

Applying Bayesian networks and belief propagation to error correction coding

W. C. Olding¹

J. C. Olivier²

B. P. Salmon³

(Received 18 December 2013; revised 26 March 2014)

Abstract

Telecommunication standards utilise numerous different subsystems to improve the quality of voice and data communications. One of these subsystems is tasked with error detection and control within the transmitted streams. This is accomplished using numerous different error correction codes for various scenarios. This article investigates the use of a Bayesian network as a universal channel decoder for the two main branches of algebraic codes. This graph-based approach is visually intuitive and is found to produce similar results to the best performing decoders in use today. A graphical representation of a multi-path channel equaliser is incorporated into the Bayesian network, obtaining near optimal performance.

<http://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/7817>

gives this article, © Austral. Mathematical Soc. 2014. Published May 7, 2014, as part of the Proceedings of the 11th Biennial Engineering Mathematics and Applications Conference. ISSN 1446-8735. (Print two pages per sheet of paper.) Copies of this article must not be made otherwise available on the internet; instead link directly to this URL for this article.

Contents

1 Introduction	C97
2 Algebraic codes	C99
3 Bayesian networks	C99
4 Modelling a communication system with a Bayesian network	C101
5 Inclusion of multi-path channel models	C103
6 Results	C105
7 Conclusion	C107
References	C108

1 Introduction

Digital communications standards that are in use today, such as 3G and 4G, build on decades of research and incorporate sophisticated transmitter and receiver protocols [9]. New innovative space-time modulation and encoding techniques utilising antenna arrays all have the same goals: to improve the quality of the transmitted and received signal, to provide better quality of service and to reduce required transmission power to prolong battery life [9]. These space-time modulation and encoding techniques are implemented using digital electronics hardware at the device level, and the complexity of the algorithms increased significantly over the last decade. Thus there is a need for more intuitive approaches to model encoding, modulation and the subsequent decoding of linear block error correction codes.

In this article we revisit the probabilistic graphical model known as a Bayesian network (BN) to represent the coded information bits and their conditional probabilities given noisy observed data [4]. It is a common scenario in a wireless communication system that there are multiple paths (known as taps) for a signal to travel between transmitter and receiver. Each tap results in a different propagation time and if not rectified by an effective equaliser, the constructive and destructive interference of the received signals leads to a greatly increased chance of detection error. We propose an intuitive graphical representation of the channel to allow for joint equalisation and decoding via the belief propagation (BP) algorithm.

The BN provides a reduced probability of error for an incorrect decision at the receiver, when compared to the case where the channel equaliser and decoder are used independently. The intuitive approach of the BN allows the method to be applied to several different applications and fields of research where unobservable parameters need to be estimated, given noisy observational data. The graphical approach is straightforward to understand and to visualise, yet it is shown to result in equivalent numerical calculations and performance when compared to existing algorithms that are applied in high speed communication systems.

To illustrate the versatility of the BN the graph-based approach is applied as a universal channel decoding method and is tested on two main types of algebraic codes (convolutional and low density parity check (LDPC) codes). Results indicate performance similar to other applicable methods, but with advantages in the ability to visualise, understand and compute via a graphical model. An important and practical case is the application of the LDPC codes over a multi-path (dispersive) channel. We show that our graph approach is able to incorporate the multi-path elegantly, without affecting the parity check functions of the code, and jointly decode and equalise effectively using BP.

2 Algebraic codes

A (n, k) linear block code (LBC) encodes a message \mathbf{m} of k bits into a codeword \mathbf{c} of n bits, $n > k$, with a corresponding code rate of $R = \frac{k}{n}$. The encoding process is expressed as a multiplication with a $(k \times n)$ generator matrix \mathbf{G} :

$$\mathbf{c} = \mathbf{m}\mathbf{G}. \quad (1)$$

The parity check matrix \mathbf{H} is used to decode the codeword and is orthogonal to the generator matrix,

$$\mathbf{G}\mathbf{H}^T = \mathbf{0}. \quad (2)$$

LDPC codes are a subclass of LBC. These codes are characterised by a sparse parity check matrix that is optimised for probabilistic graph based decoding techniques [3]. By employing BP on a Tanner graph, the LDPC codes achieve error correction performance near channel capacity.

