International Symposium on Information Theory and its Applications, ISITA2004
Parma, Italy, October 10–13, 2004

Improved Sum-Min Decoding of LDPC Codes

Gottfried LECHNER and Jossy SAYIR

Telecommunications Research Center Vienna (ftw.)
Donaucitystrasse 1/3,
1220 Vienna, Austria
Email: {lechner|sayir}@ftw.at

Abstract

EXIT charts are used to provide a bound for the performance of the Sum-Min algorithm for decoding of Low-Density Parity-Check codes. We present an optimal non-linear post-processing function of the check node outputs that achieves this bound. This function can be well approximated by a linear function, resulting in a simple normalization of the check node outputs, as proposed in [1] and [2]. A systematic approach for determining the optimal normalization factor using EXIT charts is presented.

1. INTRODUCTION

Decoding of Low-Density Parity-Check (LDPC) [3] codes using the Sum-Product Algorithm (SPA) can be too complex for hardware implementation. The algorithm can be simplified by approximating the calculation at the check nodes by a simple minimum operation, resulting in the well-known Sum-Min algorithm (SMA). While the Sum-Min algorithm is less complex to implement, it requires approximately additional 0.5dB of signal-to-noise ratio E_b/N_0 to achieve the same bit error rate when used for transmission over an additive white Gaussian noise (AWGN) channel with binary input.

It was observed in [1] and [2] that the performance of the Sum-Min algorithm can be improved by *linear post-processing* of the messages emitted by the check nodes. Simulations show that linear post-processing, i.e., a simple normalization of the messages, is sufficient to achieve good performance. The optimal normalization factor was determined in [2] using density evolution to search for the factor that yields the lowest threshold.

This paper uses Extrinsic-Information Transfer (EXIT) charts to calculate a lower bound on the threshold that can be achieved using the Sum-Min algorithm with any post-processing of the check node messages. We derive a function for optimal *non-linear post-processing*. It is obtained by deriving the a-posteriori Log-Likelihood Ratio (LLR) given the message emitted by a check node. Simulations show that a linear approximation of this function leads to a threshold close to the bound, sheding new light on the results in [1] and [2]. Finally, we present a systematic way to find the optimal linear approximation using EXIT charts which are less computationaly complex than density evolution.

It is interesting to note the complementary insights obtained by analysing the Sum-Min algorithm with density evolution and with EXIT charts. Density evolution gives an exact measure of the performance achieved by the algorithm but gives no indication that this performance can be improved by post-processing the messages. EXIT charts, on the other hand, give a bound on the performance achievable by any post-processing. This bound can be approached or attained by seeking for the optimal post-processing function using density evolution (as in [1] and [2]) or EXIT charts (as in this paper).

2. EXIT CHARTS

We use EXIT charts [4] to compare the performance of the Sum-Product algorithm and the Sum-Min algorithm. This implies that all results assume a Gaussian distribution and independence of the messages passed in the factor graph. Without loss of generality, our simulations are for a regular LDPC code with variable degree d_v, check node degree d_c and rate 0.5. The results obtained apply just as well to irregular codes.

The EXIT function of the variable nodes remains unchanged, since the operations at the variable nodes are the same for both algorithms. For the check nodes, we simulated the transfer function for both algorithms and found the difference to be on the order of 5×10^{-3} bits at most, i.e., very small.

This work was supported by the K*plus* program.

Unlike the Sum-Product algorithm, in the Sum-Min algorithm the messages emitted by the check nodes are not true log-likelihood ratios as assumed in the computation performed at the variable nodes. Therefore, the decoding trajectory of the Sum-Min decoder in Figure 1 is below the prediction of the EXIT charts. In a sense, the EXIT chart shows that there is a strong correlation between the message emitted by the check nodes and the transmitted bits, but the algorithm fails to reap the full benefits of this strong correlation. This indicates that we need to apply a function to the output of the check nodes that preserves the strong correlation but transforms the message into the log-likelihood domain, so that the subsequent operations at the variable nodes can reap the full benefit of the strong correlation. This is what we call post-processing.

