

An Analytical Comparison of EXIT and Variance Transfer (VT) Tools for Iterative Decoder Analysis

David P. Shepherd, Matt Ruan, Mark C. Reed, Zhenning Shi

Wireless Signal Processing Program

National ICT Australia

Australian National University

Locked Bag 8001 Canberra ACT 2601, Australia

Ph: +61 2 6125 6240 Fax: +61 2 6230 6121

Email: {david.shepherd,matt.ruan}@anu.edu.au, {mark.reed, zhenning.shi}@nicta.com.au

Abstract— This paper investigates the relationship between the Extrinsic Information Transfer (EXIT) analysis method and the Variance Transfer (VT) analysis method as used to analyze turbo decoding. The two methods are compared numerically and analytically. A function is derived to closely approximate the relationship between the mutual information and the input/output variance. Upper and lower bounds are derived to show the error in this approximation is small. The result of this work allows for easy translation between the EXIT and VT methods. Furthermore, three methods of measuring mutual information are compared and motivations for use of each are discussed.

I. INTRODUCTION

Powerful channel coding is possible today with the utilization of turbo codes [1]. The decoding technique iterates between two decoder blocks where the data exchange is separated by an interleaving function. The original technique can produce performance which is less than 1dB from capacity and subsequent results can get to within a small fraction of a decibel from capacity.

Three main methods exist to analyze the performance of turbo codes[1]. One method is known as EXIT analysis and determines a function through simulation which describes the change in mutual information between the input and the output of the decoder [2]. This method typically assumes a Gaussian distributed input signal, however the output signal does not need to be Gaussian distributed.

The second method that was published before ten Brink's paper is known as the Variance Transfer (VT) method [3,4]. This technique essentially determines by simulation the input-output relationship of the decoding unit by comparing the a-priori input noise variance to the a-posteriori output noise variance, for a fixed E_b/N_0 .

The third method is density evolution. Divsalar et al analyzed iterative systems by tracking the density of extrinsic information in [5]. This technique uses the signal-to-noise ratio as a metric and is useful for optimizing LDPC codes.

The open question which is addressed in this paper is how can one compare these first two techniques and what is fundamentally different between the two analysis methods. There have been several papers that have compared the EXIT and VT techniques [2, 6]. However, these papers only compared the analysis techniques numerically by utilizing simulation,

whereas in our paper we utilize analytical techniques to compare the two methods. Hagenauer compared the EXIT chart with the soft bit transfer (SOBIT) chart (also known as a fidelity chart) in [7]. The SOBIT chart however, uses the expectation of the soft bits $\lambda_E = \tanh(E/2)$, whereas the VT chart traces the mean squared error (MSE) of the soft bits.

II. SYSTEM DESCRIPTION

A. Logarithmic Likelihood Ratios and Extrinsic Information

In this paper we consider the iterative decoder for systematic parallel concatenated codes (PCC). For the systematic bits x , the soft output of each constituent decoder is the logarithmic likelihood ratio (LLR) $D = Z + A + E$ [2], where Z is the LLR of the received signal from the AWGN channel, E is the extrinsic information generated in the decoder and A is the *a priori* input to the decoder

$$A = \mu_A \cdot x + n_A \quad (1)$$

where $n_A \sim \mathcal{N}(0, \sigma_A^2)$ and $\mu_A = \sigma_A^2/2$. Capital letters are used to represent LLR's.

B. Input and Output Variance

With the VT method, variance is defined as [4]

$$\sigma_{in/out}^2 = \text{E} \left\{ \left(\tanh \left(\frac{\lambda}{2} \right) - x \right)^2 \right\}, \quad (2)$$

where λ is equal to A or E for input and output variance respectively. We assume the probability density functions of A and E are Gaussian.

C. Mutual Information

The mutual information I_A of the *a priori* input and I_E of the extrinsic output are defined as in [2], where the assumption is that the *a priori* input is an independent Gaussian random variable. No assumptions are made on the distribution of the extrinsic outputs.