Convolutional codes are another subclass of algebraic codes that encode the message \mathbf{m} through discrete convolution with a finite impulse response filter. The encoder is represented by a banded generator matrix allowing the code to be treated as if it were a LBC. Convolutional codes are optimally decoded using the Viterbi algorithm [10], which employs a maximum likelihood approach to track the most probable path through a trellis diagram.

3 Bayesian networks

A BN is a probabilistic directed graphical model that represents dependencies among variables. It simplifies the representation of a domain by factorising the joint probability function into independent conditional probability functions (CPFs) that capture the interactions between each variable X_i and the set of

variables S_i that influence X_i directly. This is expressed by

$$\Pr(X_1, \dots, X_N) = \prod_{i=1}^N \Pr(X_i | S_i), \quad (3)$$

where N is the number of variables and $S_i \subset \{X_k\}_{k=1}^{k=N} \setminus X_i$ [6].

The BP algorithm is a common method to efficiently compute posterior probability distributions for all unobservable variables given the observed variables in a probabilistic graphical model [7]. BP is a message passing algorithm capable of performing exact inference in probabilistic graphical models that are free from loops. In this context a loop is defined as an undirected cycle in the underlying graph structure. In the loop free case the algorithm is guaranteed to reach an exact solution in time proportional to the diameter of the network [6]. The algorithm is also applied to graphs containing loops, where it approximates the posterior distribution. Repeated iterations, under certain conditions, result in the posterior distributions converging to the correct values; however, convergence is not guaranteed.

Consider a discrete variable node X that takes some value x . Let the belief $\Pr(X | \mathbf{e})$ of node X be the probability distribution of X given all evidence \mathbf{e} provided by observations of other variables within the network. For a network containing no undirected loops this belief decomposes into two components. Let the evidence contained in the network above and below X be defined as \mathbf{e}_X^+ and \mathbf{e}_X^- , respectively. The belief of node X is thus

$$\Pr(X | \mathbf{e}) = \alpha \Pr(X | \mathbf{e}_X^+) \Pr(\mathbf{e}_X^- | X), \quad (4)$$

where α denotes a scaling factor used in the algorithm to ensure the probabilities for a given node sum to unity.

Let node X possess a set of n parent nodes $\mathbf{U} = \{U_1, \dots, U_n\}$, and m child nodes $\mathbf{Y} = \{Y_1, \dots, Y_m\}$. The evidence contained within the network is further subdivided into

$$\mathbf{e}_X^+ = \{\mathbf{e}_{U_1 X}^+, \dots, \mathbf{e}_{U_n X}^+\}, \quad \mathbf{e}_X^- = \{\mathbf{e}_{XY_1}^-, \dots, \mathbf{e}_{XY_m}^-\}, \quad (5)$$

where $e_{\mathbf{u}_i\mathbf{X}}^+$ denotes evidence contained within the network connected to and including the parent node \mathbf{U}_i and $e_{\mathbf{Y}_j\mathbf{X}}^-$ denotes evidence contained with the network connected to and including the child node \mathbf{Y}_j . Given this subdivision,

$$\Pr(e_{\mathbf{X}}^- | \mathbf{X}) = \prod_{j=1}^m \Pr(e_{\mathbf{Y}_j\mathbf{X}}^- | \mathbf{X}). \quad (6)$$

In the message passing scheme the distribution $\Pr(e_{\mathbf{X}\mathbf{Y}_j}^- | \mathbf{X})$ is a message sent from child node \mathbf{Y}_j to \mathbf{X} . The distribution over parent nodes is computed by marginalisation over the set \mathbf{U} :

$$\Pr(\mathbf{X} | e_{\mathbf{X}}^+) = \sum_{\mathbf{u}} \Pr(\mathbf{X} | \mathbf{u}) \prod_{i=1}^n \Pr(\mathbf{u}_i | e_{\mathbf{u}_i\mathbf{X}}^+), \quad (7)$$

where $\Pr(\mathbf{X} | \mathbf{u})$ is the CPF for node \mathbf{X} and the distribution $\Pr(\mathbf{u}_i | e_{\mathbf{u}_i\mathbf{X}}^+)$ is the message that \mathbf{X} receives from a parent node \mathbf{u}_i . The summation over the set \mathbf{u} requires summation over every combination of variables within the set.

