

RESEARCH

Open Access

Reduced complexity Log-MAP algorithm with Jensen inequality based non-recursive \max^* operator for turbo TCM decoding

Michal Sybis and Piotr Tyczka*

Abstract

In this paper, a reduced complexity Log-MAP algorithm based on a non-recursive approximation of the \max^* operator is presented and studied for turbo trellis-coded modulation (TTCM) systems. In the algorithm, denoted as AvN Log-MAP, the \max^* operation is generalized and performed on $n \geq 2$ arguments. The approximation is derived from the Jensen inequality. The non-recursive form of the \max^* calculations allows to achieve significant reduction in the decoding effort in comparison to the conventional Log-MAP algorithm. Bit-error rate performance simulation results for serial and parallel TTCM schemes in the additive white Gaussian noise and uncorrelated Rayleigh fading channels show that the AvN Log-MAP algorithm performs close to the Log-MAP. Performance and complexity comparisons of the AvN Log-MAP algorithm against the Log-MAP and several relevant reduced complexity turbo decoding algorithms proposed in the literature reveal, that it offers favorable low computational effort for the price of small performance degradation.

Keywords: Decoding algorithms; Log-MAP algorithm; Reduced-complexity algorithms; \max^* operator; Jacobian logarithm; Turbo TCM

1 Introduction

The invention of turbo codes and their iterative (turbo) decoding principle [1,2] has opened new perspectives on digital transmission and receiver design. Owing to the excellent performance, turbo codes have been extensively studied and have found applications in various wireless communication systems. Moreover, several enhanced turbo-like schemes have been proposed in recent years, with the aim of improving the overall bandwidth efficiency. On the other hand, the idea of iterative processing has found widespread applications not only in error control coding but also in other areas of digital communications, such as detection, interference suppression, equalization and synchronization. Nowadays, iterative processing has become prevalent in state-of-the-art receiver design.

It is known that in turbo decoding, the optimal algorithm for the soft-input/soft-output (SISO) component

decoders is the *maximum a posteriori* (MAP) probability algorithm [3]. In practice, in order to reduce numerical computation problems, the MAP algorithm is implemented in the logarithmic domain that results in the so-called Log-MAP algorithm [4,5]. The core operation of the Log-MAP is the calculation of the logarithm of the sum of exponential terms, denoted as \max^* operator, using the so-called Jacobian logarithm [6]. In recent years, several algorithmic approaches have been proposed aiming for a simplification of the \max^* operator and thus reducing the implementation complexity of the SISO decoders without a substantial loss of decoding performance (e.g., [7-13]). In all these algorithms, a conventional \max^* operator, i.e., defined for $n = 2$ input values, is modified, and for $n > 2$, these approximations are applied recursively $n-1$ times. Generalized non-recursive approximation methods for \max^* operator with $n > 2$ arguments have been recently presented in [14] and [15].

In this paper, we revisit a reduced-complexity Log-MAP algorithm based on a non-recursive approximation of the \max^* operator with $n \geq 2$ input values, which has been recently proposed in [16]. The novel approximation

*Correspondence: tyczka@et.put.poznan.pl
Chair of Wireless Communications, Poznan University of Technology, ul.
Polanka 3, 60-965 Poznan, Poland

given in [16], is derived from the Jensen inequality [17] and allows to achieve significant reduction in the decoding effort in comparison to the conventional Log-MAP algorithm. The purpose of this paper is to expand the initial work reported in [16] and to present comprehensive performance and complexity results and analysis of the algorithm, hereafter called the AvN Log-MAP algorithm (AvN is the acronym of 'Average with the parameter N '), for turbo trellis-coded modulation (TTCM) schemes [18,19]. Performance evaluation results of the AvN Log-MAP algorithm are presented for both parallel and serial TTCM schemes in the additive white Gaussian noise (AWGN) and uncorrelated (i.e., fully interleaved) Rayleigh fading channels. For TTCM, the trellises are non-binary and hence, in the Log-MAP operation all variables are calculated by means of the \max^* operator with $n > 2$ arguments. Bit-error rate (BER) results of the AvN Log-MAP algorithm are compared with the results of the Log-MAP and some relevant reduced complexity decoding algorithms, namely Linear Log-MAP [8], Average Log-MAP [9], Shift Log-MAP [10] and Max-Log-MAP [4]. Complexity comparison of the AvN Log-MAP algorithm against those algorithms for the investigated TTCM schemes is also given. The paper is organized as follows. In section 2, for the sake of completeness, we review the derivation of the non-recursive \max^* approximation for the AvN Log-MAP algorithm, given in [16]. Section 3 presents the performance evaluation results and their discussion. Complexity comparison of the algorithms is covered in section 4. Finally, section 5 contains concluding remarks.

2 Non-recursive \max^* approximation based on the Jensen inequality

In the Log-MAP algorithm, the calculation of the soft output as well as the forward and backward metrics and the branch metrics of trellis transitions requires computation of the \max^* operator defined as

$$\max^*(x_1, x_2, \dots, x_n) = \ln \left(\sum_{i=1}^n e^{x_i} \right) \quad (1)$$

An exact solution to this problem, used in the Log-MAP algorithm, is the application of the Jacobian logarithm

$$\begin{aligned} \max^*(x_1, x_2) &= \max(x_1, x_2) + \ln(1 + e^{-|x_2 - x_1|}) \\ &= \max(x_1, x_2) + f_c(|x_2 - x_1|) \end{aligned} \quad (2)$$

where $f_c(\cdot)$ is a correction function, usually implemented with an eight-element look-up table (LUT) [5]. To obtain the \max^* operator for more than two arguments, i.e., $n > 2$, the Jacobian logarithm (2) is applied recursively $n - 1$ times. For example, assuming $n = 3$, it yields

$$\max^*(x_1, x_2, x_3) = \max^*(\max^*(x_1, x_2), x_3). \quad (3)$$

In the approach taken in [16], in order to reduce the computational effort, the \max^* operator with $n \geq 2$ arguments is approximated directly, that is, without recursive computations required for the Jacobian logarithm, as it is shown in (3). The motivation for such a treatment comes from the observation that the recursion performed in the Log-MAP to calculate the Jacobian logarithm for $n > 2$ arguments has a major influence on the complexity of this algorithm.

