Syndrome Based Adaptive Complexity Channel Decoding and Turbo Equalization for ATSC DTV

Klaus Hueske, Jan Geldmacher and Jürgen Götze
Information Processing Lab,
Department of Electrical Engineering and Information Technology,
TU Dortmund University, Otto-Hahn-Strasse 4, 44227 Dortmund, Germany.
Email: klaus.hueske@tu-dortmund.de

***Abstract*—To realize energy efficient broadcasting receivers the complexity of the applied algorithms should be adaptive, i.e. the number of required operations should decrease with improving reception conditions. In case of channel decoding algorithms this adaptive behavior can be achieved by syndrome based decoding methods, which allow a reduction of the receiver's energy consumption in above-average reception conditions. This paper extends the syndrome decoding approach to trellis coded modulation, as applied in the ATSC DTV system. Furthermore, the application of the syndrome concept to receivers based on Turbo equalization is considered. In this case the syndrome approach enables a reduction of the soft output decoder's complexity.**

I. Introduction

Due to increased energy costs and growing awareness of users, energy efficiency has become a major topic for consumer electronics and communication infrastructure (often referred to as "Green IT"). The energy consumption of a broadcast receiver is significantly influenced by the complexity of the used algorithms.

To reduce the average computational complexity (and with it the energy consumption of a receiver), the algorithms should be able to adapt to the current reception conditions, i.e. utilize all available processing power in worst case conditions, but adaptively reduce the complexity for more favorable reception conditions. This especially applies to broadcasting systems like ATSC DTV [1], where good reception conditions experienced by a subset of users cannot be exploited by reducing transmitter power or modifying transmission parameters.

Several approaches are known to realize such an adaptive complexity behavior for decoding trellis based codes, as e.g. the T-Algorithm [2]. However, the T-Algorithm is more beneficial for codes with a large number of trellis states, which is not the case in the ATSC system, where trellis coded modulation (TCM) with a four state encoder is used [3]. An alternative approach is the application of block syndrome decoding, as described in [4]. This approach is based on identifying error-free parts of the received sequence in advance, such that the decoder has to process only the erroneous parts. The basic principles of syndrome decoding and its application to ATSC DTV are described in this work.

Turbo equalization is a powerful method to iteratively improve the quality of data estimation by exchanging information between equalizer and decoder [5]. The significantly improved estimation quality comes at the cost of an increased computational complexity due to iterative processing. The application of Turbo equalization to ATSC DTV was proposed in [6], which also showed that the equalizer complexity can be reduced by using FFT based frequency domain equalization. To reduce the computational effort of the decoder, soft output block syndrome decoding based on the BCJR algorithm can be used [7]. The syndrome based BCJR algorithm has to process only erroneous parts. This reduces the computational effort in case of convergence, as the number of remaining errors will decrease during the iteration process.

It can be summarized that the reduction of decoding complexity depends on the number of errors in the sequence to be decoded. In conventional systems a higher SNR leads to lower decoding complexity. When applying Turbo equalization the decoding complexity also decreases with increasing iteration number.

The paper is organized as follows: Section II describes the ATSC system model including transmitter, non-iterative receiver and iterative receiver. The block syndrome decoder for maximum likelihood decoding (ML) and soft output decoding (BCJR) is introduced in Section III. Simulation results are presented in Section IV. Conclusions are drawn in Section V.

II. ATSC System Model

A. Transmitter and Channel

In the ATSC DTV system [1] data is transmitted using vestigial sideband modulation (VSB) with $M = 8$ levels in combination with trellis coded modulation (TCM) [3]. Furthermore, the inner TCM encoder is serially concatenated with an outer Reed-Solomon (RS) encoder as depicted in Fig. 1. The transport stream $\mathbf{d}$ is encoded by the outer RS encoder and fed into a data interleaver. The interleaved sequence $\mathbf{u}$ is encoded

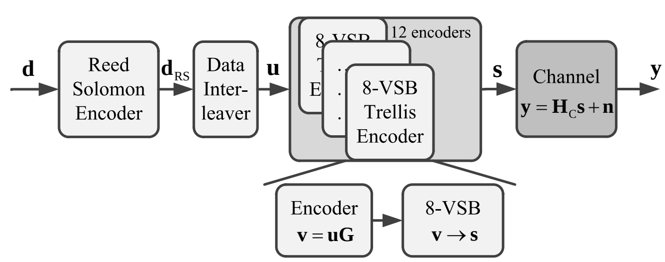

Fig. 1. Simplified ATSC DTV transmitter structure.

