T$ for $0 \leq i \leq 2$.

We shall show in a later section that the turbo decoder may be interpreted as an attempt to solve the nonlinear system of equations

$$\begin{aligned} \mathbf{p}_0(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) - \mathbf{p}_1(\boldsymbol{\lambda}_U) &= 0 \\ \mathbf{p}_0(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) - \mathbf{p}_2(\boldsymbol{\lambda}_T) &= 0 \end{aligned}$$

simultaneously using the block nonlinear Gauss Seidel iteration. First, however, we will review the specifics of the Gauss Seidel iteration method for solving a system of equations.

## III. The Gauss-Seidel Iteration

The Gauss-Seidel iteration is an iterative method for solving a system of equations. It has been discussed most widely for linear systems of equations [14] pp. 480-483, [15] pp. 251-257, and [16] pp. 510-511, but can be generalized to nonlinear systems of equations as well [17] pp. 131-133, pp. 185-197.

For the nonlinear block Gauss Seidel iterations [18] pp. 225, in particular, we would like to solve the system of nonlinear equations

$$\mathtt{F}(\mathbf{x}) = \mathbf{0} \tag{1}$$

Where, we assume in this case that $\mathtt{F} : \mathbb{R}^N \rightarrow \mathbb{R}^N$. We divide the vector $\mathbf{x}$ into halves $\mathbf{x} = (\mathbf{x}_0, \mathbf{x}_1)$ and we divide the system of equations (1) into two halves

$$\mathtt{F}_0(\mathbf{x}_0, \mathbf{x}_1) = \mathbf{0} \tag{2}$$
$$\mathtt{F}_1(\mathbf{x}_0, \mathbf{x}_1) = \mathbf{0} \tag{3}$$

After picking an initial estimate for $\mathbf{x}_1$, we use this estimate to solve (2) for $\mathbf{x}_0$. We then use this value of $\mathbf{x}_0$ to solve for a new value of $\mathbf{x}_1$ using (3), and the procedure iterates.

$$\mathbf{x}_0^{(k+1)} = \mathbf{x}_0 \text{ such that } \mathtt{F}_0(\mathbf{x}_0, \mathbf{x}_1^{(k)}) = \mathbf{0} \tag{4}$$
$$\mathbf{x}_1^{(k+1)} = \mathbf{x}_1 \text{ such that } \mathtt{F}_1(\mathbf{x}_0^{(k+1)}, \mathbf{x}_1) = \mathbf{0} \tag{5}$$

Of course, such an iteration does not always converge to the solution to the system of equations: one needs the system which you want to solve to satisfy certain conditions [1]. We will investigate what the conditions mean for the turbo decoder, but first we show how the turbo decoder can be viewed as a block nonlinear Gauss Seidel iteration.

## IV. Connection to the Turbo Decoder

Recall that we had one interpretation of the turbo decoder to be an iterative algorithm bent on solving the system of equations

$$\mathbf{p}_0(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) - \mathbf{p}_1(\boldsymbol{\lambda}_U) = \mathbf{0} \tag{6}$$
$$\mathbf{p}_0(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) - \mathbf{p}_2(\boldsymbol{\lambda}_T) = \mathbf{0} \tag{7}$$

The algorithm operates by solving (6) for $\boldsymbol{\lambda}_T$ given a initial $\boldsymbol{\lambda}_U$, and then solving (7) for a new $\boldsymbol{\lambda}_U$ given $\boldsymbol{\lambda}_T$. This $\boldsymbol{\lambda}_U$ is then used to solve for a new $\boldsymbol{\lambda}_T$ in (6) again, and the algorithm repeats, ideally until a solution to the system is found. In the case that a solution is found, it is easy to see from the system above that $\mathbf{p}_0 = \mathbf{p}_1 = \mathbf{p}_2$, and thus the pseudo-posteriors from the two decoders agree.