In [16], the novel approximation is derived from two inequalities. The first originates from the definition of the \max^* operator given in (1). Identifying that $x_1 = \max(x_1, x_2, \dots, x_n)$, we have

$$\begin{aligned} \max^*(x_1, x_2, \dots, x_n) &= \ln(e^{x_1}) + \ln \left(1 + \sum_{i=2}^n e^{x_i - x_1} \right) \\ &= \max(x_i) + \ln \left(1 + \sum_{i=2}^n e^{x_i - x_1} \right) \end{aligned} \quad (4)$$

Since the second term of the right-hand side (RHS) in (4) can be treated as a correcting term and is always greater than zero, we obtain

$$\max^*(x_1, x_2, \dots, x_n) \geq \max(x_1, x_2, \dots, x_n) \quad (5)$$

Note that the equality in (5) corresponds to the Max-Log-MAP algorithm.

For the second inequality, consider the Jensen inequality

$$\frac{\sum_{i=1}^n \alpha_i}{n} \geq \left(\prod_{i=1}^n \alpha_i \right)^{\frac{1}{n}}, \quad \alpha_i > 0. \quad (6)$$

By substituting $\alpha_i = e^{x_i}$ and taking the logarithm of the two sides of (6), we have

$$\ln \left(\sum_{i=1}^n e^{x_i} \right) \geq \ln \left(n \cdot \left(\prod_{i=1}^n e^{x_i} \right)^{\frac{1}{n}} \right) \quad (7)$$

Identifying from (1) that $\ln \left(\sum_{i=1}^n e^{x_i} \right) = \max^*(x_1, x_2, \dots, x_n)$ we may rewrite (7) as

$$\max^*(x_1, x_2, \dots, x_n) \geq \ln(n) + \frac{1}{n} \ln \left(\prod_{i=1}^n e^{x_i} \right) \quad (8)$$

and further

$$\max^*(x_1, x_2, \dots, x_n) \geq \ln(n) + \frac{1}{n} \sum_{i=1}^n \ln e^{x_i} \quad (9)$$

Since $\ln(n)$ in (9) is a positive constant value that does not depend on x_i , it can be omitted and hence, we obtain the final inequality

$$\max^*(x_1, x_2, \dots, x_n) \geq \frac{1}{n} \sum_{i=1}^n x_i \quad (10)$$

It can be easily noticed that the RHS in (10) is the average value of input values of the \max^* operator.

As it can be seen in (5) and (10), the exact value of the \max^* operator in both inequalities is greater than the RHS values. In fact, we may think of the RHSs of (5) and (10) as lower bounds to the \max^* operator. Merging inequalities (5) and (10), we may write

$$\max^*(x_1, x_2, \dots, x_n) \approx \max\left(\max_{i=1:n}(x_i), \frac{1}{n} \sum_{i=1}^n x_i\right). \quad (11)$$

Inspection of (11) reveals, however, that the decoding algorithm based on this approximation would achieve performance equal to that of the Max-Log-MAP algorithm. This conclusion results from the following relation

$$\max(x_1, x_2, \dots, x_n) \geq \frac{1}{n} \sum_{i=1}^n x_i. \quad (12)$$

The idea for overcoming this drawback, proposed in [16], is to replace the part of the approximation (11) that computes the average value of input arguments x_i by the part that will compute the sum of all x_i and then divide it by a certain parameter N , suitably selected so as to optimize the performance of the algorithm. Thus, the new approximation that is expected to have a better performance than the Max-Log-MAP algorithm can be formulated as follows [16]:

$$\max^*(x_1, x_2, \dots, x_n) \approx \max\left(\max_{i=1:n}(x_i), \frac{1}{N} \sum_{i=1}^n x_i\right), \quad (13)$$

where N is the parameter of the approximation. For a given transmission scheme, an optimal value of N minimizing BER performance at the assumed signal-to-noise ratio (SNR) level can be found by means of computer simulations. The Log-MAP algorithm with the approximation of the \max^* operator given in (13) is referred to as the AvN Log-MAP algorithm.

3 Performance evaluation results

In this section, we evaluate performance of the AvN Log-MAP decoding algorithm by means of computer simulations for parallel and serial concatenated TTCM schemes. In simulations, the AWGN channel and the uncorrelated, i.e., fully interleaved, Rayleigh fading channel with perfect channel state information (CSI) were considered. Furthermore, the block sizes of $K = 684$ and $K = 5,000$ symbols

with the S -random interleavers with the spreading factors $S = 7$ and $S = 13$, respectively, were assumed. For comparison purposes, the BER performance curves for the conventional Log-MAP (with a LUT storing eight values), Linear Log-MAP, Average Log-MAP, Shift Log-MAP, and Max-Log-MAP algorithms were also evaluated. At the receiver, eight decoding iterations for all algorithms were performed.

3.1 Parallel TTCM scheme

The block diagram of the encoder structure for a parallel concatenated TTCM scheme is shown in Figure 1. The AvN Log-MAP algorithm was examined in the TTCM schemes with two rate-3/4 8-state Ungerboeck's trellis-coded modulation (TCM) encoders and 16-QAM modulation [18].

The results of the search for an optimal value of the parameter N of the approximation (13) for parallel turbo TCM schemes with $K = 684$ and $K = 5,000$ are depicted in Figures 2 and 3, respectively. Figures 2 and 3 show the BER performance curves versus parameter N for the AvN Log-MAP algorithm in AWGN channel at several values of SNR. As it can be seen in Figures 2 and 3, for the values of SNR that correspond to the waterfall region of performance curves, the value of N has an influence on the error performance. It can be concluded from these figures that for both block sizes, the optimal value of the parameter N , which minimizes BER is 18.5. It should also be noted that the search for the optimal value of N for serial turbo TCM schemes investigated in section 3.2, gives a very similar result. Thus, in all simulations the AvN Log-MAP algorithm uses $N = 18.5$ in the approximation (13). One may comment at this point that the same optimal value of N has been obtained for TTCM systems, which differ in a structure, overall coding rate, and block size, yet having component codes with an equal number of states (i.e., eight states). This may suggest that the trellis structure of component codes is the key factor that has an influence on the value of the parameter N . Nevertheless, it remains an open problem for further studies.