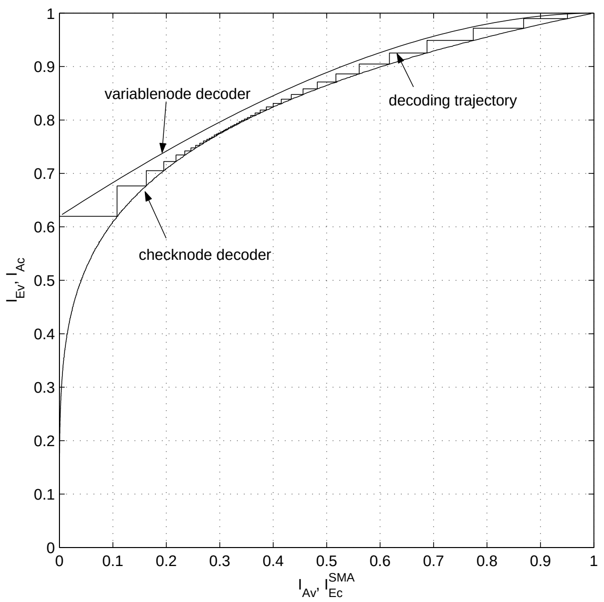

Figure 1: EXIT chart and decoding trajectory for SMA and $E_b/N_0 = 1.72$dB ($d_v = 3$, $d_c = 6$).

The simulated EXIT functions give us a bound for the performance of the Sum-Min algorithm. This bound can be achieved if optimal post-processing is applied so that the variable node decoder receives true log-likelihood ratios from the check nodes. The bounds obtained for ensembles of LPDC codes are shown in the table below and compared with the thresholds for the Sum-Product algorithm.

(d_v, d_c)	(3,6)	(4,8)	(5,10)
SPA threshold [dB]	1.13	1.57	2.02
SMA bound [dB]	1.19	1.65	2.11

These results show that the bounds for the Sum-Min algorithm are very close to the thresholds for the Sum-Product algorithm. This motivates us to post-process the messages generated by the Sum-Min algorithm in order to improve its performance.

3. POSTPROCESSING OF CHECK NODE MESSAGES

In order to get true log-likelihood ratios from the minimum computed in the check nodes of the Sum-Min algorithm, we have to calculate

$$L_{mn} = \log \frac{p\left(X_n = 0 | L_{mn}^{\mathrm{SMA}}\right)}{p\left(X_n = 1 | L_{mn}^{\mathrm{SMA}}\right)}, \tag{1}$$

where L_{mn}^{SMA} is the original messages from check node m to variable node n using the Sum-Min approximation and L_{mn} is the message after postprocessing. We write this post-processing function as

$$L_{mn} = f\left(L_{mn}^{\mathrm{SMA}}; d_c, \text{extrinsic channel parameters}\right), \tag{2}$$

where d_c is the check node degree.

Using the expression for the probability density function of the check node output derived in [5], we can determine the general expression for the optimal post-processing function. If $\zeta = L_{mn}^{\mathrm{SMA}}$ is the message emitted by the check node of the Sum-Min algorithm, then the true log-likelihood ratio is

$$L_{mn} = f(\zeta, d_c, \sigma) = \log \frac{\Psi_+(\zeta, d_c, \sigma)}{\Psi_-(\zeta, d_c, \sigma)} \tag{3}$$

where

$$\begin{aligned} \Psi_+(\zeta, d_c, \sigma) &= [p(\zeta) + p(-\zeta)] \left[\phi_+(\zeta) + \phi_-(\zeta)\right]^{d_c - 2} \\ &+ [p(\zeta) - p(-\zeta)] \left[\phi_+(\zeta) - \phi_-(\zeta)\right]^{d_c - 2} \end{aligned}$$

$$\begin{aligned} \Psi_-(\zeta, d_c, \sigma) &= [p(\zeta) + p(-\zeta)] \left[\phi_+(\zeta) + \phi_-(\zeta)\right]^{d_c - 2} \\ &- [p(\zeta) - p(-\zeta)] \left[\phi_+(\zeta) - \phi_-(\zeta)\right]^{d_c - 2}, \end{aligned}$$

$p(\zeta)$ is the probability density function of the messages at the inputs of the check nodes and $\phi_+(\zeta)$ and $\phi_-(\zeta)$ are defined as

$$\phi_+(\zeta) = \int_{+|\zeta|}^{+\infty} p(y)dy$$

$$\phi_-(\zeta) = \int_{-\infty}^{-|\zeta|} p(y)dy.$$

Under the Gaussian assumption, $p(\zeta)$ is given by

$$p(\zeta) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{(\zeta - \mu)^2}{\sigma^2}},$$

where σ and μ depend only I_{Ac} as[1]

$$\sigma = J^{-1}(I_{Ac}) \text{ and } \mu = \frac{\sigma^2}{2}$$

and the integrals can be written as

$$\phi_+(\zeta) = \frac{1}{2}\operatorname{erfc}\left(\frac{|\zeta|-\mu}{\sqrt{2}\sigma}\right)$$

$$\phi_-(\zeta) = \frac{1}{2}\operatorname{erfc}\left(\frac{|\zeta|+\mu}{\sqrt{2}\sigma}\right).$$

Since the a-priori information I_{Ac} is a bijective function of σ, we can also express the log-likelihood in function of ζ, d_c and I_{Ac}, relating it clearly to the trajectory in the EXIT chart. For $d_c = 6$, Figure 2 and Figure 3 show this function for $I_{Ac} = 0.5$ and $I_{Ac} = 0.9$ respectively. For high values of I_{Ac} this function becomes a linear function with slope 1, i.e. the Sum-Min algorithm approximation delivers true LLR values for high a-priori information.