Now, to notice that this is a block version of Gauss-Seidel, just write the function

$$\begin{aligned} \mathtt{F}_0(\boldsymbol{\lambda}_T, \boldsymbol{\lambda}_U) &= \mathbf{p}_0(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) - \mathbf{p}_1(\boldsymbol{\lambda}_U) \\ \mathtt{F}_1(\boldsymbol{\lambda}_T, \boldsymbol{\lambda}_U) &= \mathbf{p}_0(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) - \mathbf{p}_2(\boldsymbol{\lambda}_T) \end{aligned}$$

Now, just note that the turbo decoder iterations have the form

$$\boldsymbol{\lambda}_T^{(k+1)} = \boldsymbol{\lambda}_T \text{ such that } \mathtt{F}_0(\boldsymbol{\lambda}_T, \boldsymbol{\lambda}_U^{(k)}) = 0 \tag{8}$$
$$\boldsymbol{\lambda}_U^{(k+1)} = \boldsymbol{\lambda}_U \text{ such that } \mathtt{F}_1(\boldsymbol{\lambda}_T^{(k+1)}, \boldsymbol{\lambda}_U) = 0 \tag{9}$$

This is exactly the same form as (4) and (5), so the turbo decoder is a Gauss Seidel iteration in an attempt to solve the system of equations (6) and (7) simultaneously.

We must now prove that this iteration yields well defined solutions. Indeed, we can prove that it does for $\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T$ which do not result in decoder posteriors that have bitwise marginal probabilities that are on the boundary of $[0, 1]^N$. We have the following theorem.

**Thm. 1** (Uniqueness)**:** Given a $\boldsymbol{\lambda}_U$ as a pseudo posterior, the extrinsic information output $\boldsymbol{\lambda}_T$ of the first component decoder is the only solution to (6) for that $\boldsymbol{\lambda}_U$, and we can write a function $\boldsymbol{\lambda}_T^0(\boldsymbol{\lambda}_U)$ that gives the only solution to (6) for each $\boldsymbol{\lambda}_U \in \mathbb{R}$ such that $(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T) \in \mathbb{R}^N$ (i.e., $\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T$ is finite).

*Proof:* Consider the derivative of (6) with respect to $\boldsymbol{\lambda}_T$

$$\begin{aligned} \nabla_{\boldsymbol{\lambda}_T}\mathtt{F}_0 &= \nabla_{\boldsymbol{\lambda}_T}\mathbf{B}^T \frac{\exp(\mathbf{B}(\boldsymbol{\lambda}_T + \boldsymbol{\lambda}_U))}{\|\exp(\mathbf{B}(\boldsymbol{\lambda}_T + \boldsymbol{\lambda}_U))\|_1} \\ &= \mathbf{B}^T \mathsf{Diag}\left[\frac{\exp(\mathbf{B}(\boldsymbol{\lambda}_T + \boldsymbol{\lambda}_U))}{\|\exp(\mathbf{B}(\boldsymbol{\lambda}_T + \boldsymbol{\lambda}_U))\|_1}\right]\mathbf{B} - \\ &\quad \mathbf{B}^T \frac{\exp(\mathbf{B}(\boldsymbol{\lambda}_T + \boldsymbol{\lambda}_U))}{\|\exp(\mathbf{B}(\boldsymbol{\lambda}_T + \boldsymbol{\lambda}_U))\|_1} \\ &\quad \frac{\exp(\mathbf{B}(\boldsymbol{\lambda}_T + \boldsymbol{\lambda}_U))^T}{\|\exp(\mathbf{B}(\boldsymbol{\lambda}_T + \boldsymbol{\lambda}_U))\|_1}\mathbf{B} \\ &= \mathbf{P}_0 - \mathbf{p}_0\mathbf{p}_0^T = \mathbf{Q}_0 \\ &= \mathsf{Diag}[\mathbf{p}_0]\mathsf{Diag}[\mathbf{1} - \mathbf{p}_0] \end{aligned}$$

We can thus see that $\nabla_{\boldsymbol{\lambda}_T}[\mathbf{p}_0 - \mathbf{p}_1]$ is full rank unless $\mathbf{p}_0$ is on the boundary (i.e. at least one of the probabilities are one or zero). We know that a component decoder solves (6) by definition so we have at least one such $(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T)$. The implicit function theorem [19] pp. 224, [20] then tells us that we can have a function $\boldsymbol{\lambda}_T^0(\boldsymbol{\lambda}_U)$ such that

$$\mathbf{p}_0(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T^0(\boldsymbol{\lambda}_U)) - \mathbf{p}_1(\boldsymbol{\lambda}_U) = 0$$

for all $\boldsymbol{\lambda}_U \in \mathbb{R}^N$ that do not give rise to pseudo-posteriors on the boundary.