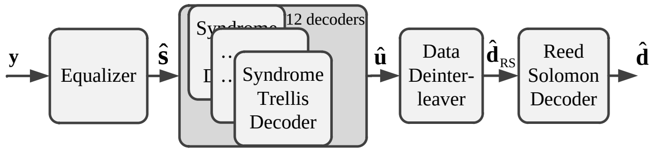

Fig. 2. Non-iterative ATSC DTV receiver structure.

by a rate $R = 2/3$ recursive convolutional code, which is defined by its generator matrix $\mathbf{G}$. The encoded data is mapped to constellation symbols $s_k \in S$ with $S = \{\pm 1 \pm 3 \pm 5 \pm 7\}$ following the concept of TCM. Trellis code interleaving is realized by using 12 identical TCM encoders, whose output is multiplexed to a sequence $\mathbf{s}$ of length P in a round-robin manner [1].

After transmission over a frequency selective channel the received sequence can be described as $\mathbf{y} = \mathbf{H}_c\mathbf{s} + \mathbf{n}$, where $\mathbf{H}_c$ is the $(P + L - 1) \times P$ channel convolution matrix corresponding to the length L channel, and $\mathbf{n}$ represents additive white Gaussian noise (AWGN) with noise power σ_n^2.

B. Non-Iterative Receiver

The structure of a conventional non-iterative receiver complements the transmitter structure as shown in Fig. 2.

The channel output $\mathbf{y}$ is fed to an equalizer, which generates symbol estimates $\hat{\mathbf{s}}$. The equalizer can be either linear (e.g. MMSE) or non-linear (e.g. DFE). The estimates $\hat{\mathbf{s}}$ are processed by 12 TCM decoders, corresponding to the transmitter's interleaving scheme. The TCM decoder is commonly based on the Viterbi algorithm, which determines the path in the encoder trellis with maximum likelihood to the received sequence. In this work the conventional decoder is replaced by the adaptive complexity syndrome decoder, which is described in Section III. The TCM decoder output $\hat{\mathbf{u}}$ is deinterleaved to $\hat{\mathbf{d}}_{RS}$ and fed to the outer RS decoder, which generates estimates of the transport stream $\hat{\mathbf{d}}$.

C. Iterative Turbo Receiver

Turbo equalization is a powerful method to iteratively improve data estimation quality. This is realized by exchanging probabilities $P(s_k = s_i)$ $(i = 1, \dots, M)$ of each symbol s_k between a soft-input/soft-output (SISO) equalizer and a SISO channel decoder. The basic principle of Turbo equalization was first described in [5], its application to ATSC DTV using a frequency domain equalizer was proposed in [6]. The modified receiver structure with SISO equalizer, SISO decoder and feedback connection is shown in Fig. 3.

The equalizer generates probabilities $P_{eq}(s_k = s_i)$ for all $i = 1, \dots, M$ possible symbols using the distorted sequence $\mathbf{y}$ and the a-priori symbol probabilities $P_{dec}(s_k = s_i)$ generated

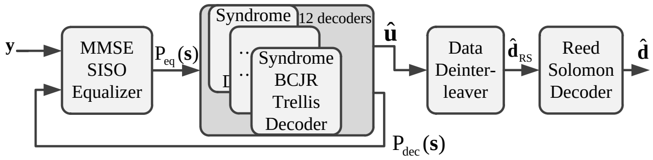

Fig. 3. Iterative ATSC DTV receiver structure.

by the decoder. In the first iteration no a-priori probabilities are available and the probability for each symbol s_i is set to a constant value $P_{dec}(s_k = s_i) = 1/M$. In the final iteration, the decoder will output an estimate $\hat{\mathbf{u}}$.

