This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2008 proceedings.

# Graph-Based Soft Channel and Data Estimation for MIMO Systems with Asymmetric LDPC Codes

Tianbin Wo, Chunhui Liu†, and Peter Adam Hoeher
Faculty of Engineering
University of Kiel
Kaiserstr. 2, 24143 Kiel, Germany
Email: wtb@tf.uni-kiel.de, liu@ti.rwth-aachen.de, ph@tf.uni-kiel.de

***Abstract*— In this paper, we propose an iterative soft channel estimation and data detection algorithm based on a factor graph. Channel coefficients as well as data symbols are treated as variable nodes and are all estimated in a low-complexity element-wise manner. Applying asymmetric LDPC codes, this algorithm is able to deliver ambiguity-free outputs for MIMO systems with or without training symbols. Training symbols are inherently utilized as a type of a priori information. This algorithm thoroughly relaxes the troublesome constraints on training design in the sense that an arbitrary (even zero) number of training symbols can be placed at arbitrary positions within a data burst.**

## I. Introduction

Consider a MIMO system which applies spatial multiplexing. The receiver is expected to separate the data streams from multiple antennas and to provide reliable data estimates. Accurate channel knowledge is the key to accomplish these two challenging tasks, since different data streams are distinguished via different channel coefficients associated with them. As channel coefficients are usually estimated from training (pilot) symbols/sequences, the system robustness relies heavily on the training. Due to this reason, training design is always critical yet often energy-taking. Besides, matrix-inversion-based channel estimation algorithms commonly require training symbols to be consecutively located. This, however, highly restricts our freedom in designing and utilizing training symbols, e.g., in case of fast fading channels, it would be nice to distribute training symbols throughout the burst instead of concentrating them into a preamble or midamble.

From an information theoretic point of view, it is highly suboptimal to perform pure training-based channel estimation [1]. In this paper, we will try to exploit the potential of channel coding to eliminate systems dependence on training. We apply asymmetric LDPC codes [2] to remove phase ambiguity, and use antenna-specific interleaver patterns to accomplish layer separation. The use of training is no longer mandatory, and all strict requirements on training design are eliminated. A low-complexity graph-based iterative soft estimation algorithm is proposed. To enable the system to start from a totally blind state, we also introduce the concept of soft channel estimation.

This work has been supported by the German Research Foundation (DFG) under contract no. HO 2226/10-1.

†Mr. Chunhui Liu is now a Ph.D. student at RWTH Aachen University, Germany.

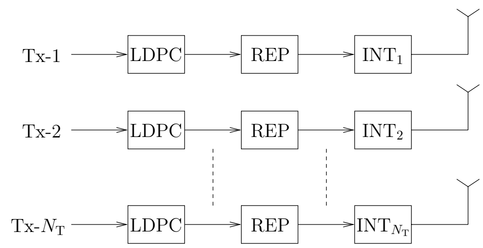


Fig. 1. Transmitter structure

## II. System Model

### A. Channel Model

Let $N_R$ denote the number of receive (Rx) antennas and $N_T$ the number of transmit (Tx) antennas. The equivalent discrete-time model of an MIMO channel is given by

$$y_n[k] = \sum_{m=1}^{N_T} h_{n,m}[k] x_m[k] + w_n[k] \ , \qquad (1)$$

where $k \in \{0, 1, \ldots, K-1\}$ is the discrete time index with $K$ denoting the burst length. $y_n[k] \in \mathbb{C}$ is the channel output at the $n$-th ($1 \leq n \leq N_R$) Rx antenna at time index $k$, and $x_m[k] \in \{\pm 1\}$ is the BPSK channel input at the $m$-th Tx antenna at time index $k$. $h_{n,m}[k] \in \mathbb{C}$ marks the coefficient of the sub-channel connecting the $n$-th Rx antenna and the $m$-th Tx antenna at time index $k$. For the sake of simplicity, block fading is assumed within this paper, i.e., the channel keeps constant within each data burst while it varies independently from burst to burst. $w_n[k]$ represents an additive white Gaussian noise (AWGN) sample with zero mean and variance $\sigma_w^2$.