**Cor. 1:** Given a $\boldsymbol{\lambda}_T$ as a pseudo posterior, the extrinsic information output of a component decoder $\boldsymbol{\lambda}_U$ is the only solution to (7) for that $\boldsymbol{\lambda}_T$ as long as $(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T) \in \mathbb{R}^N$ (i.e., $\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T$ is finite), and we can write a function $\boldsymbol{\lambda}_U^0(\boldsymbol{\lambda}_T)$ that gives the unique solution to (7) for each of these $\boldsymbol{\lambda}_T$s.

## V. Convergence Theorem

The convergence of nonlinear Gauss Seidel methods received some significant attention in the numerical analysis literature during the early 1970s. Relevant references include [21], [22], and [1], from which our major result is adapted. Note that here we will use the componentwise ordering, so that $\mathbf{x} \leq \mathbf{y} \iff x_i \leq y_i \forall i$.

**Thm. 2** (Region of Convergence for the Turbo Decoder)**:** Define the measures $\mathsf{q}$ whose $\theta$ coordinates are $\mathbf{B}\boldsymbol{\lambda}_U + \boldsymbol{\theta}_0$, $\mathsf{r}$ whose $\theta$ coordinates are $\mathbf{B}\boldsymbol{\lambda}_T + \boldsymbol{\theta}_1$, and $\mathsf{p}$ whose $\theta$ coordinates are $\mathbf{B}(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T)$. Define the set $\mathscr{D} \subseteq \mathbb{R}^N \times \mathbb{R}^N$ equal to

$$
\begin{gathered}
(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) \in \mathbb{R}^N \times \mathbb{R}^N | \exists \mathbf{x} \in \mathbb{R}^{2N} \text{ such that} \forall i \in \{1, \ldots, N\} \\
x_i \mathsf{p}\left[\xi_i = 1\right] \mathsf{p}\left[\xi_i = 0\right] > \\
\sum_{j \neq i} x_{N+j} |\mathsf{q}[\xi_i = 1]\mathsf{q}[\xi_j = 1] - \mathsf{q}\left[\xi_i = 1 \cap \xi_j = 1\right]| \\
+ x_{N+i} |\mathsf{p}[\xi_i = 1]\mathsf{p}[\xi_i = 0] - \mathsf{q}[\xi_i = 1]\mathsf{q}[\xi_i = 0]| \\
\text{and} \\
x_{N+i} \mathsf{p}\left[\xi_i = 1\right] \mathsf{p}\left[\xi_i = 0\right] > \\
\sum_{j \neq i} x_j |\mathsf{r}[\xi_i = 1]\mathsf{r}[\xi_j = 1] - \mathsf{r}\left[\xi_i = 1 \cap \xi_j = 1\right]| \\
+ x_i |\mathsf{p}[\xi_i = 1]\mathsf{p}[\xi_i = 0] - \mathsf{r}[\xi_i = 1]\mathsf{r}[\xi_i = 0]| \\
\text{and} \\
\mathsf{q}\left[\xi_i = 1\right] \mathsf{q}\left[\xi_j = 1\right] \leq \mathsf{q}\left[\xi_i = 1 \cap \xi_j = 1\right] \ \forall j \neq i, \text{ and} \\
\mathsf{r}\left[\xi_i = 1\right] \mathsf{r}\left[\xi_j = 1\right] \leq \mathsf{r}\left[\xi_i = 1 \cap \xi_j = 1\right] \ \forall j \neq i, \text{ and} \\
\mathsf{p}\left[\xi_i = 1\right] \mathsf{p}\left[\xi_i = 0\right] \leq \mathsf{q}\left[\xi_i = 1\right] \mathsf{q}\left[\xi_i = 0\right], \text{and} \\
\mathsf{p}\left[\xi_i = 1\right] \mathsf{p}\left[\xi_i = 0\right] \leq \mathsf{r}\left[\xi_i = 1\right] \mathsf{r}\left[\xi_i = 0\right] \}
\end{gathered}
$$