SISO versions of equalizer and decoder can be realized based on the trellis representation of the channel and the code, respectively. In this work the SISO decoder is realized using a reduced complexity probability based BCJR algorithm, which is described in Section III. A trellis based equalizer implementation for ATSC DTV receivers is unfeasible due to long channel impulse responses. Hence, a linear SISO equalizer is applied as described in [8] and [9].

The linear SISO equalizer computes estimates $\hat{s}_k$ of the transmitted symbols s_k by applying a time varying filter $\mathbf{c}_k$ to the difference between the received symbols $\mathbf{y}_k$ and a prediction $\mathbf{H}_b\bar{\mathbf{s}}_k$ of the received symbols,

$$\hat{s}_k = \mathbf{c}_k^H(\mathbf{y}_k - \mathbf{H}_b\bar{\mathbf{s}}_k), \quad \text{where} \tag{1}$$

- $\mathbf{c}_k$ is the $(N \times 1)$ filter vector, with $N = N_1 + N_2 + 1$
- $\mathbf{y}_k$ is a vector consisting of the received symbols y_{k-N_2} to y_{k+N_1}
- $\mathbf{H}_b$ is an $N \times (N + L - 1)$ section of $\mathbf{H}_c$
- $\bar{\mathbf{s}}_k$ is an $(N + L - 1) \times 1$ vector consisting of the expectations $\bar{s}_{k-L-N_2+1}$ to $\bar{s}_{k+N_1}$

The filter vector is computed as a weighted MMSE estimator,

$$\mathbf{c}_k = (\sigma_n^2\mathbf{I}_N + \mathbf{H}_b\mathbf{V}_k\mathbf{H}_b^H)^{-1}\mathbf{h}, \tag{2}$$

where $\mathbf{h}$ is the (N_2+L)-th column of $\mathbf{H}_b$ and $\mathbf{V}_k$ is a diagonal matrix,

$$\mathbf{V}_k = \text{diag}(\text{var}(s_{k-L-N_2+1}), \dots, \text{var}(s_{k+N_1})). \tag{3}$$

where $\text{var}(\cdot)$ denotes variance. Expectation and variance can be computed as

$$\bar{s}_k = E\{s_k\} = \sum_{s_i \in S} s_i P_{dec}(s_k = s_i) \quad \text{and} \tag{4}$$

$$\text{var}(s_k) = \sum_{s_i \in S} |s_i|^2 P_{dec}(s_k = s_i) - |\bar{s}_k|^2 \tag{5}$$

To guarantee that the filter generates an estimate $\hat{s}_k$, which is independent of $P_{dec}(s_k = s_i)$, the $(N_2 + L)$-th element in $\bar{\mathbf{s}}_k$ is set to zero and the $(N_2 + L)$-th diagonal element of $\mathbf{V}_k$ is set to 1. Considering that equalizer and decoder are separated by an interleaver, the M extrinsic probabilities $P_{eq}(s_k = s_i)$ can be computed for each k as

$$P_{eq}(s_k = s_i) \sim \exp\left(-\frac{|\hat{s}_k - \mu_k s_i|^2}{\lambda\sigma_k^2}\right) \tag{6}$$

where μ_k and σ_k^2 are the equivalent amplitude and noise variance of the equalizer output, as described in [9], and λ denotes a scaling factor. The described equalizer requires the computation of new filter coefficients for each symbol s_k. The complexity can be significantly reduced by approximating the matrix $\mathbf{V}_k$ by a diagonal matrix with constant elements as described in [8]. With this approximation filter coefficients change only once per iteration. A further reduction in complexity can be achieved by FFT based frequency domain equalization [6].