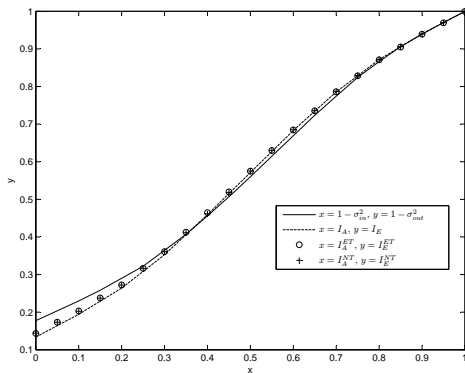


Fig. 1. EXIT and VT Characteristics of a symmetric PCC (rate $\frac{1}{2}$ memory 4 (037,021)) at $\frac{E_b}{N_0} = 0.8\text{dB}$, showing different methods of measuring mutual information

D. Simplified Mutual Information

Hagenauer [7] described two methods of measuring mutual information of the extrinsic output that do not involve determination of the extrinsic output distributions. We will refer to these methods as the Ergodicity Theorem (ET) and Nonlinear Transform (NT) methods, in which mutual information is calculated respectively as

$$I_E^{\text{ET}} = 1 - \frac{1}{N} \sum_{n=1}^N \log_2(1 + e^{-x E}) \quad (3)$$

$$I_E^{\text{NT}} = 1 - \frac{1}{N} \sum_{n=1}^N H_b\left(\frac{e^{+|E|/2}}{e^{+|E|/2} + e^{-|E|/2}}\right) \quad (4)$$

where $H_b(p) = -p \log_2 p - (1-p) \log_2(1-p)$.

E. Motivation

A major motivation for use of the EXIT chart is its ability to predict the convergence behavior of a turbo code without computationally demanding simulations. The VT, ET and NT methods could potentially be used as an alternate method of monitoring decoding behaviour. All methods have linear complexity of order n where n is the block length. We determine the complexity by timing the simulation run-time and averaging the results over the range of σ_E values tested. These results are shown in Figure 2. The VT method is in this case less complex than any other method, while the NT method has the highest complexity. Since ten Brinks' method uses histogram measurements the n samples of E are sorted into bins and further calculations are dependent only upon the number of bins used in the histogram. This explains why the other methods are less complex when the block length is small. Since all methods have linear complexity, increasing or reducing the bin size would simply lower or raise (respectively) the line $y = 1$. While the NT method clearly has the highest complexity it may be useful since knowledge of the systematic bits x is not required. I_E can therefore be determined after each iteration to estimate bit error rates for early termination decisions. These results must be interpreted with care since they are of course dependent upon the implementation.

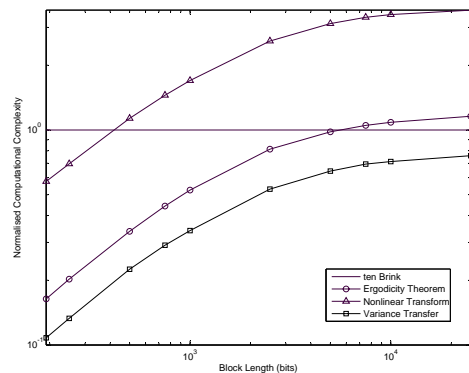


Fig. 2. Comparison of computational complexity per bit for each method of calculating I_E and σ_{out}^2 , normalized relative to ten Brinks' method

F. Computing the Transfer Chart

To compute the transfer functions $I_E = T(I_A, E_b/N_0)$ and $\sigma_{out}^2 = T(\sigma_{in}^2, E_b/N_0)$ we generated the *a priori* input A using (1) over a range of values of σ_A . The values of I_A and σ_{in}^2 were calculated and the corresponding I_E and σ_{out}^2 were obtained through Monte Carlo simulation in which A was applied as the *a priori* input to the decoder. When computing the transfer chart for a constituent decoder we used a block length of 10^4 bits, which was determined in [2] to be sufficiently long to suppress tail effects.

III. COMPARISON OF EXIT AND VT CHARTS

It is interesting to note the symmetry of EXIT and VT charts and that if one of the charts is flipped on the $y = 1 - x$ axis that the charts are very similar. Figure 1 shows EXIT and the “flipped” VT characteristics, $(1 - \sigma_{in}^2)$ vs $(1 - \sigma_{out}^2)$, of the code discussed in Section II-D. We can clearly see the similarity in these curves, especially as the decoder approaches convergence where $I_E \rightarrow 1$ and $\sigma_{out}^2 \rightarrow 0$. This is intuitive since the iterations begin with zero *a priori* knowledge, $\sigma_{in}^2 = 1$ and $I_A = 0$, and as the iteration proceeds $\sigma_{in}^2 \rightarrow 0$ and $I_A \rightarrow 1$, which suggests such a relationship. We can follow a similar argument for the extrinsic output E if we make the Gaussian assumption on the output data.

Since σ_{in}^2 and I_A are both functions of σ_A we must express them as functions of σ_A in order to compare the two metrics.