Consider any open set $\mathscr{C} \subseteq \mathscr{D}$. Then, given an initialization $\boldsymbol{\lambda}^{(0)} = (\boldsymbol{\lambda}_U^{(0)}, \boldsymbol{\lambda}_T^{(0)}) \in \mathscr{C}$ such that the $\mathbf{a}$ and $\mathbf{b}$ defined by

$$
\begin{aligned}
\mathbf{x} &= \mathtt{F}(\boldsymbol{\lambda}^{(0)}) \\
a_i &= \min(\mathbf{x}_i, 0) \quad \forall i \in \{1, \ldots, N\} \\
b_i &= \max(\mathbf{x}_i, 0) \quad \forall i \in \{1, \ldots, N\}
\end{aligned}
$$

are in $\mathtt{F}(\mathscr{C})$, and the set $\{\mathbf{x} | \mathtt{F}^{-1}(\mathbf{a}) \leq \mathbf{x} \leq \mathtt{F}^{-1}(\mathbf{b})\} \subseteq \mathscr{C}$ the turbo decoder converges to a unique fixed point in $\mathscr{C}$.

*Proof:* The outline of our proof parallels that of [1]:

- The conditions for $\mathscr{D}$ make the Jacobian of the nonlinear system of equations

$$
\mathtt{F}(\boldsymbol{\lambda} = \left[ \begin{array}{c} \boldsymbol{\lambda}_U \\ \boldsymbol{\lambda}_T \end{array} \right]) = \left[ \begin{array}{c} \mathbf{p}_0 - \mathbf{p}_1 \\ \mathbf{p}_0 - \mathbf{p}_2 \end{array} \right]
$$

  strictly diagonally dominant anywhere within $\mathscr{D}$. This, in turn makes the Jacobian an M-matrix [23].
- Since the Jacobian is an M-matrix everywhere within $\mathscr{D}$, it follows that $\mathtt{F}$ is an M-function [1], [22].
- Since $\mathtt{F}$ is an M-function everywhere in $\mathscr{D}$, the Gauss Seidel block iteration on $\mathtt{F}$ converges. Hence, the turbo decoder converges.

Begin by determining the Jacobian of $\mathtt{F}$ with respect to $\boldsymbol{\lambda}_U$ and $\boldsymbol{\lambda}_T$. To facilitate this, define $\boldsymbol{\lambda} = \left[\boldsymbol{\lambda}_T^T, \boldsymbol{\lambda}_U^T\right]^T$

$$
\nabla_{\boldsymbol{\lambda}} \mathtt{F} = \left[ \begin{array}{cc} \mathbf{Q}_0 & \mathbf{Q}_0 - \mathbf{Q}_1 \\ \mathbf{Q}_0 - \mathbf{Q}_2 & \mathbf{Q}_0 \end{array} \right]
$$