In terms of hardware or digital signal processor (DSP) implementation of decoding algorithms, it is desirable to

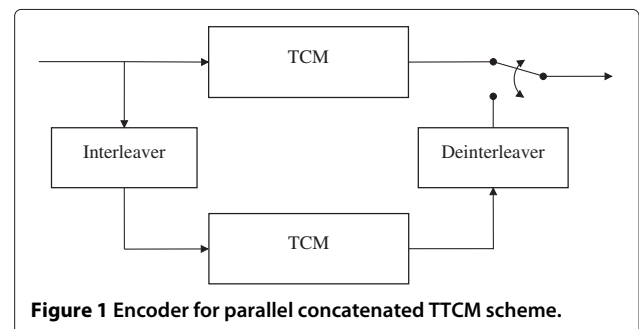


Figure 1 Encoder for parallel concatenated TTCM scheme.

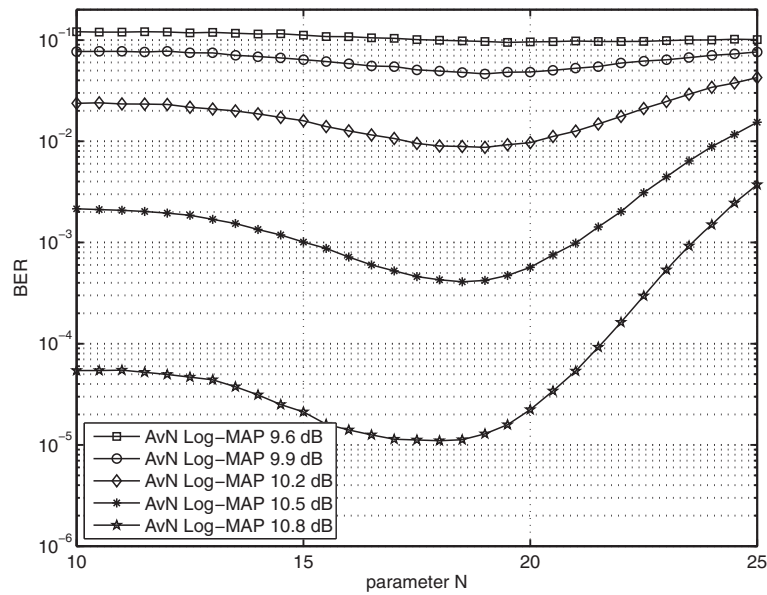


Figure 2 BER performance versus approximation parameter N (parallel TTCM, AvN Log-MAP, AWGN, $K = 684$).

avoid multiplications, which are rather complex operations. Therefore, the number of multiplications required by the algorithm can be regarded as one of the crucial factors in determining its complexity. As it will be shown in section 4, the AvN Log-MAP algorithm requires some multiplications. These multiplications are due to the division by parameter N in (13). It can be noticed, however, that if we select $N = 2^m, m = 1, 2, \dots$, then the multiplications can be replaced by bit shifts which simplifies the implementation. In order to obtain such a simplified algorithm for parallel TTCM, based on the results

from Figures 2 and 3, we may select $N = 16$ although this value of parameter N is not optimal but close to the optimal $N = 18.5$. Hence, one may expect only a small performance degradation at the gain of further reduction of the algorithm complexity when $N = 16$ is selected. In the remainder of this paper, the AvN Log-MAP algorithm with $N = 16$ will be denoted as the AvN Log-MAP hardware efficient (HE).

BER performance evaluation results in the AWGN channel for the small ($K = 684$ symbols) and large ($K = 5,000$ symbols) interleaver sizes are given in Figures 4 and

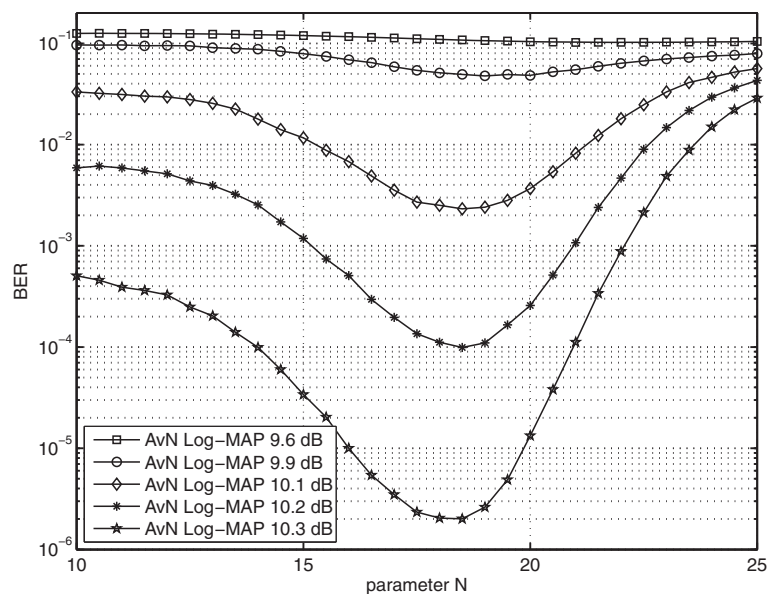


Figure 3 BER performance versus approximation parameter N (parallel TTCM, AvN Log-MAP, AWGN, $K = 5,000$).