### B. Transmitter Structure

The adopted transmission scheme is illustrated in Fig. 1. Channel coding is done separately on each transmit antenna. Each info bit stream is first encoded by an asymmetric LDPC code in order to benefit from coding gain and to eliminate the phase ambiguity. Afterwards, repetition encoding together with random interleaving is applied. To eliminate the potential permutation ambiguity, the interleaver patterns must be antenna-specific. Using asymmetric LDPC to avoid phase ambiguity is a relatively new idea [2], whereas using different interleaver patterns to separate superimposed data streams is a well-proven method from the field of interleave-division multiple access (IDMA) [3].

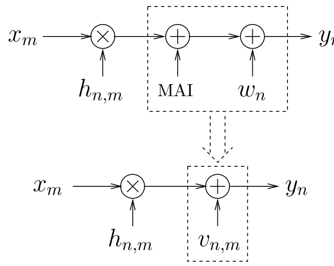


Fig. 2. Relationship between a data symbol and its observation

## III. Preliminary Remarks

To enhance the readability of the algorithm description, we introduce several essential concepts in advance.

### A. Gaussian Approximation

Considering an arbitrary symbol $x_m$ (index $k$ is omitted for simplicity) and one of its observations $y_n$, we have

$$\begin{aligned} y_n &= \textstyle\sum_{i=1}^{N_T} h_{n,i} x_i + w_n \\ &= h_{n,m} x_m + \underbrace{\textstyle\sum_{i=1, i\neq m}^{N_T} h_{n,i} x_i}_{\text{MAI}} + \underbrace{w_n}_{\text{AWGN}} \ , \end{aligned} \quad (2)$$

where MAI stands for multi-antenna interference. We define $v_{n,m}$ as the effective noise sample in the observation $y_n$ w.r.t. the symbol $x_m$:

$$v_{n,m} \doteq y_n - h_{n,m} x_m \ , \quad (3)$$

as illustrated in Fig. 2. If we approximate the probability density function (PDF) of $v_{n,m}$ by $\mathcal{CN}(\mu_{v_{n,m}}, \sigma^2_{v_{n,m}})$ with

$$\begin{aligned} \mu_{v_{n,m}} &\doteq \mathrm{E}\{v_{n,m}\} \\ \sigma^2_{v_{n,m}} &\doteq \mathrm{E}\left\{|v_{n,m} - \mu_{v_{n,m}}|^2\right\} \ , \end{aligned} \quad (4)$$

the computation of likelihood function can be dramatically simplified into

$$p(y_n \,|\, x_m) \approx \frac{1}{\pi \sigma^2_{v_{n,m}}} \exp\Big( - \frac{|y_n - h_{n,m}\, x_m - \mu_{v_{n,m}}|^2}{\sigma^2_{v_{n,m}}} \Big) \ .$$

### B. Soft Channel Estimation

The adjective "soft" here has a two-fold meaning. It means that we should utilize soft data information for channel estimation, and the channel estimator must provide soft channel information as well. In general, it is difficult to name a metric to describe the reliability of a channel estimate. Nevertheless, if the additive noise has a Gaussian distribution, it is indeed very easy to represent soft channel information. Given an unbiased estimate $\hat{h}$, the PDF of $h$ will be a Gaussian function with mean $\mu_h = \hat{h}$, as illustrated in Fig. 3. The variance $\sigma_h^2$

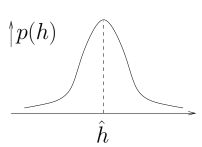


Fig. 3. Probability density function of $h$

of $h$ will be determined by the noise and signal power. In this case, we may say that $\sigma_h^2$ carries the reliability information of $\hat{h}$. Certainly, the smaller $\sigma_h^2$ is, the more reliable $\hat{h}$ is.

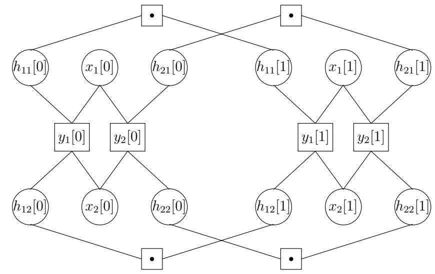


Fig. 4. Factor graph of a MIMO channel with $N_T = N_R = 2$

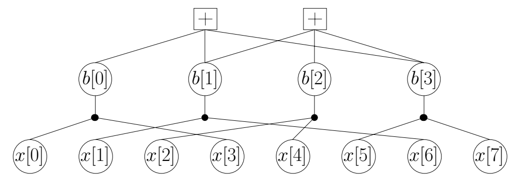


Fig. 5. Factor graph of concatenated LDPC code and repetition code

### C. Factor Graph

Admitting the fact that uncertainties exist in channel coefficients, a general factor graph of a MIMO channel should include channel coefficients as variable nodes as well. Making the following independence approximation[1]