Next, note that the block diagonal components of this matrix are indeed diagonal matrices, since the pmf whose $\theta$ coordinates are $\mathbf{B}(\boldsymbol{\lambda}_U + \boldsymbol{\lambda}_T)$ is a product measure. Thus we have that $\mathbf{Q}_0 = \mathsf{Diag}[\mathbf{p}_0]\mathsf{Diag}[1 - \mathbf{p}_0]$. Now, suppose $(\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T) \in \mathscr{D}$ and also that they are finite (i.e. $\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T \in \mathbb{R}^N$), so that $\mathbf{p}_0$ and $1 - \mathbf{p}_0$ have no elements which are equal to zero so that the diagonal elements of $\nabla_{\boldsymbol{\lambda}} \mathtt{F}$ are all positive. Then, the last four conditions for $\mathscr{D}$, imply that the off-diagonal components of $\nabla_{\boldsymbol{\lambda}} \mathtt{F}$ are non-positive. The other two conditions for $\mathscr{D}$, together with the positivity of the diagonal components and the negativity of the off diagonal components then imply that $\nabla_{\boldsymbol{\lambda}} \mathtt{F}$ is generalized diagonally dominant, which in turns implies, together with its positive diagonal elements and non-positive off diagonal elements, that $\nabla_{\boldsymbol{\lambda}} \mathtt{F}$ is an M-matrix (see [23] pp. 205 Theorem 6.5). This means then, that the function $L(\boldsymbol{\lambda}, \mathbf{x}) = (\nabla_{\boldsymbol{\lambda}} \mathtt{F})\mathbf{x}$ is inverse isotone with respect to $\mathbf{x}$ for any fixed $\boldsymbol{\lambda} \in \mathscr{D}$, so that $L(\boldsymbol{\lambda}, \mathbf{x}_0) \leq L(\boldsymbol{\lambda}, \mathbf{x}_1) \implies \mathbf{x}_0 \leq \mathbf{x}_1$. Furthermore, $L(\boldsymbol{\lambda}, \mathbf{x}_0)$ for a fixed $\boldsymbol{\lambda} \in \mathscr{D}$ is also off diagonally antitone, so that, if we define the $i$th component of $L(\boldsymbol{\lambda}, \mathbf{x})$ by $L_i(\boldsymbol{\lambda}, \mathbf{x})$, and if we consider $\mathbf{y}, \mathbf{x}$ such that $\mathbf{x} \leq \mathbf{y}$ and $y_i = x_i$, then we have $L_i(\boldsymbol{\lambda}, \mathbf{y}) \geq L_i(\boldsymbol{\lambda}, \mathbf{x})$. Differentiability of $\mathtt{F}$, and thus the theory of best affine approximations, gives this same property to $\mathtt{F}(\boldsymbol{\lambda})$ at any point in the interior of $\mathscr{D}$ [1]. Thus, on the interior of $\mathscr{D}$, $\mathtt{F}(\boldsymbol{\lambda})$ is inverse isotone and off-diagonally antitone.

Now, consider the function which relates $\boldsymbol{\lambda}^{(k+1)} = [\boldsymbol{\lambda}_U^{(k+1)}; \boldsymbol{\lambda}_T^{(k+1)}]$ to $\boldsymbol{\lambda}^{(k)} = [\boldsymbol{\lambda}_U^{(k)}; \boldsymbol{\lambda}_T^{(k)}]$ and whose solution

$$
\mathsf{G}(\boldsymbol{\lambda}^{(k+1)}, \boldsymbol{\lambda}^{(k)}) = 0
$$

describes the evolution of the turbo decoder. In particular, since the turbo decoder is a block Gauss Seidel iteration on $\mathtt{F} = 0$, we have

$$
\mathsf{G}(\boldsymbol{\lambda}^{(k+1)}, \boldsymbol{\lambda}^{(k)}) = \left[ \begin{array}{c} \mathbf{p}_0(\boldsymbol{\lambda}_U^{(k)}, \boldsymbol{\lambda}_T^{(k+1)}) - \mathbf{p}_1(\boldsymbol{\lambda}_U^{(k)}) \\ \mathbf{p}_0(\boldsymbol{\lambda}_U^{(k+1)}, \boldsymbol{\lambda}_T^{(k+1)}) - \mathbf{p}_2(\boldsymbol{\lambda}_T^{(k+1)}) \end{array} \right]
$$

To see that the turbo decoder solves this iteration, just recall the Jacobian of $\mathtt{F}$, and note that the block diagonal components which were $\mathsf{Diag}[\mathbf{p}_0]\mathsf{Diag}[\mathbf{1} - \mathbf{p}_0]$ are injective for any real valued $\boldsymbol{\lambda}_U, \boldsymbol{\lambda}_T$, so that there are no zero or one bitwise marginal probabilities. The implicit function theorem applied to the top half of $\mathsf{G} = \mathbf{0}$, then gives that $\boldsymbol{\lambda}_T^{(k+1)}$ can be written as a function of $\boldsymbol{\lambda}_U^{(k)}$ and is unique for any particular $\boldsymbol{\lambda}_U^{(k)}$, so that no two $\boldsymbol{\lambda}_T^{(k+1)}$s satisfy the top half of $\mathsf{G} = \mathbf{0}$. Repeating this argument of the bottom half of $\mathsf{G} = \mathbf{0}$ shows that $\boldsymbol{\lambda}_U^{(k+1)}$ can be written as a function of $\boldsymbol{\lambda}_T(k+1)$ and is unique for that $\boldsymbol{\lambda}_T^{(k+1)}$, which shows that it is unique given $\boldsymbol{\lambda}_U^{(k)}$. Thus, we have an equation whose repeated solution describes the behavior of the turbo decoder. We have