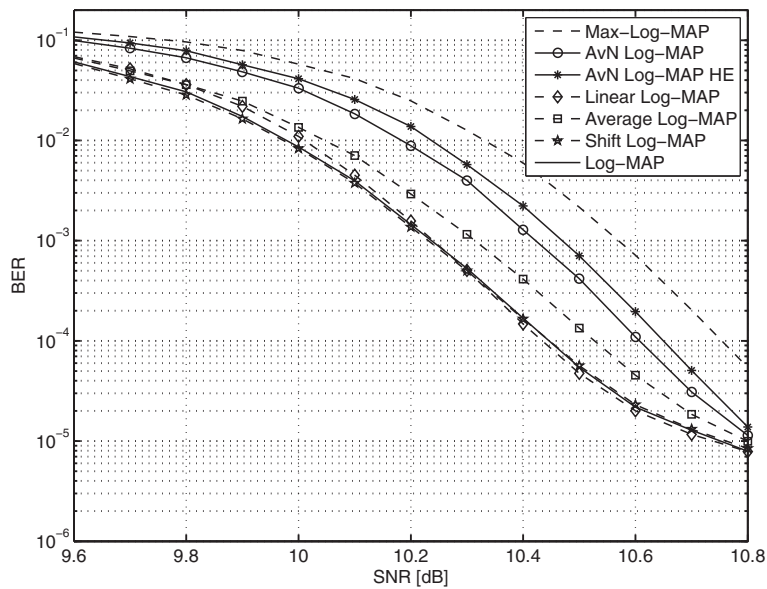


Figure 4 BER performance comparison of AvN Log-MAP and AvN Log-MAP HE with other decoding algorithms ($K = 684$, parallel TTCM, AWGN).

5, respectively. As it can be seen, the AvN Log-MAP algorithm outperforms the Max-Log-MAP by 0.15 dB and is less than 0.2 dB inferior to the Log-MAP at $BER = 10^{-4}$. Its loss to other algorithms is in the range of 0.1 to 0.2 dB. The AvNLog-MAP HE algorithm is merely less than 0.1 dB worse than the AvN Log-MAP at the same BER level. It can be noted that both novel algorithms outperform the Max-Log-MAP in the whole range of simulated SNRs. BER performance results of the AvN

Log-MAP algorithm for various number of iterations are presented in Figure 6. As expected, the error performance improvement with the increasing number of iterations is seen.

Simulation results in a Rayleigh fading channel are shown in Figures 7 and 8. Here, the AvN Log-MAP algorithm slightly outperforms the Max-Log-MAP but the performance loss to the Log-MAP remains in the same range of 0.1 to 0.2 dB at $BER = 10^{-4}$.

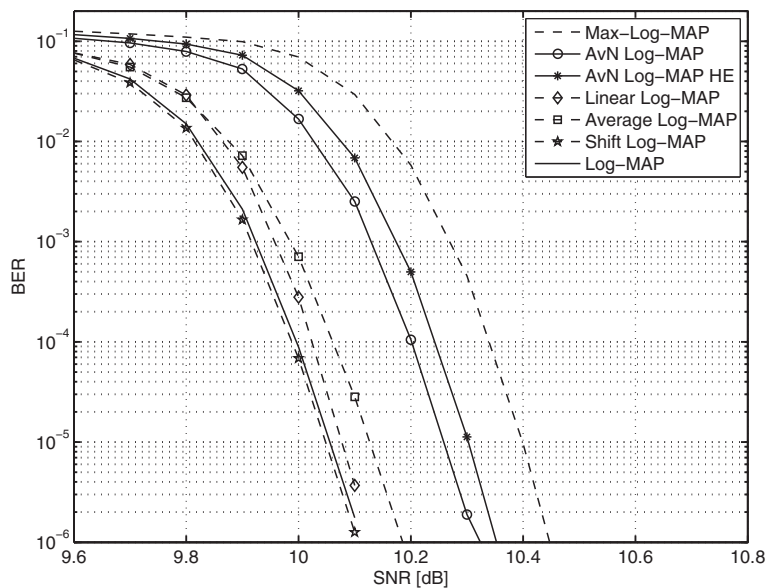


Figure 5 BER performance comparison of AvN Log-MAP and AvN Log-MAP HE with other decoding algorithms ($K = 5,000$, parallel TTCM, AWGN).

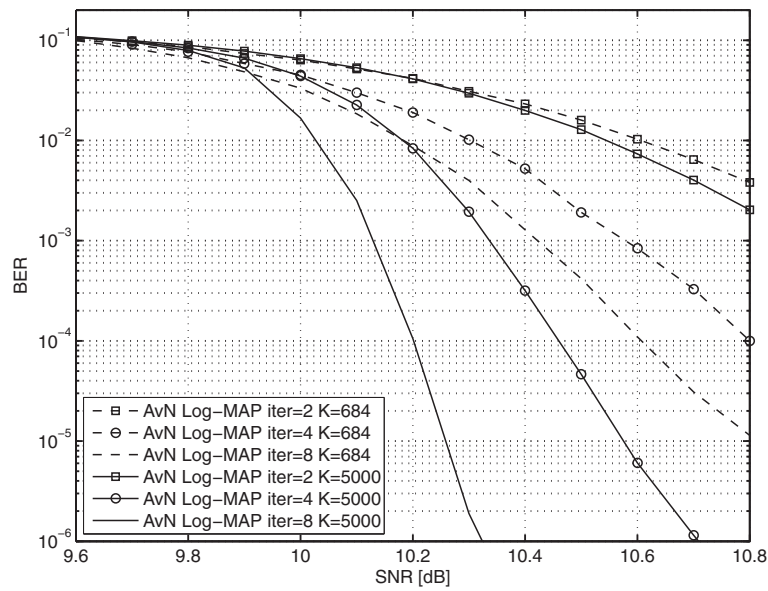


Figure 6 BER performance of AvN Log-MAP algorithm for various number of iterations, parallel TCM scheme, AWGN channel.

3.2 Serial TCM scheme

Encoder structure for a serial concatenated TCM scheme is depicted in Figure 9 [19]. A rate-2/3 (systematic feedback) convolutional encoder with parity polynomials (in octal notation) $h_0 = 13$, $h_1 = 15$, and $h_2 = 17$ was applied as an outer encoder. For the inner encoder, we used the same encoder as in the parallel TCM scheme. Hence, the overall code rate of the serial TCM scheme is $R_c = 1/2$.