$$p(\mathbf{Y} \,|\, x_m[k]) \approx \prod_{n=1}^{N_R} p(y_n[k] \,|\, x_m[k]) \ , \quad (5)$$

where $\mathbf{Y}$ is the matrix which collects all channel outputs of the current data burst, a factor graph of a MIMO channel will look like Fig. 4. The mark $\boxdot$ stands for the relationship between $h_{n,m}[k]$ and $h_{n,m}[k+1]$. Since we assume that the channel is constant within each data burst, the relationship between $h_{n,m}[k]$ and $h_{n,m}[k+1]$ is simply an equality. Nevertheless, for fast fading channels, this $\boxdot$ should be a suitable transfer function which describes the degree of variation of a channel coefficient between two neighboring time indices.

Concerning the concatenation of an LDPC code and a repetition code together with interleaver, the corresponding factor graph will look like Fig. 5, where each filled circle marks a repetition code node and each box-plus sign denotes a parity check node.

### D. Asymmetric LDPC Code

A parity check is called asymmetric if it is connected with an odd number of summands. For example, [011] is a valid solution for the following parity check sum

$$b_1 \oplus b_2 \oplus b_3 = 0 \ , \quad (6)$$

but its negation [100] is an invalid solution due to the odd amount of summands. An LDPC code consisting of asymmetric parity checks is called an asymmetric LDPC code. The percentage of its parity checks being asymmetric gives the degree of asymmetry. If all parity checks of an LDPC code are asymmetric, phase ambiguity incurred by uncertain channel coefficients can be easily eliminated by using such a code.

[1]This approximation is made to reduce the data detection complexity. The authors would refer interested readers to [4], [5] for detailed explanation on this issue.

## IV. Graph-Based Iterative Gaussian Detection

Due to the common relationships between data symbols, channel coefficients, and channel observations, a more abstract notation is adopted for ease of explanation. In the following, we use $y$ to denote an observation node, $x$ to denote a symbol node, and $h$ to denote a channel coefficient node. The complicated indices $n$, $m$, and $k$ are in general replaced by a single index $i$.

### A. Starting Point

We assume that all channel coefficients are zero-mean Gaussian distributed, and all subchannels have the same average power. This assumption is valid for most of the practical applications. Note that this assumption does not have to be accurate since this initial setup is discarded as soon as the algorithm starts to run. In case that no training symbols are transmitted at all, the initial mean value of all channel coefficients should be set to a tiny non-zero value in order to start the algorithm, which is a common way of launching a blind detection algorithm.

### B. Message Update Rule at Observation Nodes

Revisiting Fig. 4, we will find that the relationship between an observation node and its associated variable nodes can be written as

$$y = \sum_{i=1}^{Q} h_i x_i + w, \tag{7}$$

where $Q = N_T$ is the amount of associated symbol nodes or channel coefficient nodes, $h_i$ is the channel coefficient linking $y$ and $x_i$, and $w$ is the additive white Gaussian noise sample. In each iteration, one observation node will receive messages from its variable nodes in the form of probability functions. Then new messages are generated and redistributed to these variable nodes according to the Turbo principle, that is only extrinsic informationshould be exchanged. A schematic diagram of message propagation is given in Fig. 6.

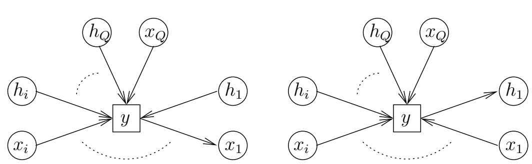

Fig. 6. Message exchange at an observation node

*1) Soft Data Detection with Soft Channel Information:* Based on (7), we define $v_m$ ($1 \leqslant m \leqslant Q$) as the effective noise sample in the observation $y$ w.r.t. the symbol $x_m$:

$$v_m \doteq y - h_m x_m = \sum_{i=1, i\neq m}^{Q} h_i x_i + w \ . \tag{8}$$

Approximating $p(v_m)$ as $v_m \sim \mathcal{CN}(\mu_{v_m}, \sigma^2_{v_m})$, and knowing that $h_m \sim \mathcal{CN}(\mu_{h_m}, \sigma^2_{h_m})$, the log-likelihood ratio (LLR) of symbol $x_m$ is calculated as

$$\mathrm{LLR}(x_m) = 4\ \mathrm{Re}\{\mu^*_{h_m}(y - \mu_{v_m})\}/(\sigma^2_{h_m} + \sigma^2_{v_m}) \ . \tag{9}$$