If we express input variance in (2) as

$$\sigma_{in}^2 = f(\sigma_A),$$

where

$$f(\sigma_A) = \mathbb{E} \left\{ \left(\tanh\left(\frac{\lambda}{2}\right) - x \right)^2 \right\},$$

and mutual information as

$$I_A = 1 - g(\sigma_A),$$

where

$$g(\sigma_A) = \int_{-\infty}^{+\infty} \frac{e^{-((\xi - \frac{\sigma_A}{2})^2 / 2\sigma_A^2)}}{\sqrt{2\pi}\sigma_A} \log_2[1 + e^{-\xi}] d\xi.$$

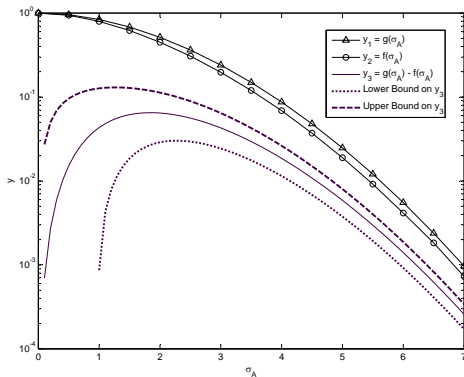


Fig. 3. Comparison of input variance and mutual information as functions of the variance of the a priori input.

We see that $f(\sigma_A) \approx g(\sigma_A)$ and can therefore postulate that

$$I_A \approx 1 - \sigma_{in}^2 \quad (5)$$

This is shown graphically in Figure 3, where the different functions are plotted. The triangles represent $g(\sigma_A)$, while the circles represent $f(\sigma_A)$. As can be seen there is little difference in these functions. The solid line represents the difference between the two functions.

IV. VARIANCE AND MUTUAL INFORMATION

In the previous section we derived a relationship between mutual information and noise variance based on a comparison via simulation. In this section we derive analysis to support our claim of the relationship between mutual information and input noise variance. Using the definition of input variance from (2) and

$$\tanh(z) = (e^{2z} - 1)/(e^{2z} + 1),$$

we get

$$\sigma_{in}^2 = 2 \int_{-\infty}^{+\infty} \left[\frac{p(\xi|x=+1)}{(e^\xi + 1)^2} + \frac{e^{2\xi}p(\xi|x=-1)}{(e^\xi + 1)^2} \right] d\xi.$$

Due to the symmetry in conditional PDFs, i.e. $p(\xi|x=-1) = e^{-\xi}p(\xi|x=+1)$, σ_{in}^2 can be simplified as

$$\sigma_{in}^2 = \int_{-\infty}^{+\infty} p(\xi|x=+1)r(\xi)d\xi, \quad (6)$$

where $r(\xi) = 2/(1 + e^\xi)$.

If we compare this to mutual information, which can be written as

$$I_A = 1 - \int_{-\infty}^{+\infty} p(\xi|x=+1)q(\xi)d\xi, \quad (7)$$

where $q(\xi) = \log_2(1 + e^{-\xi})$, we see that in order to validate the observation made in (5) we need to show that the difference between $r(\xi)$ and $q(\xi)$ is approximately zero. Substituting (6) and (7) into (5) and rearranging, we get

$$\phi(\sigma_A) \triangleq \int_{-\infty}^{+\infty} [q(\xi) - r(\xi)]p(\xi|x=+1)d\xi \approx 0. \quad (8)$$

We must therefore show that $\phi(\sigma_A) \approx 0$, to this end we define lower and upper bound functions which are linear

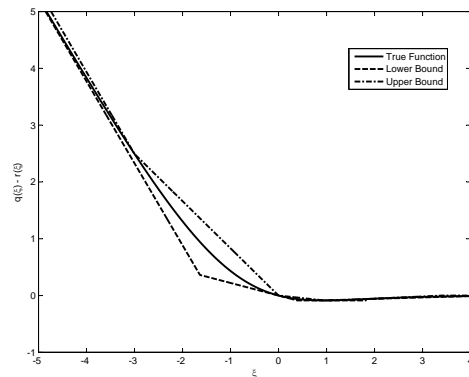


Fig. 4. Linear Approximation Upper and Lower Bounds to function $h(\xi)$

approximations to the true function $h(\xi) \triangleq q(\xi) - r(\xi)$ as shown in Figure 4. We use the complementary error function to evaluate (8) using the linear approximation of $h(\xi)$ to get the bounds

$$\phi_{lb}(\sigma_A) < \phi(\sigma_A) < \phi_{ub}(\sigma_A).$$

The derivation of these bounds is shown in the Appendix. Figure 3 graphically presents the lower and upper bounds derived, which are shown to be quite tight across a range of σ_A . The function $\phi(\sigma_A)$ represents the difference between the mutual information transfer function (EXIT) and the variance transfer function (VT).