The messages are computed as described by Pearl [7] using

$$\Pr(\mathbf{X} | e_{\mathbf{X}\mathbf{Y}_j}^+) = \alpha \left[\prod_{k \neq j} \Pr(e_{\mathbf{Y}_k\mathbf{X}}^- | \mathbf{X}) \right] \sum_{\mathbf{u}} \Pr(\mathbf{X} | \mathbf{u}) \prod_i \Pr(\mathbf{u}_i | e_{\mathbf{u}_i\mathbf{X}}^+), \quad (8)$$

$$\Pr(e_{\mathbf{X}\mathbf{u}_i}^- | \mathbf{u}_i) = \beta \sum_{\mathbf{X}} \Pr(e_{\mathbf{X}}^- | \mathbf{X}) \sum_{\mathbf{u}_k, k \neq i} \Pr(\mathbf{X} | \mathbf{u}) \prod_{k \neq i} \Pr(\mathbf{u}_k | e_{\mathbf{u}_k\mathbf{X}}^+), \quad (9)$$

where α ensures that the probability distribution across \mathbf{X} sums to unity, β is an arbitrary constant.

4 Modelling a communication system with a Bayesian network

McEliece et al. [4] showed that BN can be used to compactly describe probabilistic decoders. They also showed that when the belief propagation algorithm

is applied to these networks the resulting decoder exhibits good error correction properties. This section presents simple decoder networks and the next section illustrates how this is extended to multi-path channel models.

The purpose of any communication receiver is to estimate the transmitted message given some noisy received signal. Digital communication requires an estimation of a set of unobservable (hidden) message symbols, \mathbf{M} , given a set of noisy observations, \mathbf{Z} . The estimate of message bit M_i given an observation $Z_i = z_i$ is

$$\hat{m}_i = \operatorname{argmax}_{M_i} \left\{ \frac{\Pr(M_i) \Pr(z_i | M_i)}{\Pr(z_i)} \right\}. \quad (10)$$

To model an additive white Gaussian noise (AWGN) channel, the CPF for each observation node Z_i is a Gaussian function with a mean determined by the corresponding transmitted bit M_i , and a variance σ^2 determined by the channel noise. The line coding function $L(x)$ defines the binary phase shift keying modulation used within the communication system, where $L : [0, 1] \rightarrow [+1, -1]$, which produces the CPF

$$\Pr(z_i | m_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp \left\{ -\frac{[z_i - L(m_i)]^2}{2\sigma^2} \right\}. \quad (11)$$

Upon instantiation of an observation node for a given BN, the channel model maps the received value into a probability that is propagated throughout the network.

Equation (1) is modelled in a BN with the inclusion of an additional layer of hidden codeword symbols. The connection between the message symbols and the codeword symbols is characterised by the \mathbf{G} matrix where nonzero entries indicates a directed connection. The CPF for the codeword nodes is deterministic and represents the modulo-2 multiplication of all parent variables. The BP algorithm on this network yields the posterior distributions for every message node with respect to the channel observations, code constraints and prior distributions. An example network for a rate $\frac{1}{2}$ convolutional code is shown in Figure 1.

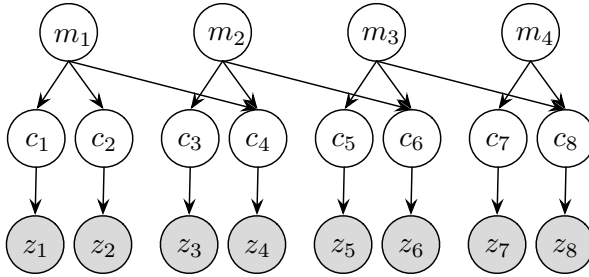


Figure 1: BN for rate $\frac{1}{2}$ convolutional code

Figure 1 shows how information from any channel observation nodes may indirectly influence the belief of any message symbol node by propagation through the network. In this case, odd numbered code symbol nodes simply copy the values of their single parents. They are only included for clarity and are not required.