III. Adaptive Complexity TCM Decoder

The proposed adaptive complexity decoding approach is based on the syndrome decoding principle as introduced by Schalkwijk et al. [10]. For this decoding algorithm, error sequences/symbols are estimated, instead of information sequences/symbols. This property, combined with syndrome based pre-processing can be used to implement a simple but powerful adaptive complexity decoding algorithm as described in [4] and [7]. The following subsections introduce syndrome based decoding of TCM for maximum likelihood, as used for non-iterative receivers, and SISO decoding, as used for iterative receivers.

A. Syndrome Trellis for TCM

While conventional decoders operate on the encoder trellis defined by $\mathbf{G}(D)$, the syndrome decoder operates on the trellis of the syndrome former $\mathbf{H}^T(D)$, which is defined to be orthogonal to the code, i.e. $\mathbf{G}(D)\mathbf{H}^T(D) = \mathbf{0}$.

Let $\mathbf{r} = \mathbf{uG} + \mathbf{e}_c$ be a hard decision of the time-domain distorted encoded sequence, then

$$\mathbf{b} = \mathbf{rH}^T = \mathbf{uGH}^T + \mathbf{e}_c\mathbf{H}^T = \mathbf{e}_c\mathbf{H}^T \tag{7}$$

is called the syndrome sequence. Note that $\mathbf{b}$ only depends on the channel error $\mathbf{e}_c$. Thus, if we construct a trellis for $\mathbf{H}^T(D)$, where only the transitions according to $\mathbf{b}$ are allowed, then each trellis path is an admissible error sequence of $\mathbf{r}$. Therefore, this trellis can be used to generate a maximum likelihood estimate of the error sequence or symbol-by-symbol maximum-a-posteriori estimates of the error symbols. The former is realized by the Viterbi algorithm and results in a hard-decision estimate (cf. III-B), while the latter can be used to generate soft-outputs using the BCJR algorithm [11] (cf. III-C).

The TCM encoder, which is used in ATSC DTV, is shown in Fig. 4. The precoder is not considered. The encoder can be represented by the generator matrix

$$\mathbf{G}(D) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \frac{D}{1+D^2} \end{bmatrix},$$

and a corresponding syndrome former is readily found to be $\mathbf{H}^T(D) = \left[0,\ D,\ D^2+1\right]^T$.

B. Maximum Likelihood Decoding

The equalizer generates estimates $\hat{s}_k$ of the transmitted symbols. In order to generate the $\mathbf{H}^T$-trellis, the syndrome sequence $\mathbf{b}$ has to be computed. This is realized by taking a hard demapping of the symbols $\hat{s}_k$, $\mathbf{r}_k = \text{DeMap}\,(\hat{s}_k)$, where

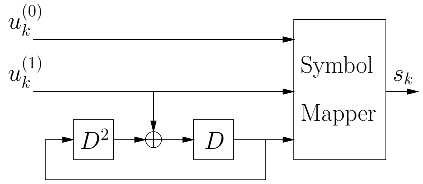

Fig. 4. TCM encoder.

$\mathbf{r}_k = \{r_k^{(j)}\}$, $j = 0, \ldots, \text{ld}(M) - 1$, and shifting the binary sequence $\mathbf{r} = \{\mathbf{r}_k\}$ into the realization of $\mathbf{H}^T(D)$.

The ML error sequence $\hat{\mathbf{e}}$ is found by applying the Viterbi algorithm to the trellis of $\mathbf{H}^T(D)$ subject to $\mathbf{b}$ and searching for the error sequence that corresponds to the symbol sequence with minimum euclidean distance to $\hat{\mathbf{s}}$. This can be realized as follows: Let the branch metric of a transition from state p to q at time instant k be denoted by $\mu_k(p,q)$ and let $\mathbf{e}(p,q)$ be the error symbol associated with this transition. Then $\mu_k(p,q)$ is given by

$$\mu_k(p,q) = \|\text{Map}\,(\mathbf{e}(p,q) \oplus \mathbf{r}_k) - \hat{s}_k\|_2^2\,, \tag{8}$$

where $\text{Map}(\cdot)$ is the mapping of $\mathbf{e}(p,q) \oplus \mathbf{r}_k$ to the according constellation symbol. The ML error sequence $\hat{\mathbf{e}}$ is then given by the path with minimum accumulated metric. Finally, the systematic part of $\hat{\mathbf{e}}$ is applied to the systematic part of $\mathbf{r}$ to get the ML estimate of $\hat{\mathbf{u}}$.