V. CONCLUSION

We have derived a function that links EXIT chart analysis to VT chart analysis, where this function is $I_A \approx 1 - \sigma_{in}^2$. We have validated the function through the utilization of upper and lower bounds. This function can be useful to translate between EXIT and VT analysis methods.

We compared several methods of measuring mutual information and found the Egodicity Theorem method to have the lowest computational complexity for small block lengths.

APPENDIX

Consider the error function

$$\phi(\sigma_A) = \int_{-\infty}^{+\infty} [q(\xi) - r(\xi)]p(\xi|x=+1)d\xi \approx 0$$

where $r(\xi) = 2/(1 + e^\xi)$ and $q(\xi) = \log_2(1 + e^{-\xi})$, representing the difference between the mutual information transfer function and the variance transfer function. We use linear equations to bound the function

$$h(\xi) \triangleq q(\xi) - r(\xi)$$

in order to create an Upper and Lower Bound of $\phi(\sigma_A)$. We used the derivative and double derivative of $h(\xi)$ to determine the gradient and concavity of the function over several intervals of ξ . Since

$$h(\xi) = \log_2(1 + e^{-\xi}) - \frac{2}{1 + e^\xi},$$

then

$$h'(\xi) = \frac{e^\xi (2 \ln 2 - 1) - 1}{(1 + e^\xi)^2 \cdot \ln 2}$$

and

$$h''(\xi) = \frac{e^\xi [(2 \ln 2 + 1) - e^\xi (2 \ln 2 - 1)]}{(1 + e^\xi)^3 \cdot \ln 2}$$

We define

$$\xi_{min} = -\ln(2 \ln 2 - 1),$$

such that $h'(\xi_{min}) = 0$, and

$$\xi_{turn} = \ln \frac{2 \ln 2 + 1}{2 \ln 2 - 1},$$

such that $h''(\xi_{turn}) = 0$.

We calculate a number of linear approximations to the function $h(\xi)$ over a number of subsections of ξ . These subsections are characterized by the values of $h(\xi)$, $h'(\xi)$ and $h''(\xi)$, and are not distinct (there is some overlap). Thus we define the bounding function as the concatenation of the uppermost (Lower Bound) or lowest (Upper Bound) linear approximation for each value of ξ .

A. Lower Bound

Here we use five subsections of ξ .

$$\forall \xi \in (-\infty, 0],$$

$$h(\xi) > \log_2(e^{-\xi}) - \frac{2}{1 + e^{-\infty}} = -\frac{\xi}{\ln 2} - 2 = L_1(\xi).$$

$$\forall \xi \in (-\infty, \xi_{turn}), h''(\xi) > 0$$

$$h(\xi) > h(0) + h'(0) \cdot (\xi - 0) = \frac{\ln 2 - 1}{2 \ln 2} \xi = L_2(\xi).$$

$$\forall \xi \in (-\infty, +\infty),$$

$$h(\xi) > h(\xi_{min}) = \frac{1 - \ln 2 + \ln \ln 2}{\ln 2} = L_3(\xi).$$

$$\forall \xi \in (\xi_{turn}, \xi_l), h''(\xi) < 0$$

$$h(\xi) > h(\xi_{turn}) + (\xi - \xi_{turn}) \frac{h(\xi_l) - h(\xi_{turn})}{\xi_l - \xi_{turn}} = L_4(\xi),$$

where ξ_l is any real number satisfying $\xi_l > \xi_{turn}$.

$$\forall \xi \in (0, +\infty),$$

$$h(\xi) > \log_2(1 + 0) - \frac{2}{0 + e^\xi} = -2e^{-\xi} = L_5(\xi).$$

Thus, the lower bound can be written as

$$L(\xi) = \max_{k=1}^5(L_k(\xi))$$

B. Upper Bound

Here we divide the domain into six subsections.

$$\forall \xi \in (-\infty, \xi_0),$$

$$h(\xi) < -\frac{1}{\ln 2}(\xi - \xi_0) + h(\xi_0) = U_1(\xi)$$

(see Proof 1), where ξ_0 is an arbitrary real number satisfying $\xi_0 < 0$. The tightness of the upper bound may be adjusted through selection of ξ_0 , the bound shown in Figure 4 is for $\xi_0 = -3$.