In each iteration, the LLR of each symbol is calculated according to (9) and distributed over the factor graph. Interested readers may find the mathematical derivation of (9) in [6].

Let $\mathrm{P}(x_i = \pm 1)$ be denoted by $\mathrm{P}_{i,\pm 1}$ and note that $h_i$ and $x_i$ are statistically independent, the PDF of $h_i x_i$ will be

$$p(h_i x_i) = \mathrm{P}_{i,+1} \cdot p(h_i) + \mathrm{P}_{i,-1} \cdot p(-h_i) \ , \tag{10}$$

which is indeed a mixed Gaussian function with two peaks. Applying $h_i \sim \mathcal{CN}(\mu_{h_i}, \sigma^2_{h_i})$ and after some mathematical derivation, we will obtain

$$\begin{aligned} \mu_{h_i x_i} &= \mu_{h_i}(\mathrm{P}_{i,+1} - \mathrm{P}_{i,-1}) \\ \sigma^2_{h_i x_i} &= \sigma^2_{h_i} + 4\ \mathrm{P}_{i,+1}\mathrm{P}_{i,-1}|\mu_{h_i}|^2 \ . \end{aligned} \tag{11}$$

Finally, the mean and variance of $v_m$ can be calculated as

$$\begin{aligned} \mu_{v_m} &= \textstyle\sum_{i=1, i\neq m}^{Q} \mu_{h_i x_i} \\ \sigma^2_{v_m} &= \textstyle\sum_{i=1, i\neq m}^{Q} \sigma^2_{h_i x_i} + \sigma^2_w \ . \end{aligned} \tag{12}$$

As a matter of fact, the effective noise sample $v_m$ is a summation of $Q-1$ independent mixed Gaussian variables plus one independent Gaussian variable. Since all these component variables are continuously valued, the Gaussian approximation of $v_m$ is pretty good even at high SNRs.

*2) Soft Channel Estimation with Soft Data Information:* Let us rewrite (8) into

$$y = h_m x_m + v_m \ . \tag{13}$$

The information of $h_m$ contained in $y$ is fully represented by the conditional probability density function $p(y|h_m)$, which may be computed as follows:

$$\begin{aligned} p(y|h_m) &= \sum_{x_m \in \{\pm 1\}} p(y|h_m, x_m)\mathrm{P}(x_m) \\ &= \mathrm{P}_{m,+1} \frac{1}{\pi\sigma^2_{v_m}} \exp\left(-\frac{|h_m - (y - \mu_{v_m})|^2}{\sigma^2_{v_m}}\right) + \\ &\quad\ \mathrm{P}_{m,-1} \frac{1}{\pi\sigma^2_{v_m}} \exp\left(-\frac{|h_m + (y - \mu_{v_m})|^2}{\sigma^2_{v_m}}\right) . \end{aligned} \tag{14}$$

Excluding a priori information and after considering the issue of normalization, the following statement holds:

$$p(h_m) = p(y|h_m) \ . \tag{15}$$

Clearly, it is again a mixed Gaussian function, which is troublesome to be utilized in the stage of data detection, particularly when the data symbol $x_m$ is of higher-order modulation formats other than BPSK. Therefore, suitable approximation is necessary to simplify this channel knowledge. Note that if the data detection is carried out successfully, we will have

$$\mathrm{P}_{m,+1} \gg \mathrm{P}_{m,-1} \tag{16}$$

or

$$\mathrm{P}_{m,+1} \ll \mathrm{P}_{m,-1} \tag{17}$$

as the iterations go on. Therefore, it is reasonable to make the approximation $h_m \sim \mathcal{CN}(\mu_{h_m}, \sigma^2_{h_m})$ with

$$\begin{aligned} \mu_{h_m} &= (y - \mu_{v_m})(\mathrm{P}_{m,+1} - \mathrm{P}_{m,-1}) \\ \sigma^2_{h_m} &= \sigma^2_{v_m} + 4\mathrm{P}_{m,+1}\mathrm{P}_{m,-1}|y - \mu_{v_m}|^2 \ , \end{aligned} \tag{18}$$

which are calculated according to (14) and (15). Relevant discussions can be also found in [7].