Note that trellis complexity and required trellis operations are identical to the conventional TCM decoder. Marginal additional effort is required for hard-demapping, syndrome computation and mapping of $\mathbf{e}(p,q) \oplus \mathbf{r}_k$.

C. SISO decoding

For Turbo equalization, the probability based BCJR algorithm is applied as follows: The SISO decoder receives the equalizer output $P_{eq}(s_k = s_i)$ for all $i = 1, \ldots, M$, which consists of extrinsic probabilities for each time instant k, as input. In a first step, the decoder generates an estimate of the coded binary symbols $\mathbf{r}_k$ as

$$\mathbf{r}_k = \text{DeMap}(\arg\max_{s_i} P_{eq}(s_k = s_i)), \tag{9}$$

which means that the binary representation of the most probable symbol is assigned to $\mathbf{r}_k$ at each time instant. Using $\mathbf{r} = \{\mathbf{r}_k\}$, the syndrome $\mathbf{b} = \mathbf{rH}^T$ can be computed, which defines the trellis. Note that $\mathbf{r}$ changes from iteration to iteration and that the number of erroneous bits in $\mathbf{r}$ decreases in case of convergence.

Based on the syndrome former trellis, subject to $\mathbf{b}$, the probability version of the BCJR decoder can be implemented. It consists of forward and backward recursions

$$\alpha_k(q) \sim \sum_p \alpha_{k-1}(p)\gamma_k(p,q), \quad \beta_k(p) \sim \sum_q \gamma_k(p,q)\beta_{k+1}(q), \tag{10}$$

where $\alpha_k(q)$ and $\beta_k(p)$ are state probabilities for forward and backward recursions, respectively. The transition probability $\gamma_k(p,q)$ from state p to state q and time instant k is determined by $P_{eq}\,(s_k = \text{Map}(\mathbf{r}_k \oplus \mathbf{e}(p,q)))$, where again $\mathbf{r}_k \oplus \mathbf{e}(p,q)$ is the coded symbol for the transition from state p to state q at time instant k and $\text{Map}(\mathbf{r}_k \oplus \mathbf{e}(p,q))$ denotes the corresponding constellation symbol.

During the backward recursion, a-posteriori probabilities are computed following

$$P_{dec}(s_k = s_i) \sim \sum_{(p,q)\in\mathcal{T}_i} \alpha_k(p)\gamma_k(p,q)\beta_{k+1}(q), \tag{11}$$

where $\mathcal{T}_i$ denotes transitions (p, q) that correspond to an output $s_i = \text{Map}(\mathbf{r}_k \oplus \mathbf{e}(p, q))$. The estimate of the transmitted symbol in the final iteration is the symbol with highest probability.

Finally, extrinsic probabilities can be generated by

$$P^e_{dec}(s_k = s_i) \sim P_{dec}(s_k = s_i)/P_{eq}(s_k = s_i). \qquad (12)$$

Note that proper normalization is required in (10), (11), (12). A more detailed discussion of syndrome based SISO decoding can be found in [7].

D. Pre-processing

The objective of the pre-processing is to separate the decoder's input sequence into erroneous and error-free parts, where only the erroneous blocks are processed by the decoder. Depending on the used decoding method, error-free blocks are treated as follows: For ML decoding corresponding blocks in $\hat{\mathbf{e}}$ are set to zero, i.e. the received sequence is not corrected in these blocks. For BCJR decoding corresponding probabilities for the most likely symbols are set to one.