Figures 10 and 11 illustrate BER performance results for serial TCM scheme in the AWGN channel and Figures 12 and 13 in the Rayleigh fading channel. For the AWGN channel, it is observed that the AvN Log-MAP is 0.3 dB inferior to the Log-MAP and 0.1 to 0.3 dB to other algorithms, except for the Max-Log-MAP, at BER of 10^{-4} for both block lengths. Comparing to the Max-Log-MAP algorithm, the performance of the AvN Log-MAP is improved by almost 0.5 dB. Similar to the results

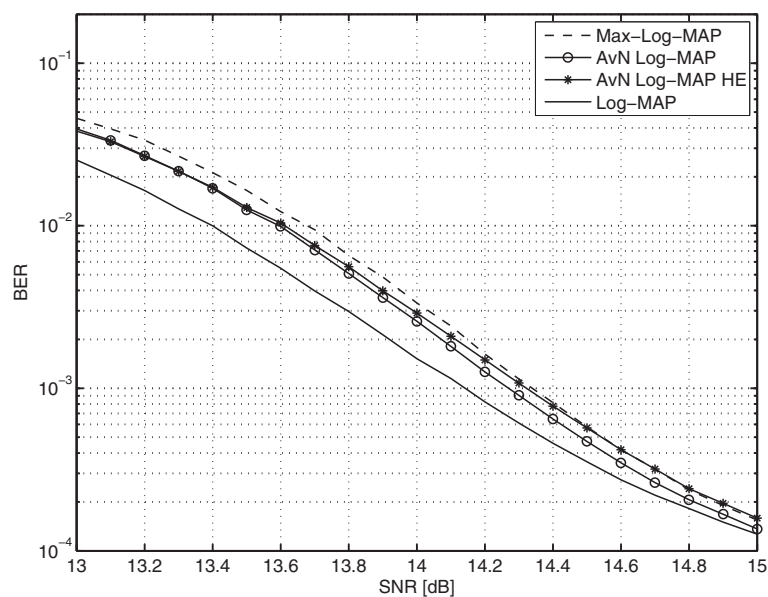


Figure 7 BER performance comparison of AvN Log-MAP, AvN Log-MAP HE, Log-MAP, and Max-Log-MAP algorithms ($K = 684$, parallel TCM, Rayleigh fading channel).

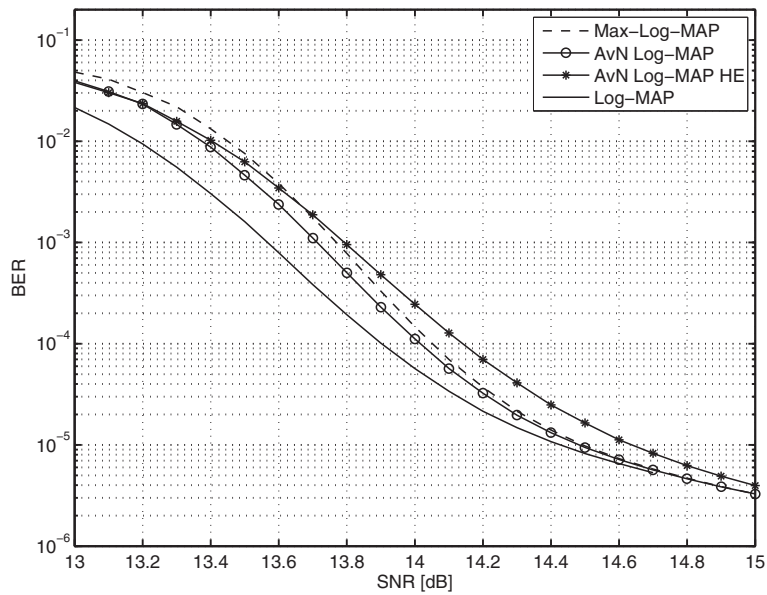


Figure 8 BER performance comparison of AvN Log-MAP, AvN Log-MAP HE, Log-MAP and Max-Log-MAP algorithms ($K = 5,000$, parallel TTCM, Rayleigh fading channel).

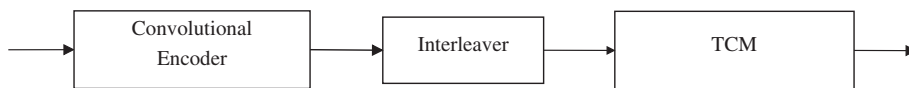


Figure 9 Encoder for serial concatenated TTCM scheme.

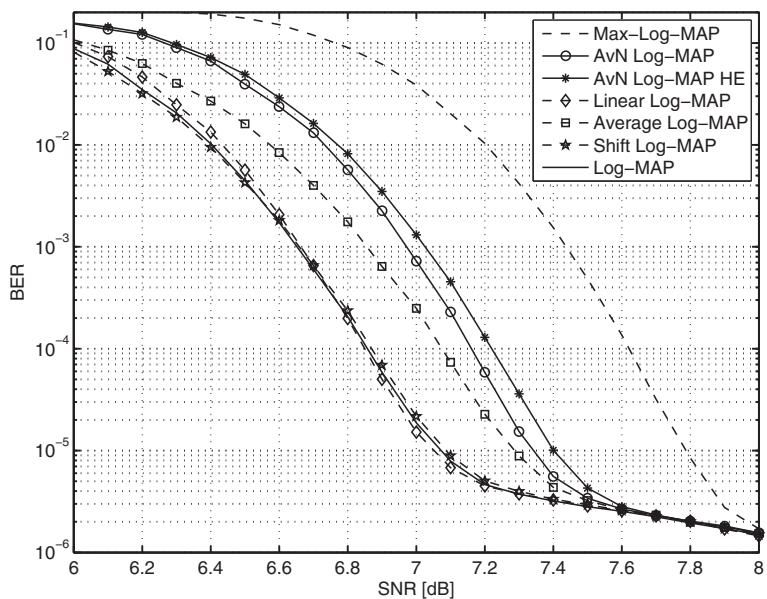


Figure 10 BER performance comparison of AvN Log-MAP and AvN Log-MAP HE with other decoding algorithms ($K = 684$, serial TTCM, AWGN).