$$
\mathsf{G}(\boldsymbol{\lambda}^{(k+1)}, \boldsymbol{\lambda}^{(k)}) = \left[ \begin{array}{c} \mathtt{F}_0(\boldsymbol{\lambda}_T^{(k+1)}, \boldsymbol{\lambda}_U^{(k)}) \\ \mathtt{F}_1(\boldsymbol{\lambda}_T^{(k+1)}, \boldsymbol{\lambda}_U^{(k+1)}) \end{array} \right]
$$

This form makes it easy to see that, given that $\mathtt{F}$ is off diagonal antitone and inverse isotone on the interior of $\mathscr{D}$, $\mathsf{G}(\cdot, \mathbf{x})$ is is inverse isotone for a fixed $\mathbf{x}$, and $\mathsf{G}(\mathbf{y}, \cdot)$ is antitone for a fixed

$\mathbf{x}$ (all considered in the interior of $\mathscr{D}$). This in turn implies the implicit function $\mathsf{h}$, which satisfies

$$\mathsf{G}(\mathsf{h}(\mathbf{x}), \mathbf{x}) = 0$$

is isotonic. To see this, consider $\mathbf{x}_0 \leq \mathbf{x}_1$. Now, antitonicity of $\mathsf{G}$ gives

$$\mathsf{G}(\mathsf{h}(\mathbf{x}_1), \mathbf{x}_0) \geq \mathsf{G}(\mathsf{h}(\mathbf{x}_1), \mathbf{x}_1) = 0 = \mathsf{G}(\mathsf{h}(\mathbf{x}_0), \mathbf{x}_0)$$

Now, inverse isotonicity of $\mathsf{G}$ means then that

$$\mathsf{G}(\mathsf{h}(\mathbf{x}_0), \mathbf{x}_0) \leq \mathsf{G}(\mathsf{h}(\mathbf{x}_1), \mathbf{x}_0) \implies \mathsf{h}(\mathbf{x}_0) \leq \mathsf{h}(\mathbf{x}_1)$$

We now have a function $\mathsf{h}$ that describes the turbo decoder through

$$\boldsymbol{\lambda}^{(k+1)} = \mathsf{h}(\boldsymbol{\lambda}^{(k)})$$

Of course, we could have written this function from the turbo decoder iterations when we started, but defining it this way made it easy to see some of its properties, namely that it is isotone in $\mathscr{D}$.

Now, consider some initialization $\boldsymbol{\lambda}^{(0)}$ in the interior of $\mathscr{D}$, and set $\mathbf{x} = \mathtt{F}(\boldsymbol{\lambda}^{(0)})$.

$$\begin{aligned} a_i &= \min(\mathbf{x}_i, 0),\ b_i = \max(\mathbf{x}_i, 0) \quad \forall i \in \{1, \ldots, N\} \\ \mathbf{y} &= \mathtt{F}^{-1}(\mathbf{a}),\ \mathbf{z} = \mathtt{F}^{-1}(\mathbf{b}) \end{aligned}$$