Error-free parts in $\mathbf{r}$ can be determined by identifying parts of consecutive zeros in $\mathbf{b}$ with sufficient length. This exploits that $\mathbf{b}$ is a function solely of the channel error, as shown in (7), and that error-free parts in $\mathbf{r}$ propagate to zero sequences in $\mathbf{b}$. The minimum length of consecutive zeros, that is required to consider a block as error-free is a design parameter, which is called ℓ_{min} in the following.

The number of 1's in the syndrome sequence depends on the number of errors in the distorted sequence $\mathbf{r}$. Thus, more and longer parts of consecutive zeros appear in $\mathbf{b}$ and more error-free blocks are identified

- in case of the ML decoder with increasing SNR and
- in case of Turbo equalization with increasing SNR and with increasing iteration number.

More generally, the result of this procedure is a decoding complexity that adapts to the number of remaining errors in the decoder's input.

The parameter ℓ_{min} influences the decoding performance in terms of Bit-Error-Rate (BER) and the reduction of effort (amount of error-free parts). It may be selected heuristically as a trade-off between required BER and reduction of decoding effort. We refer to [4] and [7] for a more detailed discussion.

IV. SIMULATIONS

BER performance and achievable savings in decoding operations are determined for non-iterative and iterative ATSC DTV receivers.

A. Non-Iterative Receiver

Simulation results for the non-iterative system were obtained using transmitter and receiver as shown in Fig. 1 and 2. As the focus is on decoder performance, no fading effects were considered, i.e. the channel is modeled as additive white Gaussian noise and no equalizer has to be used in the receiver.

Fig. 5 shows the BER performance after RS decoding over SNR for the TCM block syndrome decoder (BSD) and as a reference the conventional TCM Viterbi decoder. For $\ell_{min} = 9$ the BER performance of the BSD is basically identical to that of the Viterbi decoder (loss $< 0.1dB$). If the number of required zeros is decreased to $\ell_{min} = 7$, the gap between BSD and Viterbi increases. The losses at *threshold of visibility* (TOV) and at *quasi error free* (QEF) threshold are $0.15dB$ and $0.25dB$, respectively.

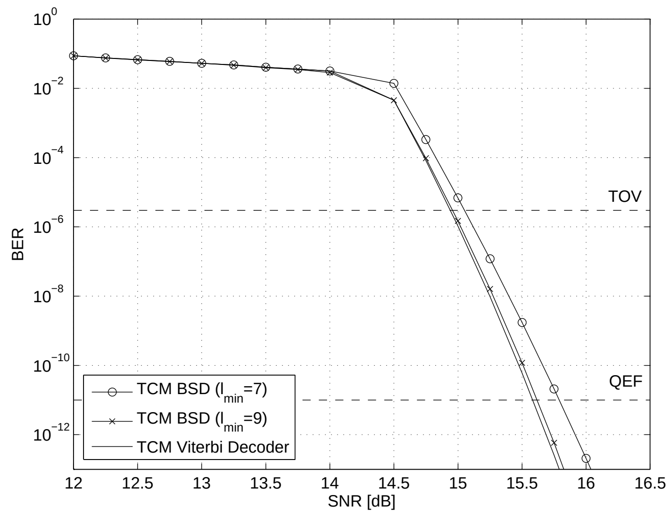

Fig. 5. BER after RS over SNR for TCM BSD and TCM Viterbi decoder.

The achievable savings in terms of decoding operations for both parameters are shown in Fig. 6. In both cases the amount of saved decoding operations is a function of SNR, while for smaller ℓ_{min} higher savings can be achieved. At the TOV about 20% of decoding operations are saved for $\ell_{min} = 9$ and 25% for $\ell_{min} = 7$. A further increase of SNR leads to significantly higher savings, which is relevant for users with above-average reception conditions (e.g. line of sight or small distance to transmitter). The results also illustrates the trade-off between BER performance and decoding effort: Decreasing ℓ_{min} leads to increased savings at the cost of lower BER performance. A further reduction can be achieved by adapting ℓ_{min} with SNR, as described for DVB-T receivers in [7].