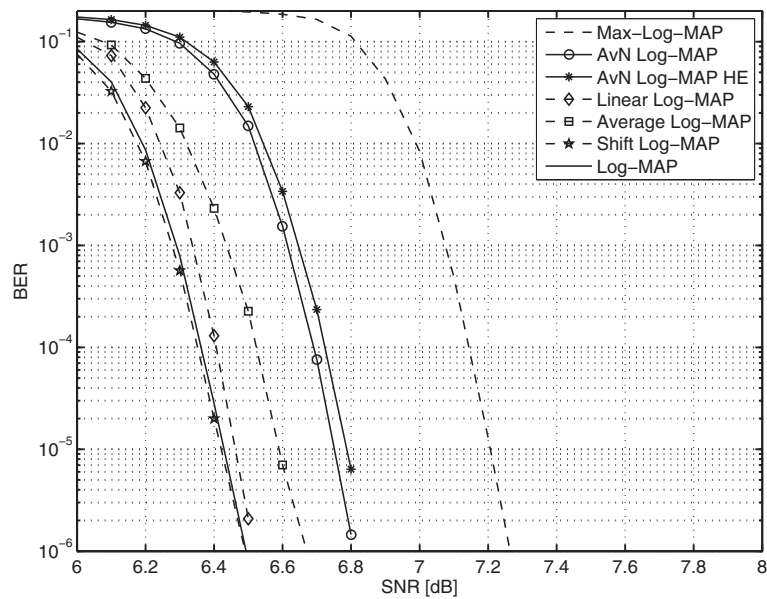


Figure 11 BER performance comparison of AvN Log-MAP and AvN Log-MAP HE with other decoding algorithms ($K = 5,000$, serial TTCM, AWGN).

for a parallel TTCM, the performance loss of the AvN Log-MAP HE algorithm against the AvN Log-MAP is less than 0.1 dB. In the case of transmission over a Rayleigh fading channel, the AvN Log-MAP outperforms the Max-Log-MAP by 0.3 dB and reveals a loss of 0.2 dB to the Log-MAP at BER 10^{-4} . Insignificant performance degradation of the AvN Log-MAP HE algorithm with respect to

the AvN Log-MAP, which uses the optimal value of N can be also noted.

4 Complexity comparison of the algorithms

From the implementation point of view, the key aspect of the AvN Log-MAP algorithm is its complexity compared against the Log-MAP and the reduced complexity

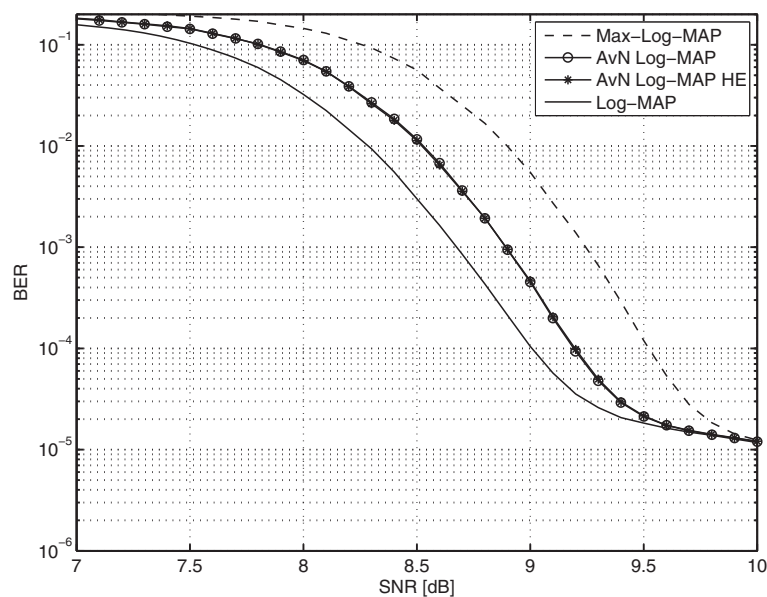


Figure 12 BER performance comparison of AvN Log-MAP, AvN Log-MAP HE, Log-MAP and Max-Log-MAP algorithms ($K = 684$, serial TTCM, Rayleigh fading channel).

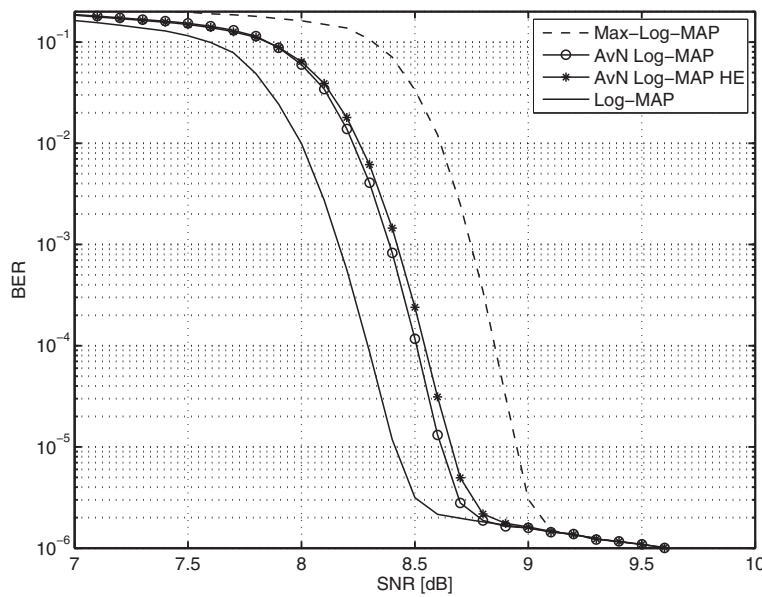


Figure 13 BER performance comparison of AvN Log-MAP, AvN Log-MAP HE, Log-MAP and Max-Log-MAP algorithms ($K = 5,000$, serial TTCM, Rayleigh fading channel).

algorithms previously proposed in the literature. Complexity comparison of the algorithms has been performed for both TTCM schemes from section 3 in software (i.e., computer based) and DSP implementations. DSP evaluation was accomplished on an Ultra Low Power TMS320VC5510 processor that uses a fixed-point number representation [20].