The inverse function $\mathtt{F}^{-1}$ is guaranteed to exist within the interior of $\mathscr{D}$, because $\nabla_{\boldsymbol{\lambda}}\mathtt{F}$ is an M-matrix and thus non-singular ([23] lemma 6.1 page 202). This, in turn, allows the inverse function theorem to be applied ([19] Theorem 9.24 page 221). Then we have both $\mathbf{a} \leq \mathbf{0} \leq \mathbf{b}$ and by inverse isotonicity of $\mathtt{F}$, we have $\mathbf{y} \leq \boldsymbol{\lambda}^{(0)} \leq \mathbf{z}$. Isotonicity of $\mathsf{h}$ then implies that the sequences defined by

$$\mathbf{y}^{(k+1)} = \mathsf{h}(\mathbf{y}^{(k)}),\ \mathbf{z}^{(k+1)} = \mathsf{h}(\mathbf{z}^{(k)})$$

will obey $\mathbf{y}^{(k)} \leq \boldsymbol{\lambda}^{(k)} \leq \mathbf{z}^{(k)}$, $\mathbf{y}^{(k+1)} \geq \mathbf{y}^{(k)}$, and $\mathbf{z}^{(k+1)} \leq \mathbf{z}^{(k)}$. This shows that $\mathbf{y}^{(k)}, \mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k)}$ all converge to fixed points. Furthermore, $\mathtt{F}$ is injective because it was inverse isotone, which is because it was an M-function ([1] Theorem 3.2(a) page 511). The injectivity of $\mathtt{F}$ then implies that this fixed point must be unique in $\mathscr{C}$. This proves the theorem.

## VI. Conclusions

Existing methods at analyzing the turbo decoder have all been limited by assumptions, for example large block lengths or graphs without cycles, which can sometimes not be true in practice. Given the need for describing the turbo decoder's convergence behavior in these cases, in this paper we characterized the turbo decoder as the iterative solution of a system of nonlinear equations. We then were able to recognize the form of the iterative solution as a nonlinear block Gauss Seidel iteration. This connection allowed us to adapt existing convergence results from the numerical analysis literature to the turbo decoder. We were then able to prove a theorem which gave conditions for the convergence of the turbo decoder regardless of the block length.


## Acknowledgments

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. 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.


## References


[1] Werner C. Rheinboldt, "On classes of n-dimensional nonlinear mappings generalizing several types of matrices," in *Proc. Symposium on the Numerical Solution of Partial Differential Equations. II*. 1970, pp. 501–546, Academic Press.

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

[3] 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.

[4] 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.

[5] 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.

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

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

[8] 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.

[9] B. Muquest, P. Duhamel, and M. de Courville, "Geometrical interpretations of iterative 'turbo' decoding," in *Proceedings ISIT*, June 2002.

[10] J. Walsh, P. Regalia, and C. R. Johnson, Jr., "A refined information geometric interpretation of turbo decoding," in *To appear at ICASSP '05*, Mar. 2005.

[11] 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.

[12] P. Moqvist and T. M. Aulin, "Turbo-decoding as a numerical analysis problem," in *Proc. IEEE Internation Symposium on Information Theory, 2000.*, June 2000, p. 485.

[13] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate.," *IEEE Trans. Inform. Theory*, vol. 20, Mar. 1974.

[14] Forman S. Acton, *Numerical Methods that Work*, Harper & Row Publishers, 1970.

[15] James L. Buchanan and Peter R. Turner, *Numerical Methods and Analysis*, McGraw Hill, Inc., 1992.

[16] Gene H. Golub and Charles F. Van Loan, *Matrix Computations 3rd Edition*, The Johns Hopkins University Press, 1996.

[17] Dimitri P. Bertsekas and John N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, 1997.

[18] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, 1970.

[19] Walter Rudin, *Principles of Mathematical Analysis 3rd Edition*, McGraw Hill, 1976.

[20] Robert S. Strichartz, *The Way of Analysis*, Jones & Bartlett Publishers, 2000.

[21] Werner C. Rheinboldt, "On m-functions and their application to nonlinear gauss-seidel iterations and network flows," *Journal Mathematical Analysis and Applications*, pp. 274–307, 1971.

[22] Jorge J. Moré, "Nonlinear generalizations of matrix diagonal dominance with application to gauss-seidel iterations," *SIAM Journal on Numerical Analysis*, vol. 9, pp. 357–378, June 1972.

[23] Owe Axelsson, *Iterative Solution Methods*, Cambridge University Press, 1994.