### *C. Message Update Rule at Channel Coefficient Nodes*

For block-fading channels, the channel coefficients keep constant within each data burst. Therefore, each channel coefficient node is associated with $Q = K$ observation nodes as depicted in Fig. 7. Similar with the message updating

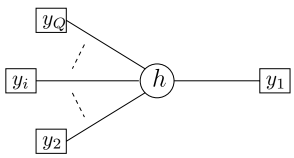


Fig. 7. A channel coefficient node and its observation nodes

at observation nodes, only extrinsic information should be exchanged at channel coefficient nodes. For example, if a channel coefficient node receives PDF messages from its observation nodes as shown in the left part of Fig. 8, then the updated messages are generated in a way shown in the right part of Fig. 8. Note that the product of two Gaussian PDFs

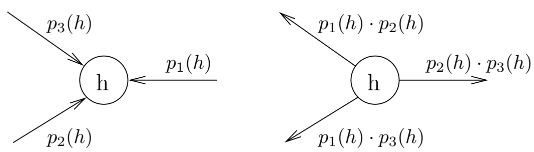


Fig. 8. Message exchange at a channel coefficient node

gives a new Gaussian PDF. Suppose we have the following two messages:

$$\begin{aligned} p_1(h) &: \quad h \sim \mathcal{CN}(\mu_1, \sigma_1^2) \\ p_2(h) &: \quad h \sim \mathcal{CN}(\mu_2, \sigma_2^2) \,, \end{aligned} \tag{19}$$

then the product of these two messages will be given by

$$p_1(h) \cdot p_2(h) \quad : \quad h \sim \mathcal{CN}(\mu_h, \sigma_h^2) \tag{20}$$

with

$$\begin{aligned} \mu_h &= (\sigma_2^2\mu_1 + \sigma_1^2\mu_2)/(\sigma_1^2 + \sigma_2^2) \\ \sigma_h^2 &= \sigma_1^2\sigma_2^2/(\sigma_1^2 + \sigma_2^2) \,. \end{aligned} \tag{21}$$

Therefore, we can still use mean value and variance to represent the newly generated message. As no extra effort is needed, this operation can be repeated iteration by iteration.

### *D. Message Update Rules at Symbol and Code Nodes*

The message update rules at symbol nodes, parity check code nodes, and repetition code nodes can be easily derived according to the principle of belief propagation. Due to limited space, we would refer interested readers to [8][9] for detailed discussions on this topic.

### *E. Scheduling and Phase Correction*

If training symbols (no matter how many) are available, an initial channel estimation is performed before the first iteration. The observation nodes connected with training symbols deliver messages to channel coefficient nodes in order to provide a reasonable starting point for the iterative processing algorithm. If there are no training symbols, then this initial channel estimation is simply skipped. Afterwards in each iteration, the message updating operation is performed once per node. Using the same analysis as in [4], we will find that the complexity of this algorithm is strictly linear in the number of transmit/receive antennas and the spreading factor.

A special procedure in this algorithm is the phase correction operation. If the estimated data stream from a particular transmit antenna violates most of the associated parity checks, then with high probability this estimated data stream is negated in phase. In this case, messages generated at all nodes belonging to this specific transmit antenna should be reversed in phase. Afterwards, these phase-corrected messages will be propagated to the parity checks to continue the iterative processing.

In case of completely phase-negated soft inputs, an asymmetric LDPC decoder will mostly fall into a local optimum insteal of a global optimum. Therefore, we should not purely rely on iterative decoding to correct the phase. In the first global iteration, the phase correction strategy described above should be used, and it proves to be helpful to carry out several local iterations among nodes except LDPC nodes. Whereas in the later iterations, we can perform iterative message passing among all nodes just in a normal way.