$$\forall \xi \in (\xi_0, 0), h''(\xi) > 0,$$

$$h(\xi) < h(0) + (\xi - 0) \frac{h(\xi_0) - h(0)}{\xi_0 - 0} = \frac{h(\xi_0)}{\xi_0} \xi = U_2(\xi).$$

$$\forall \xi \in (0, \xi_{min}), h''(\xi) > 0,$$

$$h(\xi) < h(0) + (\xi - 0) \frac{h(\xi_{min}) - h(0)}{\xi_{min} - 0} = \frac{h(\xi_{min})}{\xi_{min}} \xi = U_3(\xi).$$

$$\forall \xi \in (\xi_{min}, \xi_{turn}), h''(\xi) > 0,$$

$$h(\xi) < h(\xi_{min}) + (\xi - \xi_{min}) \frac{h(\xi_{turn}) - h(\xi_{min})}{\xi_{turn} - \xi_{min}} = U_4(\xi).$$

$$\forall \xi \in (\xi_{turn}, +\infty), h''(\xi) < 0,$$

$$h(\xi) < h(\xi_{turn}) + h'(\xi_{turn})(\xi - \xi_{turn}) = U_5(\xi).$$

$$\forall \xi \in (\xi_{turn}, +\infty),$$

$$h(\xi) < 0 = U_6(\xi)$$

(see Proof 2).

Thus, the upper bound can be written as

$$U(\xi) = \min_{k=1}^6(U_k(\xi))$$

C. Bounds

Our bounding functions are therefore L and U , which are linear approximations of the error function $h(\xi)$. By imposing these bounds on h we in turn bound ϕ , since ϕ is a function of h .

Proof 1:

Define $f(\xi) = h(\xi) - U_1(\xi)$.

When $-\infty < \xi < \xi_0$, we have

$$f(\xi_0) = 0$$

$$f'(\xi) = h'(\xi) + \frac{1}{\ln 2}$$

$$\lim_{\xi \rightarrow -\infty} f'(\xi) = 0$$

$$f''(\xi) = h''(\xi) > 0.$$

When $\xi > -\infty$,

$$f'(\xi) > f'(-\infty) = 0 \Rightarrow f(\xi) < f(\xi_0) = 0 \Rightarrow h(\xi) < U_1(\xi)$$

□

Proof 2:

$$h(\xi_{turn} < 0, \lim_{\xi \rightarrow +\infty} h(\xi) = 0, \lim_{\xi \rightarrow +\infty} h'(\xi) = 0$$

When $\xi_{turn} < \xi < +\infty$, we have

$$h''(\xi) < 0 \Rightarrow h'(\xi) > h'(+\infty) = 0 \Rightarrow h(\xi) < h(+\infty) = 0$$

□

ACKNOWLEDGMENTS

D.P. Shepherd, M. Ruan, M.C. Reed, and Zhenning Shi are with National ICT Australia and affiliated with the Australian National University. National ICT Australia is funded through the Australian Government's *Backing Australia's Ability* initiative and in part through the Australian Research Council.

REFERENCES

- [1] C. Berrou, A. Glavieux, and P. Thitmajshima, "Near Shannon limit error-correcting coding and decoding:turbo-codes," in *IEEE Int. Conf. on Communications*, (Geneva, Switzerland), pp. 1064–1070, May 1993.
- [2] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, Oct. 2001.
- [3] P. D. Alexander, A. J. Grant, and M. C. Reed, "Performance analysis of an iterative decoder for code-division multiple-access," *European Trans. on Telecom.*, vol. 9, pp. 419–426, Sep./Oct. 1998.
- [4] Z. Shi and C. Schlegel, "Joint iterative decoding of serially concatenated error control coded CDMA," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 8, pp. 1646–1653, 2001.
- [5] S. D. D. Divsalar and F. Pollara, "Iterative turbo decoder analysis based on density evolution," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 891–907, May 2001.
- [6] M. Tuchler, S. ten Brink, and J. Hagenauer, "Measures for tracing convergence of iterative decoding algorithms and channel coding," in *12th IEEE/ITG Conf. on Source*, (Berlin, Germany), pp. 53–60, Jan. 2002.
- [7] J. Hagenauer, "The EXIT Chart - introduction to extrinsic information transfer in iterative processing," in *12th European Signal Processing Conference*, pp. 1541–1548, September 2004.