Tables 1 and 2 depict the required number of operations (i.e., additions, comparisons, bit shifts, conversion to integer and assignment) per decoding step in software implementation of the decoding algorithms for the parallel and serial concatenated TTCM scheme, respectively. As shown in Table 1, the proposed AvN Log-MAP algorithm in parallel TTCM scheme is 41.1% simpler than the Log-MAP algorithm. This significant reduction in the number of operations comes at an expense of small BER

performance degradation, as it was shown by simulation results. When compared with the Max-Log-MAP algorithm, it is found that the AvN Log-MAP algorithm requires 34.0% more operations.

According to Table 2, the reduction in complexity of the AvN Log-MAP algorithm against the Log-MAP is also significant in serial TTCM scheme and amounts to 31.2%. Comparison to the simple Max-Log-MAP algorithm reveals that the AvN Log-MAP requires 43.3% more operations, but according to the BER results presented in the previous section, it exchanges for a performance improvement of the proposed algorithm by 0.3 to 0.5 dB.

It can also be observed in Tables 1 and 2 that the AvN Log-MAP HE algorithm performs the same overall number of operations as the AvN Log-MAP but

Table 1 Complexity comparison of decoding algorithms per decoding step for parallel TTCM scheme in software implementation

Algorithm	Log-MAP	AvN Log-MAP	AvN Log-MAP HE	Linear Log-MAP	Average Log-MAP	Shift Log-MAP	Max-Log-MAP
Additions	680	512	512	848	680	764	344
Multiplications	0	24	0	0	0	0	0
Comparisons	357	213	213	357	357	441	189
Bit shifts	168	0	24	168	84	126	0
Conversion to integer	168	0	0	0	0	0	0
Assignment	233	197	197	0	233	233	173
Overall	1,606	946	946	1,606	1,354	1,564	706

Table 2 Complexity comparison of decoding algorithms per decoding step for serial TCM scheme in software implementation

Algorithm	Log-MAP	AvN Log-MAP	AvN Log-MAP HE	Linear Log-MAP	Average Log-MAP	Shift Log-MAP	Max-Log-MAP
Additions	492	432	432	606	492	549	264
Multiplications	0	24	0	0	0	0	0
Comparisons	249	165	165	249	249	306	141
Bit shifts	114	0	24	114	57	86	0
Conversion to integer	114	0	0	0	0	0	0
Assignment	185	173	173	185	185	185	149
Overall	1,154	794	794	1,154	983	1,126	554

instead of multiplications it realizes bit shifts. This feature of the AvN Log-MAP HE is favorable in hardware implementation of the algorithm and can be considered as an additional complexity reduction.

Tables 3 and 4 summarize the comparison of the AvN Log-MAP against all competing algorithms in terms of overall number of operations in the parallel and serial scenarios, respectively. It can be easily seen that the AvN Log-MAP algorithm offers much higher reduction in the number of operations with regards to the Log-MAP than the Linear Log-MAP, Average Log-MAP, and Shift Log-MAP algorithms. Inspection of Tables 3 and 4 shows that this reduction is more than twice as much as that offered by the least complex algorithm among the competitors – the Average Log-MAP: 41.1% vs. 15.7% in a parallel scheme and 31.2% vs. 14.8% in a serial scheme. Taking into account only a small performance degradation of the AvN Log-MAP algorithm, this substantial reduction in computational effort makes the proposed algorithm attractive for the practical implementation.

DSP implementation-based complexity comparison of the algorithms is presented in Tables 5 and 6. The numbers of processor’s cycles needed per single decoding step for the considered algorithms in the parallel and serial concatenated TCM schemes are shown. As it can be seen

in Table 5, the AvN Log-MAP algorithm in parallel TCM scheme is 48.9% simpler than the Log-MAP algorithm and requires only 8.0% more processor’s cycles than the simple Max-Log-MAP. In serial concatenation TCM scheme (Table 6), these comparisons give the numbers of 41.6% and 27.9%, respectively. Comparison of the AvN Log-MAP algorithm to the remaining algorithms (Table 5) shows that its reduction in the number of processor’s cycles against Log-MAP is more than 30% higher than that achieved by the Linear Log-MAP. From the analysis given in this section, we conclude that the AvN Log-MAP algorithm offers significant savings in decoding effort with respect to the Log-MAP and relatively small increase of complexity as compared to the Max-Log-MAP algorithm. It is also substantially less complex than all reduced complexity algorithms given as a reference in this paper. It should be emphasized that all complexity comparisons presented in Tables 1, 2, 3, 4, 5, and 6 are per single decoding step in a constituent decoder and hence, these results do not depend on the number of iterations or the block size.

5 Conclusions

A low complexity AvN Log-MAP algorithm has been presented and investigated for turbo TCM schemes. It is

Table 3 Complexity comparison of decoding algorithms per decoding step

Algorithm	Overall number of operations	Reduction wrt Log-MAP
Log-MAP	1,606	—
AvN Log-MAP	946	41.1%
Linear Log-MAP	1,606	0%
Average Log-MAP	1,354	15.7%
Shift Log-MAP	1,564	2.6%
Max-Log-MAP	706	56%

For parallel TCM scheme in software implementation – overall number of operations.

Table 4 Complexity comparison of decoding algorithms per decoding step

Algorithm	Overall number of operations	Reduction wrt Log-MAP
Log-MAP	1,154	—
AvN Log-MAP	794	31.2%
Linear Log-MAP	1,154	0%
Average Log-MAP	983	14.8%
Shift Log-MAP	1,126	2.5%
Max-Log-MAP	554	52%

For serial TCM scheme in software implementation – overall number of operations.