## V. Numerical Results

Since the algorithm is designed to work for systems with or without training symbols, we test its performance under different training lengths. We use $K_{\mathrm{T}}$ to denote the number of training symbols per burst per antenna. Note that, with the proposed algorithm, training symbols do not have to be consecutively placed. They can for example spread over the whole burst in order to track fast fading channels. The a priori knowledge of training symbols is utilized in an element-wise manner within this algorithm.

### *A. Simulation Setup*

For the numerical results provided in this section, a block-Rayleigh-fading MIMO channel model with $8$ transmit antennas and $8$ receive antennas is used. The coefficient of each subchannel is normalized to have an average power of $1$. The SNR per info bit $E_b/N_0$ is calculated as $1/\sigma_w^2$, where $\sigma_w^2$ denotes the variance of the additive noise.

Channel coding is done separately at each transmit antenna. A binary regular $(3, 5, 200, 80)$ LDPC code is applied. The parity check matrix has a uniform column weight of $3$, and a uniform row weight of $5$ which makes the code strongly asymmetric. The code word length is given by $200$, and the info word length is given by $80$. During simulations, the parity check matrix is randomly generated without optimization.

The rate of the repetition code is set to be $1/4$. Scrambling with fixed pattern is applied, that is, every second bit of a code word is flipped. Scrambling is beneficial for the proposed algorithm since it assumes that all data symbols come with zero mean. Antenna-specific random interleaving is applied to enhance the symbols independence as well as to eliminate the potential permutation ambiguity.

Finally, the number of global iterations of the algorithm is fixed to be $5$. In the following, we will call the proposed algorithm as graph-based iterative Gaussian detector (GIGD).

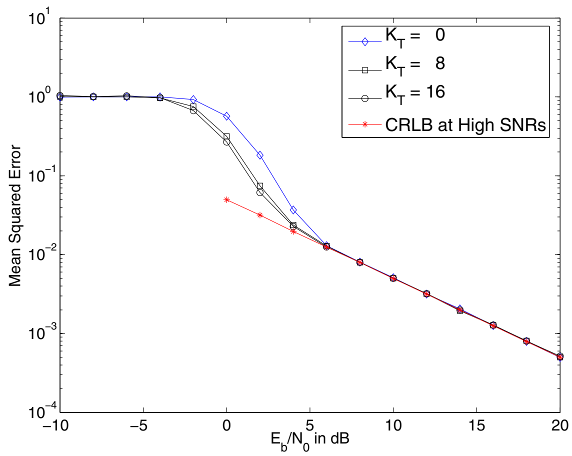


Fig. 9. MSE vs. $E_b/N_0$, $N_T = N_R = 8$

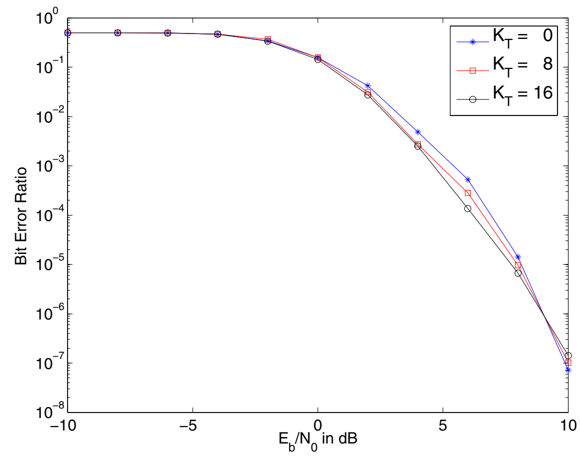


Fig. 10. BER vs. $E_b/N_0$, $N_T = N_R = 8$

### *B. Mean Squared Error*

For unbiased semi-blind channel estimation [10][11], the theoretical tight bound of MSE is given by the Cramer-Rao lower bound (CRLB) [12]. Therefore, in case of having training symbols, the channel estimation performance of GIGD will be bounded by the CRLB. Fig. 9 compares the MSE curves of GIGD with the corresponding bound. Surprisingly, GIGD approaches the CRLB at high SNRs even under $K_{\mathrm{T}} = 0$, i.e., in a totally blind mode. This amazing performance verifies the efficiency of element-wise channel estimation via belief propagation. Although no matrix inversion is applied at all, the resulting performance still achieves the optimum.