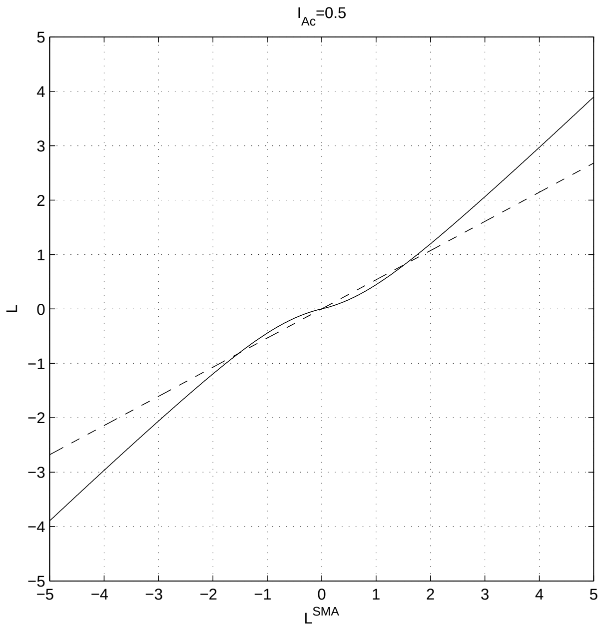

Figure 2: Postprocessing function $L = f(L^{\text{SMA}}; d_c = 6, I_{Ac} = 0.5)$ (solid) and linear approximation (dashed).

It would be too complex to implement this parameterized function. Remember that the aim of the Sum-Min algorithm approximation is to simplify the decoder. However, we can approximate $f(\cdot)$ by a linear function which results in a normalization of the

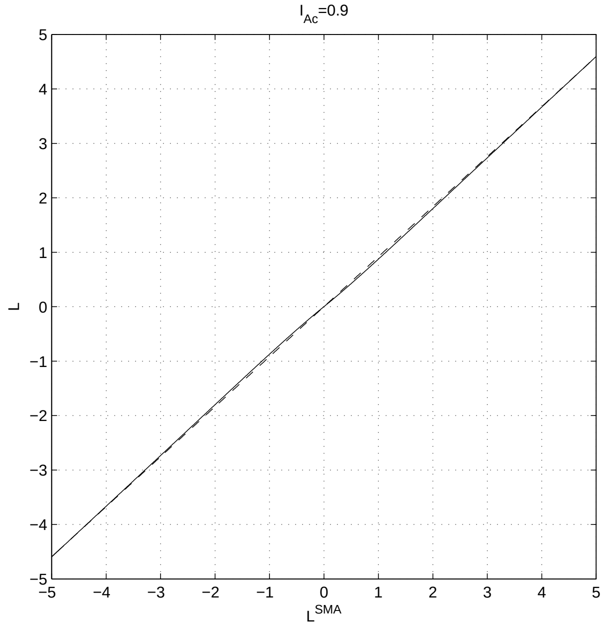

Figure 3: Postprocessing function $L = f(L^{\text{SMA}}; d_c = 6, I_{Ac} = 0.9)$ (solid) and linear approximation (dashed).

check node outputs. The approximation of $f(\cdot)$ has to be done considering the probability density function $p(L^{\text{SMA}}; I_{Ac})$ of L^{SMA} parameterized by I_{Ac}.

We define $\alpha(d_c, I_{Ac})$ as the normalization factor which minimizes the expected squared error as

$$\alpha(d_c, I_{Ac}) = \underset{\alpha'}{\operatorname{argmin}} \int_{-\infty}^{\infty} \left[f(L^{\text{SMA}}; d_c, I_{Ac}) - \frac{1}{\alpha'} L^{\text{SMA}} \right]^2 \cdot p(L^{\text{SMA}}; I_{Ac}) dL^{\text{SMA}}.$$

Using this linear approximation, Equation 2 becomes

$$L_{mn} = \frac{L_{mn}^{\text{SMA}}}{\alpha(d_c, I_{Ac})}. \tag{4}$$

The approximation of the nonlinear function is shown as a dashed line in Figure 2 and Figure 3.

