Parallelized Optimization of LDPC Decoding Algorithms

Michael Lunglmayr[1], Jens Berkmann[1], and Mario Huemer[2]

[1] Infineon Technologies AG, Am Campeon 1-12, D-85579 Neubiberg, Germany
{michael.lunglmayr,jens.berkmann}@infineon.com
[2] University of Klagenfurt, Embedded Systems Group, Universitaetsstrasse 65-67, A-9020 Klagenfurt, Austria, mario.huemer@uni-klu.ac.at

Abstract. One of the major problems in communications engineering is to efficiently recover errors of transmitted data blocks by forward error correction (FEC). Low Density Parity Check (LDPC) codes are very popular for FEC. They are known to be one of the best coding schemes in terms of error correction performance. Because the decoding of practically used LDPC codes is NP-hard, only suboptimal algorithms can be applied. To optimize decoding algorithms in terms of decoding performance, a high amount of numerical simulations is needed. We present a parallelized method for evaluating and optimizing the performance of LDPC decoding algorithms. Using performance criteria of the decoding algorithms, the optimization is done by applying nonlinear optimization methods to select algorithm parameters.

1 Introduction

Low Density Parity Check (LDPC) codes [1] became very popular for error correction in the recent years. It was shown that by using LDPC codes, nearly the theoretical bound on the error correction performance – the Shannon limit – can be reached. Because of complexity reasons, only suboptimal decoding algorithms can be used for LDPC codes. Although considerable research has been spent on the theoretical investigation of those decoding algorithms, many open questions concerning the decoding performance in dependence of certain parameters of the decoding algorithm still exist. This lack of understanding complicates the design of efficient decoding algorithms for implementation of an LDPC decoder on a microchip. For this reason often the performance of a decoding algorithm is evaluated by numerical simulations and improved by manually altering the algorithms' parameters. Because of the computational complexity of the decoding algorithms and the requirement of high confidence levels of the result, the evaluation of the decoding performance can be very time-consuming. For an optimization of the decoding algorithms, a high amount of numerical simulations is needed. We will show in this work how the result of performance simulations of decoding algorithms can be used to optimize algorithm parameters and that the use parallel processing can not only dramatically improve the performance of the evaluation but can also make the computational optimization of decoding algorithms feasible in practice.

2 Decoding of LDPC Codes

In a communication system, data is transmitted to a receiver via a communication channel – e.g. a mobile communication channel – which often corrupts the sent data. A simplified schematic of a communication system using an LDPC code is shown in Fig. (a). Data is encoded at the transmitter. This encoding means that a data word is mapped to a codeword, allowing reconstructing the data word at the receiver. This reconstruction is called decoding. The reconstruction is possible because during encoding redundant information is added.

LDPC codes are linear block codes with a sparse $m \times n$ parity check matrix $\mathbf{H}$. This parity check matrix $\mathbf{H}$ can be used to define a code. A binary linear block code can be defined by the vectors $\mathbf{x} \in \{0, 1\}^n$ satisfying the parity check equations $\mathbf{Hx} = \mathbf{0}$ (all calculations are done in the binary field $\mathbb{F}_2$). Every row of $\mathbf{H}$ corresponds to one party check. The parity checks form dependences between the bits of a code word. So the information of one bit can be seen as spread among multiple of bits. This distribution of information is used for decoding.

For decoding of the received data, two approaches are frequently discussed in literature [4]. One approach is called symbol-by-symbol maximum a posteriori (SBS-MAP) decoding. This approach minimizes the bit error probability of the decoded data. The other approach is called maximum likelihood sequence decoding (MLSD). This approach minimizes the word error probability of the decoded data. Unfortunately, both approaches are NP-hard.

For this reason, approximation algorithms are used for decoding of LDPC codes. As an approximation of the SBS-MAP decoding, usually decoding algorithms based on belief propagation are used [2]. For approximation of MLSD decoding an approach based on linear programming can be used [3].

There is a certain lack of analytical understanding on the dependence of the variation of parameters of these decoding algorithms on the decoding performance. For this reason, the parameters of an LDPC coding system are often adjusted manually until a desired performance is obtained. In contrast, we present a method for optimizing decoding algorithms leading to some sort of semi-automatic algorithm design.

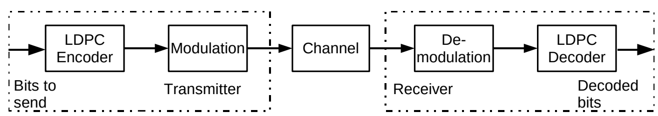
Using the parity check equations, the decoding problem can be represented in two ways suitable for the mentioned decoding algorithms. One is by building a bipartite graph, a so-called Tanner graph [2]. This graph consists of a group of nodes assigned the code bits and a group of nodes assigned to the parity checks and appropriate connections between those groups. Belief propagation decoding can be depicted as passing messages, which correspond to probabilities of bit values, on a Tanner graph. Because of the non-optimality of the algorithm, variations on the scheduling of the messages on the Tanner graph, or the application of correction factors for the messages influence the decoding performance. These parameters can be optimized using the proposed method.