5 Inclusion of multi-path channel models

This section shows how more complex channel models are included in a BN representation of a communication system. Consider a multi-path channel characterised by a discrete impulse response (IR) with L -taps:

$$H(\mathbf{n}) = \alpha_0 \delta(\mathbf{n}) + \dots + \alpha_{L-1} \delta[\mathbf{n} - (L - 1)],$$

where δ is the Kronecker delta function and α_l is the amplitude of the l th tap. These channel properties are included in the channel layer in much the same way as a convolutional encoder. Inclusion of a multi-path channel model allows for simultaneous equalisation and decoding known as turbo-equalisation.

Each channel observation node z_i is connected to L previous codeword nodes $c_i, \dots, c_{i-(L-1)}$, as shown in Figure 2(a). This results in a directed edge

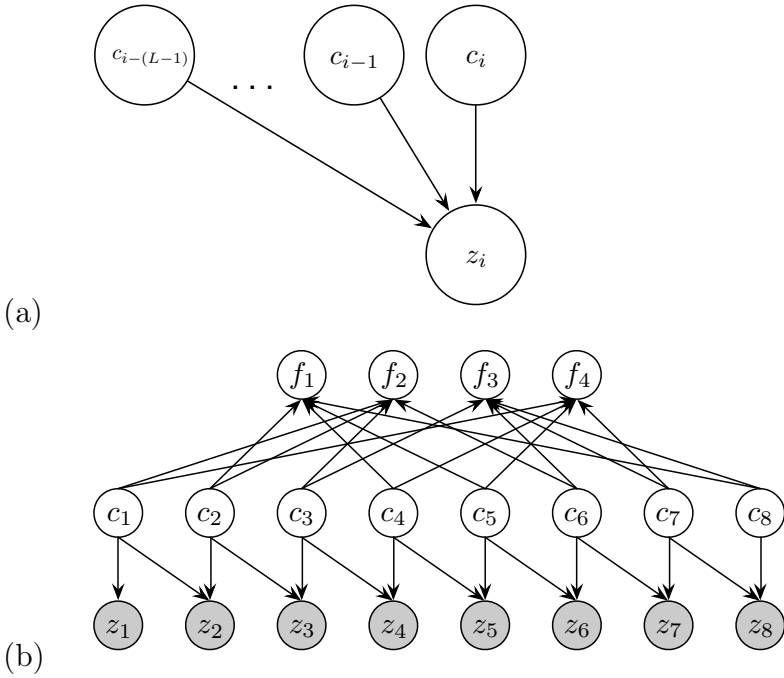


Figure 2: Multi-path channel models in BNs for (a) observation node for L-tap channel model; and (b) $(8,4)$ LDPC code with 2-tap multi-path channel model.

creating short loops within the network, as depicted in Figure 2(b). The CPF of the observation nodes becomes a Gaussian function with the mean determined by the sum of the previously transmitted bits weighted by the channel IR. The complexity of marginalisation over the CPF increases exponentially with the number of taps. For this reason only the most significant taps should be included [1]. Although the inclusion of a multi-path channel introduces many short loops to the network, Myburgh et al. [5] showed that the BP algorithm will converge provided the primary tap is sufficiently dominant. By applying a minimum phase pre-filter to the received channel observations it is possible to enforce this constraint. By including a model of the channel