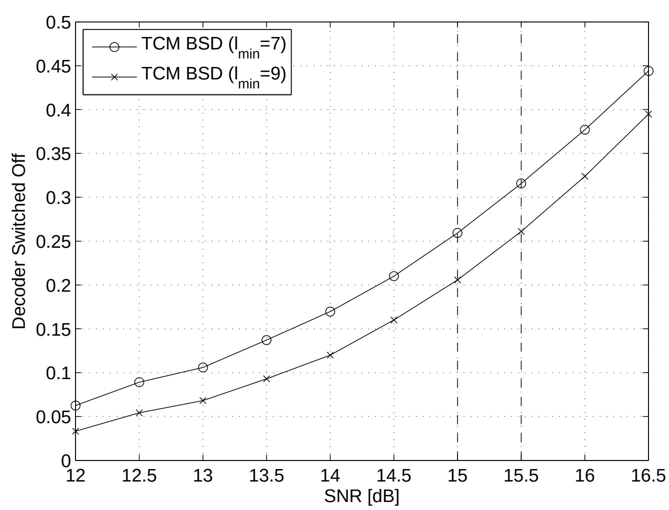

Fig. 6. Number of trellis segments not processed by decoder ($1.0 = 100\%$).

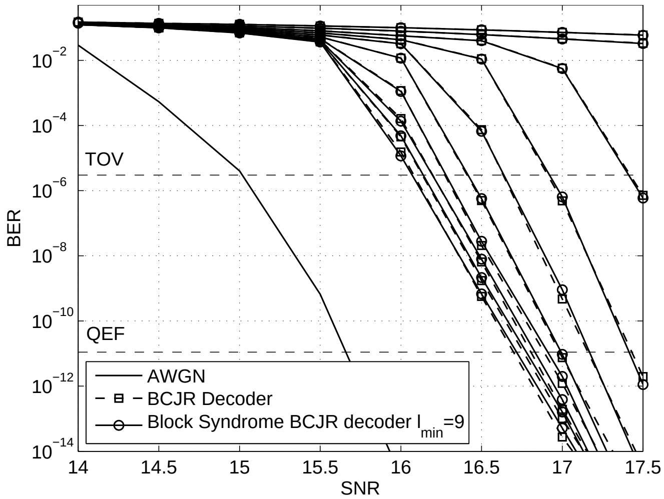

Fig. 7. BER after RS over SNR for conventional BCJR, syndrome BCJR and AWGN.

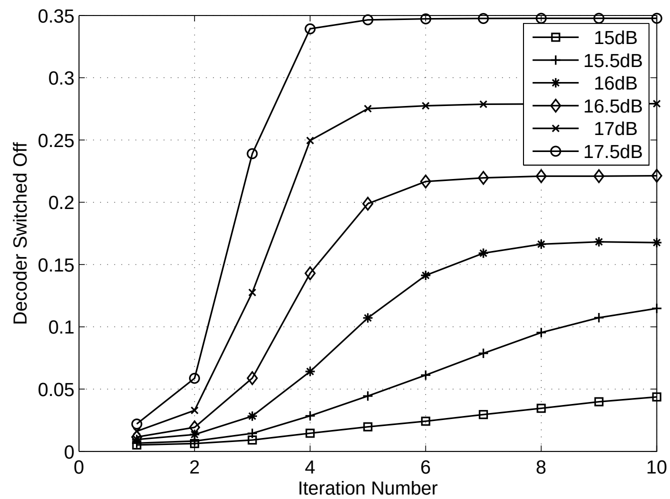

Fig. 8. Number of trellis segments not processed by decoder ($1.0 = 100\%$).

B. Iterative Turbo Receiver

Simulation results for the iterative Turbo system were obtained using transmitter and receiver as shown in Fig. 1 and 3. The channel model is based on the Brazil E profile, which describes three taps of equal power with a distance of $1\mu s$. The maximum number of iterations was set to 10. The parameters of the approximated equalizer from Section II-C were set to $N_1 = 200$, $N_2 = 200$ and $\lambda = 3.75$. To avoid BER performance losses the minimum number of zeros in $\mathbf{b}$ was set to $\ell_{min} = 9$.