For the three curves in Fig. 9, the only different parameter is $K_{\mathrm{T}}$ - the number of training symbols per burst. As we can see, longer training sequences are helpful at low SNRs, but bring no benefit at high SNRs. We should also keep in mind that longer training always means less signal power and lower spectral efficiency. According to Fig. 9, the optimal training length is 0 for SNRs $\geq 6$ dB, which deserves to be an interesting phenomenon from an information theoretic point of view.

### *C. Bit Error Ratio*

To further verify the performance of GIGD, we check its BER performance for different amounts of training symbols. As training symbols appropriate the energy from data symbols and they do not help in channel estimation, we may predict that GIGD should achieve the best performance at $K_{\mathrm{T}} = 0$ in the high SNR range. Fig. 10 confirms our conjecture. GIGD does deliver the lowest BER in a totally blind mode for high SNRs, which is difficult to imagine for conventional receiver algorithms. This observation indeed reveals the power of channel coding in eliminating detection ambiguities. Furthermore, it shows the promise of message passing algorithms based on factor graph. Unlike conventional blind estimation algorithms, GIGD does not rely on the whiteness of data streams. Consequently, it is not vulnerable to the temporal correlation introduced by channel coding. As a matter of fact, GIGD fully exploits the redundancy from channel codes to deliver ambiguity-free outputs.

## VI. Conclusions

In this paper, a graph-based iterative soft channel and data estimation algorithm is proposed. Channel coefficients are estimated element-wise via belief propagation over a factor graph. An asymmetric LDPC code is used to eliminate the phase ambiguity, and a repetition code together with interleaving is used to eliminate the permutation ambiguity. By exploiting the potential of channel coding, this algorithm is able to deliver ambiguity-free outputs with arbitrary length of training. For high SNRs, this algorithm achieves the best performance without using any training symbols.

## References

[1] B. Hassibi and B.M. Hochwald, "How much training is needed in multiple-antenna wireless links?" *IEEE Trans. Inform. Theory*, vol. 49, no. 4, pp. 951-963, Apr. 2003.

[2] A. Scherb and K.-D. Kammeyer, "Non-Coherent LDPC Decoding on Graphs," in *Proc. IEEE Inform. Theory Workshop (ITW)*, Chengdu, China, Oct. 2006.

[3] P.A. Hoeher, H. Schoeneich, and J.Ch. Fricke, "Multi-layer interleave-division multiple access: Theory and practice," accepted for publication in *European Trans. Telecommun. (ETT)*, vol. 19, no. 5, Aug. 2008.

[4] T. Wo and P.A. Hoeher, "A simple iterative Gaussian detector for severely delay-spread MIMO channels," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Glasgow, Scotland, June 2007.

[5] T. Wo, J.Ch. Fricke, and P.A. Hoeher, "A graph-based iterative Gaussian detector for frequency-selective MIMO channels," in *Proc. IEEE Inform. Theory Workshop (ITW)*, Chengdu, China, Oct. 2006.

[6] T. Wo, C. Liu, and P.A. Hoeher, "Graph-based iterative Gaussian detection with soft channel estimation for MIMO systems," in *Proc. ITG Int. Conf. on Source and Channel Coding*, Ulm, Germany, Jan. 2008.

[7] M. Tüchler, R. Otnes, and A. Schmidbauer, "Performance of soft iterative channel estimation in Turbo equalization," in *Proc. IEEE Int. Conf. Commun. (ICC)*, New York, USA, Apr. 2002.

[8] F.R. Kschischang, J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498-519, Feb. 2001.

[9] R.G. Gallager, *Low Density Parity Check Codes*, Monograph, M.I.T. Press, 1963.

[10] C. Cozzo and B.L. Hughes, "Joint channel estimation and data detection in space-time communications," *IEEE Trans. Commun.*, vol. 51, pp. 1266-1270, Aug. 2003.

[11] M. Loncar, R.M.J. Wehinger, and T. Abe, "Iterative joint detection, decoding, and channel estimation for dual antenna arrays in frequency selective fading," in *Proc. 5th Int. Symp. on Wireless Personal Multimedia Communication*, pp. 125-129, Honolulu, USA, Oct. 2002.

[12] T. Wo, A. Scherb, P.A. Hoeher, and K.-D. Kammeyer, "Analysis of semiblind channel estimation for FIR-MIMO systems," in *Proc. 4th Int. Symp. on Turbo Codes & Related Topics*, Munich, Germany, Apr. 2006.