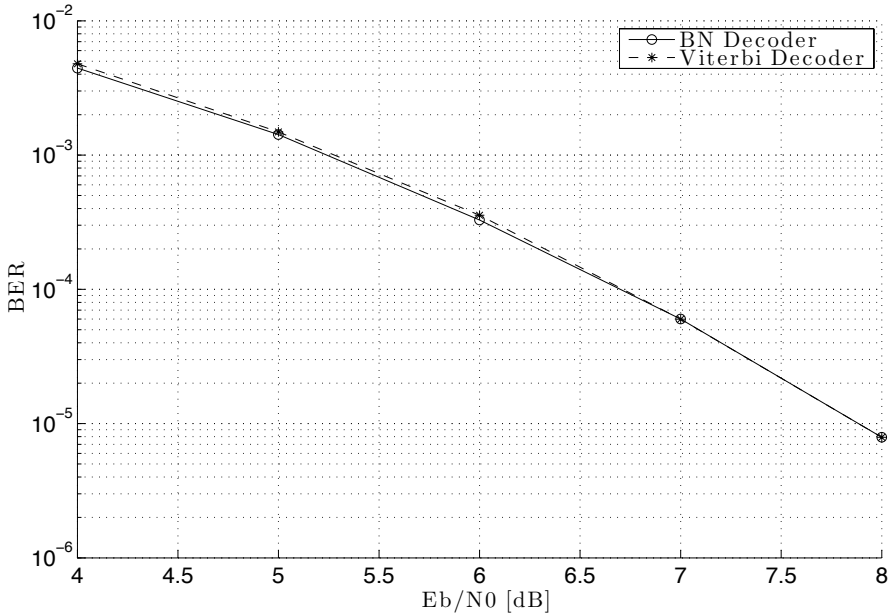


Figure 3: Comparison of BER for a $L = 1$, $k = 1$, $n = 2$ convolutional code (following the notation of Proakis et al. [8]) of length 2×10^3 decoded using BN decoder with conventional Viterbi decoder.

directly in the decoder network it becomes capable of turbo-equalisation [2].

6 Results

A convolutional decoder was implemented as described in Section 4 and was tested by simulating transmission over an AWGN channel. Each point on the curve was generated by simulating transmission until 1000 errors were accumulated. The number of errors is divided by the number of total bits transmitted to obtain the bit error rate (BER). As the convolutional code

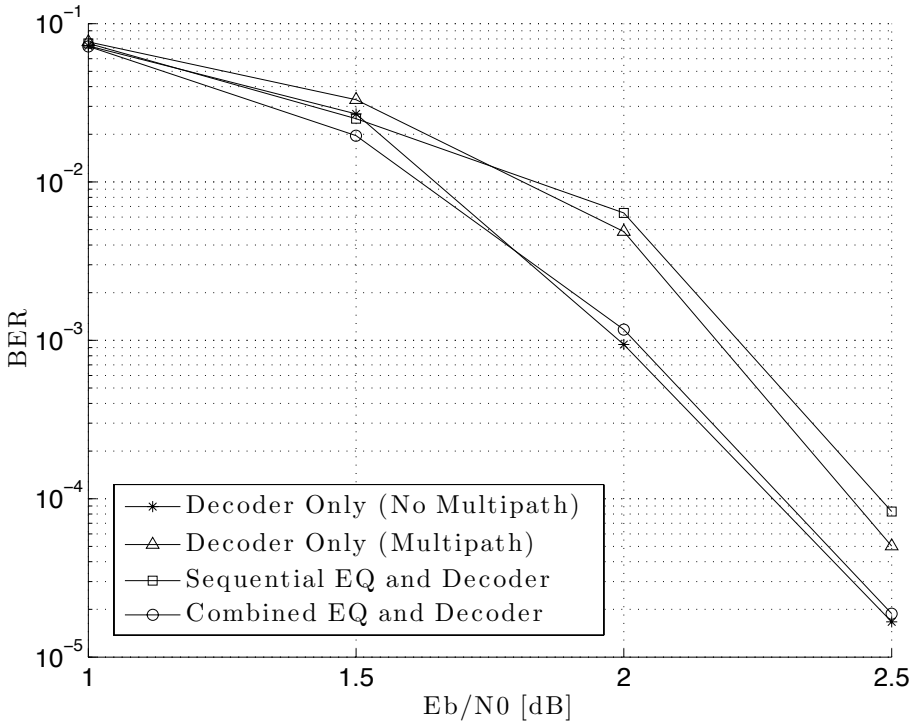


Figure 4: Comparison of BER for a $j = 3$, $k = 6$ LDPC code (following the notation of Gallager [3]) of length 2×10^3 using equalisation techniques across multi-path channel.

network is free from loops, the BP algorithm converges to exact results in time proportional to the diameter of the network. The BER performance of the BN decoder is verified by comparison with a conventional Viterbi decoder [8], as shown in Figure 3.