Fig. 7 shows the BER performance after RS decoding for the conventional BCJR algorithm and the block syndrome BCJR algorithm over SNR for all iterations. For comparison the BER of a transmission over an AWGN channel with conventional TCM Viterbi decoder is provided. Following the results of the non-iterative receiver, for the chosen parameters the loss between BCJR and syndrome BCJR is negligible. The gap between AWGN performance and the Turbo equalized system is about $1dB$ at TOV and QEF. While Turbo equalized systems commonly approach AWGN performance in high SNR regions, this is not the case for the ATSC DTV system. The reason for this is that the interleaving scheme used in ATSC DTV is inappropriate for Turbo equalization. With a modified transmission system that applies random interleaving, the gap can be reduced to about $0.2dB$ at QEF.

The savings for the Turbo equalized system in terms of non-performed decoding operations over iteration number are provided in Fig. 8 for different SNR. As expected, the amount of savings is not only a function of SNR but also of the iteration number. While the entire sequence is processed in the first iteration, more and more error-free blocks can be found for higher iterations. This leads to savings of up to 35% for higher SNR and iteration number.

V. Conclusions

The application of syndrome based decoding methods to ATSC DTV was investigated. The syndrome based TCM decoder provides significant savings (about 20%) in terms of decoding operations for minimum reception conditions (TOV), which further increase with improving SNR. This allows an adaptation of decoding complexity and energy consumption of the ATSC DTV receiver regarding current reception conditions. For receivers employing iterative Turbo equalization further savings in decoding operations can be achieved (up to 35%), because in case of convergence the number of errors decreases with increasing iteration number.

References

[1] ATSC, "Digital Television Standard: Part 2 RF/Transmission System Characteristics (A/53, Part 2:2007)," Advanced Television Systems Committee, Tech. Rep., 2007.

[2] S. Simmons, "Breadth-first trellis decoding with adaptive effort," *IEEE Transactions on Communications*, vol. 38, no. 1, pp. 3–12, 1990.

[3] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Transactions on Inf. Theory*, vol. 28, no. 1, pp. 55 – 67, Jan. 1982.

[4] J. Geldmacher, K. Hueske, and J. Götze, "An Adaptive and Complexity Reduced Decoding Algorithm for Convolutional Codes and its Application to Digital Broadcasting Systems," in *International Conference on Ultra Modern Telecommunications (ICUMT'09)*, St. Petersburg, 2009.

[5] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo-equalization," *European Transactions on Telecommunications*, vol. 6, no. 5, pp. 507–511, 1995.

[6] H. Liu, P. Schniter, H. Fu, and R. Casas, "Frequency Domain Turbo Equalization for Vestigial Sideband Modulation with Punctured Trellis Coding," in *IEEE 7th Workshop on Signal Processing Advances in Wireless Communications (SPAWC'06)*, 2006.

[7] J. Geldmacher, K. Hueske, J. Götze, and S. Bialas, "Adaptive low complexity MAP decoding for turbo equalization," in *6th International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*, Brest, France, 2010, pp. 63 –67.

[8] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: Principles and new results," *IEEE Transactions on Communications*, vol. 50, no. 5, pp. 754–767, May 2002.

[9] A. Dejonghe and L. Vandendorpe, "Turbo-equalization for multilevel modulation: An efficient low-complexity scheme," in *IEEE International Conference on Communications (ICC'02)*, vol. 3, 2002, pp. 1863 – 1867 vol.3.

[10] J. Schalkwijk, A. Vinck, and K. Post, "Syndrome decoding of binary-rate n/k-convolutional codes," *IEEE Transactions on Information Theory*, vol. 24, no. 5, pp. 553–562, Sep 1978.

[11] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. 20, no. 2, pp. 284 – 287, March 1974.