Table 5 Complexity comparison of decoding algorithms per decoding step for parallel TCM scheme in DSP implementation

Algorithm	Cycles	Reduction wrt Log-MAP
Log-MAP	12,236	—
AvN Log-MAP	6,253	48.9%
Linear Log-MAP	10,088	17.6%
Average Log-MAP	10,581	13.5%
Shift Log-MAP	11,816	3.4%
Max-Log-MAP	5,790	52.7%

based on the approximation of the \max^* operator with $n \geq 2$ values, derived from the Jensen inequality. As is apparent from the simulation results obtained for AWGN and uncorrelated Rayleigh fading channels, the performance of the proposed algorithm is slightly inferior to that of the Log-MAP but its computational requirements are substantially lower. In particular, we have found that the AvN Log-MAP algorithm performs about 0.1 to 0.4 dB worse than the Log-MAP at BER of 10^{-4} , depending on TCM scheme and radio channel, while it is also as high as 31.2% to 48.9% much simpler. Performance and complexity comparisons with some relevant reduced complexity decoding algorithms proposed in the literature, i.e., Linear Log-MAP, Average Log-MAP, and Shift Log-MAP, have shown that the AvN Log-MAP algorithm reveals significant decrease in decoding complexity and its computational requirements are much lower than those of these algorithms. The penalty paid for this favorable low complexity of the AvN Log-MAP algorithm is a small performance degradation with respect to those algorithms. Moreover, proper selection of parameter N in the approximation will offer further simplification of the algorithm by means of replacing multiplication operations with simple bit shifts. Finally, it should be emphasized that the AvN Log-MAP algorithm is also suitable for implementation in SISO modules for iterative scenarios other than turbo TCM.

Table 6 Complexity comparison of AvN Log-MAP, Log-MAP, and Max-Log-MAP decoding algorithms per decoding step

Algorithm	Cycles	Reduction wrt Log-MAP
Log-MAP	9,902	—
AvN Log-MAP	5,780	41.6%
Max-Log-MAP	4,519	54.4%

For serial TCM scheme in DSP implementation.

Competing interests

The authors declare that they have no competing interests.

Received: 22 February 2013 Accepted: 16 September 2013

Published: 26 September 2013

References

1. C Berrou, A Glavieux, P Thitimajshima, Near Shannon limit error-correcting coding and decoding, in *Proc. IEEE International Conference on Communications (ICC)* (Geneva, Switzerland, 1993), pp. 1064–1070
2. C Berrou, A Glavieux, Near optimum error-correcting coding and decoding: Turbo-codes. *IEEE Trans. Commun.* **44**(10), 1261–1271 (1996)
3. LR Bahl, J Cocke, F Jelinek, J Raviv, Optimal decoding of linear codes for minimizing symbol error rate. *IEEE Trans. Inform. Theory* **IT-20**(2), 284–287 (1974)
4. P Robertson, E Villebrun, P Hoeher, A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain, in *Proc. IEEE International Conference on Communications (ICC)* (Seattle, USA, 1995), pp. 1009–1013
5. P Robertson, P Hoeher, E Villebrun, Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding. *ETT* **8**(2), 119–125 (1997)
6. AJ Viterbi, An intuitive justification and a simplified implementation of the MAP decoder for convolutional codes. *IEEE J. Sel. Areas Commun.* **16**(2), 260–264 (1998)
7. WJ Gross, PG Gulak, Simplified MAP algorithm suitable for implementation of turbo decoders. *Electron. Lett.* **34**(16), 1577–1578 (1998)
8. JF Cheng, T Ottosson, Linearly approximated Log-MAP algorithms for turbo decoding, in *Proc. 51st IEEE Vehicular Technology Conference (VTC 2000-Spring)* (Tokyo, Japan, 2000), pp. 2252–2256
9. B Classon, K Blankenship, V Desai, Channel coding for 4G systems with adaptive modulation and coding. *IEEE Wireless Commun. Mag.* **9**(2), 8–13 (2002)
10. L Xi-Zhong, M Zhi-Gang, Y Yi-Zheng, C Yan-Min, A simplification of the Log-MAP algorithm for turbo decoding, in *Proc. of IEEE Asia-Pacific Conference on Circuits and Systems, Volume 2* (Tainan, Taiwan, 2004), pp. 1057–1060
11. H Wang, H Yang, D Yang, Improved Log-MAP decoding algorithm for turbo-like codes. *IEEE Commun. Lett.* **10**(3), 186–188 (2006)
12. S Talakoub, L Sabeti, B Shahrrava, M Ahmadi, An improved Max-Log-MAP algorithm for turbo decoding and turbo equalization. *IEEE Trans. Instrum. Meas.* **56**(3), 1058–1063 (2007)
13. S Papaharalabos, PT Mathiopoulos, G Masera, M Martina, On optimal and near-optimal turbo decoding using generalized \max^* operator. *IEEE Commun. Lett.* **13**(7), 522–524 (2009)
14. M Sybis, Log-MAP equivalent Chebyshev inequality based algorithm for turbo TCM decoding. *Electron. Lett.* **47**(18), 1049–1050 (2011)
15. S Papaharalabos, M Sybis, P Tyczka, PT Mathiopoulos, Novel generalized \max^* approximation method for simplified decoding of turbo and TCM codes. *Wireless Personal Commun.* **69**, 373–386 (2013)
16. M Sybis, P Tyczka, S Papaharalabos, PT Mathiopoulos, Simplified Log-MAP decoding with \max^* approximation based on the Jensen inequality, in *Proc. 21st IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2010)* (Istanbul, Turkey, 2010), pp. 283–287
17. W Rudin, *Real and Complex Analysis* (McGraw-Hill, New York, 1986)
18. P Robertson, T Wörz, Bandwidth-efficient turbo trellis-coded modulation using punctured component codes. *IEEE J. Select. Areas Commun.* **16**(2), 206–218 (1998)
19. CB Schlegel, LC Perez, *Trellis and Turbo Coding* (Wiley, New York, 2004)
20. Texas Instruments, *TMS320VC5510/5510A Fixed-point Digital Signal Processors. Data Manual* (Texas Instruments, Dallas, 2007)

doi:10.1186/1687-1499-2013-238

Cite this article as: Sybis and Tyczka: Reduced complexity Log-MAP algorithm with Jensen inequality based non-recursive \max^* operator for turbo TCM decoding. *EURASIP Journal on Wireless Communications and Networking* 2013 **2013**:238.