A second experiment was conducted to test the turbo-equalisation performance of the BN in a multi-path channel. Simulations were conducted for the following cases:

- LDPC code in an AWGN channel;
- LDPC code in an multi-path channel;
- sequential equalisation and LDPC decoding in multi-path channel with soft values passed between stages;
- turbo-equalisation of LDPC code in multi-path channel.

In each case a randomly generated regular LDPC code was used and in the multi-path channel cases a model with relative tap weights of $[1, 0.1]$ was used. Again 1000 errors were simulated for each point.

Figure 4 shows that in the sequential equalisation and decoding cases the equaliser exhibits little ability to recover errors due to the multi-path and hinders the decoder at higher signal quality. The same equalisation and decoding elements combined on the same BN shows a performance that approaches that of the AWGN only channel.

7 Conclusion

The graph-based algorithm is a visually intuitive approach to assist in understanding the interactions among the unobservable parameters that need to be estimated and the observable data. The use of a BN as a universal channel decoder was explored and we found it produced similar results to two popular decoding algorithms used in telecommunications, namely the

decoding of convolutional codes and LDPC codes. The BN was also deployed as a turbo-equaliser to mitigate the effects of multi-path propagation. The graph produced near optimal results if the majority of the power is contained in the primary tap [5]. We conclude that a BN can be used successfully as a universal channel decoding method, provided a graph can be defined to describe the connections within the code.

References

- [1] G. Colavolpe. On LDPC codes over channels with memory. *IEEE T. Wirel. Commun.*, 5(7):1757–1766, 2006. doi:[10.1109/TWC.2006.1673087](https://doi.org/10.1109/TWC.2006.1673087). C104
- [2] C. Douillard, M. Jézéquel, C. Berrou, A. Picart, P. Didier, and A. Glavieux. Iterative correction of intersymbol interference: Turbo-equalization. *Eur. T. Telecommun.*, 6(5):507–511, 1995. doi:[10.1002/ett.4460060506](https://doi.org/10.1002/ett.4460060506). C105
- [3] R. G. Gallager. *Low-density parity-check codes*. PhD thesis, Massachusetts Institute of Technology, 1963. <http://www.rle.mit.edu/rgallager/documents/ldpc.pdf>. C99, C106
- [4] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng. Turbo decoding as an instance of Pearl’s “belief propagation” algorithm. *IEEE J. Sel. Area. Comm.*, 16(2):140–152, 1998. doi:[10.1109/49.661103](https://doi.org/10.1109/49.661103) C98, C101
- [5] H. C. Myburgh, J. C. Olivier, and A. J. van Zyl. Reduced complexity turbo equalization using a dynamic bayesian network. *EURASIP J. Adv. Sig. Pr.*, 2012:136, 2012. doi:[10.1186/1687-6180-2012-136](https://doi.org/10.1186/1687-6180-2012-136). C104, C108
- [6] J. Pearl. Fusion, propagation, and structuring in belief networks. *Artif. Intell.*, 29(3):241–288, 1986. doi:[10.1016/0004-3702\(86\)90072-X](https://doi.org/10.1016/0004-3702(86)90072-X). C100

- [7] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers, 1988. C100, C101
- [8] J. G. Proakis and D. G. Manolakis. *Digital communications*. McGraw-Hill New York, 1995. C105, C107
- [9] International Telecommunication Union. Requirements related to technical performance for IMT-Advanced radio interface(s). Technical report, <http://www.itu.int/pub/R-REP-M.2134-2008>, 2008. C97
- [10] A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE T. Inform. Theory*, 13(2):260–269, 1967. doi:10.1109/TIT.1967.1054010. C99

Author addresses

1. **W. C. Olding**, School of Engineering, University of Tasmania, Hobart, Tasmania 7001, Australia
<mailto:oldingwc@gmail.com>
2. **J. C. Olivier**, School of Engineering, University of Tasmania, Hobart, Tasmania 7001, Australia
3. **B. P. Salmon**, School of Engineering, University of Tasmania, Hobart, Tasmania 7001, Australia