4. OPTIMAL NORMALIZATION FACTOR

Our derivation so far requires the normalization factor to be adapted throughout the iteration process as a function of the a-priori mutual information between the check node inputs and the transmitted code digits. This may be inconvenient for practical implementations.

If, on the other hand, we require a fixed normalization factor for post-processing at the check nodes during the whole decoding process, there are several

[1] The J-function is defined as $J(\sigma) = 1 - \int_{-\infty}^{\infty} \frac{e^{-\frac{(\theta-\sigma^2/2)^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma} \log_2\left(1+e^{-\theta}\right) d\theta$.

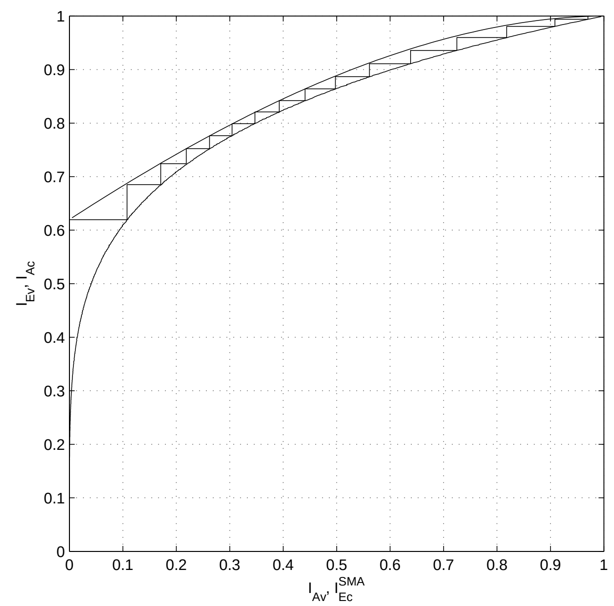

Figure 4: EXIT chart and decoding trajectory for Sum-Min algorithm with normalization $\alpha = 1.25$ and $E_b/N_0 = 1.72$dB ($d_v = 3$, $d_c = 6$).

approaches to finding the optimal value of this normalization factor. One approach is to optimize α for the first iteration as in [1][2]. This attempt does not provide the best value, as shown in [2], where α was determined by performing density evolution.

Our approach is to find the most critical point in the EXIT chart, i.e. the point where the gap between the extrinsic information transfer functions is narrowest and goes to zero for decreasing SNR. For the example of a regular LDPC code with $d_v = 3$ and $d_c = 6$, we find that the EXIT functions have their narrowest gap at $I_{Ac} = 0.75$. In the context of our computation, we obtain $\alpha = 1.25$. This is the same value as found with density evolution.

Using the optimal normalization factor for this code, we obtain a decoding trajectory that complies with the EXIT chart prediction as shown in Figure 4.

5. CONCLUSION

We used EXIT charts to analyze the performance of the SMA decoder. This analysis showed that SMA decoding can get very close to SPA decoding if the messages from check nodes to variable nodes are postprocessed to get true LLR values. We derived an analytical expression for the optimal nonlinear postprocessing function. We showed that this function can be linearly approximated and that further simplification can be done, by using a constant approximation independent of the channel parameter and iteration. Finally, we presented a systematic approach to determine this optimal linear approximation.

[2] We note that the authors defined $\alpha = \frac{E(|L^{\mathrm{SMA}}|)}{E(|L^{\mathrm{SPA}}|)}$ heuristically, which yields similar results.

References

[1] Jinghu Chen and Marc P. C. Fossorier, "Near optimum universal belief propagation based decoding of low-density parity check codes," *IEEE Transactions on Communications*, vol. 50, no. 3, pp. 406–414, March 2002.

[2] Jinghu Chen and Marc P. C. Fossorier, "Density evolution for two improved BP-based decoding algorithms of LDPC codes," *IEEE Communications Letters*, vol. 6, no. 5, pp. 208–210, May 2002.

[3] R.G. Gallager, "Low density parity check codes," *IRE Transactions on Information Theory*, vol. IT-8, pp. 21–28, Jan 1962.

[4] A. Ashikhmin, G. Kramer, and Stephan ten Brink, "Extrinsic information transfer functions: model and erasure channel properties," To appear in IEEE Transactions on Information Theory, 2004.

[5] X. Wei and A.N. Akansu, "Density evolution for low-density parity-check codes under max-log-map decoding," *Electronics Letters*, vol. 37, no. 18, pp. 1125–1126, 2001.