The second representation for the decoding problem is via an integer linear program. The corresponding linear programming relaxation is then used for decoding. The performance of linear programming decoding considerably depends on the used linear programming formulation. Using additional constraints can improve the performance of linear programming decoding [5]. The selection of such constraints can also be done by our optimization system.

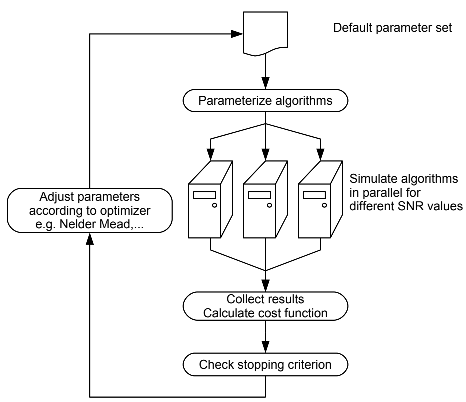
3 Optimization of decoding algorithms

Because of the non-optimality of the algorithms, different algorithm parameters can be adjusted, influencing the decoding performance. In this optimization also other parts of the communication system can be included, as for example the mapping to modulation symbols as shown in [6].

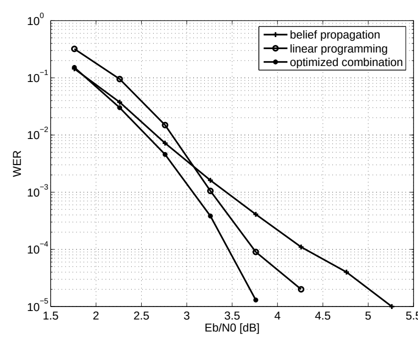
In this work an additive white Gaussian noise (AWGN) channel is used. This channel is defined by one parameter, the signal-to-noise ratio (SNR). A major



(a) Schematic drawing of a communication system with LDPC codes



(b) Schematic overview of the optimization system



(c) Word error rate for optimized decoding algorithm

problem is that a parameter set often performs differently for different channel SNR values. To do an optimization over a practically relevant SNR range, many simulations must be done with the same parameter set but with different channel SNR values.

For the optimization of decoding algorithms we used the structure as depicted in Fig. (b). The optimization starts with a default parameter set of the algorithm. The algorithms are parameterized with the parameter set and then simulated in parallel for different channel SNR values. To get high confidence levels of the simulation results, up to 10^9 code words (data blocks) have to be simulated. This is also easily parallelizable, because the decoding of a received data word can be done independently of other data words. The results of the performance simulations are then used to assign certain costs to a parameter set. Instead of using the bit and word error rates (BER/WER) directly, an error rate gain at specified SNR values is used as a performance measure. Let P be a parameter set of a decoding algorithm. We define the costs assigned to this parameter set as

$$c(P) = \sum_s \mathrm{WER}_{\mathrm{ref}}(s) - \mathrm{WER}(s). \tag{1}$$

This means that at specified SNR values s the word error rate obtained by the decoding algorithm using a parameter set P is compared to an adequate reference $\mathrm{WER}_{\mathrm{ref}}$ (e.g. a non-optimized version of the algorithm). This allows optimizing the algorithm for adequate SNR ranges: the higher the sum of the WER gains the better the parameter set. The optimization algorithm - e.g. a Nelder-Mead downhill simplex algorithm - then uses these costs to find the parameter set with the highest WER gain.

Instead of only mutually exclusively using BP and LP algorithms, we also optimized a novel combination of both algorithms. The results of both algorithms are combined using a weight function found by the optimization system.

4 Simulation Results

In Fig. (c) we show the simulation results using the optimized version of a combined LP/BP Algorithm and the results of the single algorithms for an example LDPC code. The combined algorithm performs significantly better in terms of lowering the word error rate than the single algorithms especial in the high SNR range.

References

1. R. G. Gallager, *Low-Density Parity-Check Codes.* Cambridge, MA: MIT Press, 1963.
2. F. Kschischang, B. Frey, H. Loeliger, "Factor graphs and the sum-product algorithm", in *IEEE Transactions on Information Theory*, Vol. 51, Iss. 3, pp. 954-972, March 2005.
3. J. Feldman, M. Wainwright, D. Karger, "Using linear programming to Decode Binary linear codes", in *IEEE Transactions on Information Theory*, Vol. 47, Iss. 2, pp. 498-519, Feb. 2001. 2003.
4. T. K. Moon, *Error Correcting Coding: Mathematical Methods and Algorithms.* New York: Wiley, 2005.
5. M. Taghavi, P. Siegel, "Adaptive Methods for Linear Programming Decoding", online at: http://arxiv.org/abs/cs/0703123
6. M. Lunglmayr, J. Berkmann, "Optimized Mapping Schemes for LDPC Coded Higher Order Modulated QAM Transmission", in *Lecture Notes in Computer Science - EUROCAST 2007*, Vol. 4739, pp. 952-959, Springer, 2007.