Low Complexity Decoding for Punctured Trellis-Coded Modulation Over Intersymbol Interference Channels

Fabian Schuh and Johannes B. Huber
Institute for Information Transmission, Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
mail: {schuh, huber}@LNT.de

Abstract—Classical trellis-coded modulation (TCM) as introduced by Ungerboeck in 1976/1983 uses a signal constellation of twice the cardinality compared to an uncoded transmission with one bit of redundancy per PAM symbol, i.e., application of codes with rates \( \frac{n-1}{n} \) when \( 2^n \) denotes the cardinality of the signal constellation. The original approach therefore only comprises integer transmission rates, i.e., \( R = \{2, 3, 4 \ldots \} \), additionally, when transmitting over an intersymbol interference (ISI) channel an optimum decoding scheme would perform equalization and decoding of the channel code jointly.

In this paper, we allow rate adjustment for TCM by means of puncturing the convolutional code (CC) on which a TCM scheme is based on. In this case a nontrivial mapping of the output symbols of the CC to signal points results in a time-variant trellis. We propose an efficient technique to integrate an ISI-channel into this trellis and show that the computational complexity can be significantly reduced by means of a reduced state sequence estimation (RSSE) algorithm for time-variant trellises.

Index Terms—trellis-coded modulation (TCM); punctured convolutional codes; Viterbi-Algorithm (VA); ISI-channel;

I. INTRODUCTION

Ungerboeck’s trellis-coded modulation (TCM) \( [1] \) is a bandwidth efficient digital transmission scheme when very low overall latency is desired. Low latency is ensured by the use of convolutional codes instead of block codes (cf. \( [2] \)) and the dispense with interleaving (as opposed to conventional bit-interleaved coded modulation \( [3] \)).

Ungerboeck showed that a significant increase in the Asymptotic Coding Gain (ACG) can be achieved when considering channel coding and modulation jointly. By expanding the constellation from \( 2^{n-1} \) to \( 2^n \) signal points and employing a rate-\( \frac{n-1}{n} \) convolutional encoder one can improve the robustness of the transmission against noise by up to 6dB without any further costs besides some computational effort. However, TCM is strictly limited to integer transmission rates.

Our approach applies punctured TCM (P-TCM) with an arbitrary rate. We extend P-TCM to intersymbol interference (ISI)-channel scenarios. In this case, ML-decoding can be performed by efficiently incorporating the ISI-channel into the trellis.

We show that reduced-state sequence estimation (RSSE) can be applied in order to reduce computational complexity. We were able to show in \( [4], [5] \) that for minimum phase channels, the number of states to decode must not be significantly higher than the number of states in the channel encoder. In this paper we will describe the application of RSSE to P-TCM and discuss the partitioning of the time-variant trellis into hyperstates.

This paper deals with convolutionally encoded pulse-amplitude modulated (PAM) transmission as depicted in Fig. \( 1 \) (Here, the term PAM is used for complex-valued signal constellations \( A \) as well including amplitude-shift keying (ASK), phase-shift keying (PSK) or quadrature-amplitude modulation (QAM).) A binary data sequence \( \{u\} \) is encoded using a rate-

\[
\frac{n-1}{n_c} \text{ binary convolutional encoder } \mathcal{C} \text{ with generator polynomials } g_i(D), 1 \leq i \leq n_c; 1 \leq j \leq n_c-1, \text{ with delay operator } D, n_c-1 \text{ parallel binary-input symbols and } n_c \text{ parallel output symbols at each time instant.}
\]

At each output of the encoder, the symbols traverse through a puncturing system with puncturing \( P \) with labeling and modulation \( (n_p = 0, n_c = 2) \).

\[
u[l] \xrightarrow{\mathcal{C}} \xrightarrow{P} \xrightarrow{\mathcal{A}} y[l]
\]

Each punctured constellation \( A \) is a \( \frac{n}{n_c} \)-ary signal constellation and can be decoded with an optimum decoding scheme.

The modulated (possibly complex-valued) transmit signal traverses through a memory- \( L \) discrete-time ISI-channel with \( L + 1 \) channel coefficients \( h[k] \) with \( k \) denoting the discrete time index.

The task of the receiver is to estimate for the information bits given the transmit signal \( y[k] \) plus additive noise. Here, we focus on perfect channel knowledge at the receiver-side.
III. Punctured Trellis-Coded Modulation

In the following, we will briefly recapitulate punctured convolutional trellis coded transmission over ISI-channel scenarios as introduced in [5].

In contrast to classical TCM, our approach using punctured convolutional codes results in nontrivial mapping of coded bits to modulation symbols. As a consequence, the trellis is time-variant as already described in [5]-[7].

In order to briefly recapitulate decoding concept for punctured trellis coded modulation, we focus on 4-ary ASK-modulation, a memory-2 convolutional code, and a short puncturing scheme namely \( P = [(1 \, 1)^\top \, (0 \, 1)^\top] \). Note that, whenever the number of erased bits in one period of the puncturing scheme is not divisible by \( \log_2(M) \), the puncturing scheme has to be repeated until this condition is fulfilled. This restriction ensures that entire modulation symbols can be constructed by the finite state machine (FSM). In our example, the puncturing period (e.g., \( (1 \, 1)^\top \, (0 \, 1)^\top \)) has to be applied twice. As can be seen from the encoding process in Fig. 2, the third and the seventh encoded symbol are punctured and do not contribute to the labeling and modulation process. Thus, the second symbol, i.e., \( a[k+1] \), contains information about \( u[l+1] \) and \( u[l+2] \) and the third symbol, i.e., \( a[k+2] \), contains information about \( u[l+2] \) and \( u[l+3] \).

When equalizing (uncoded) digital PAM signaling over a discrete-time ISI-channel with \( L+1 \) tabs using DFSE (i.e., no decoding), the trellis is constructed from the first \( L \) \( \tilde{L} \) channel tabs only. Thus, the number of states is reduced from \( M^L \) to \( M^{\tilde{L}} \). The remaining \( L+1-\tilde{L} \) channel tabs are considered in a delayed decision-feedback equalization (DFE) that is performed in each trellis state using the delayed path register of the corresponding state.

The main difference to full state equalization appears in the metric computation for each time instant. From (1) it becomes clear that the state specific path register \( p_{\text{reg}}[k, s] \) is delayed by \( \tilde{L} \) and its elements are multiplied by the subsequent channel coefficients \( h_{\text{DFE}}[h] \) which have not been considered in the trellis. The branch metric \( \lambda(s, u) \) (e.g., Euclidean distance of the received symbol \( y[k] \) to the hypotheses \( h(s, u) \) for the states \( s \) and bits \( u \)) thus includes the correction factor \( \delta \):

\[
\delta = \sum_{\kappa} p_{\text{reg}}[k-\tilde{L} + \kappa, s] \cdot h_{\text{DFE}}[\kappa] \tag{1}
\]

\[
\lambda(s, u) = |y[k] - h(s, u) - \delta|^2
\]

B. Reduced State Sequence Estimation

For our coded transmission over ISI-channel we consider RSSE instead. Here, \( Z \) arbitrary MLSE states, each with \( M = 2^K \) possible branches to adjacent states, are combined.
into \( Z_R = \mathbb{Z}/2^J \); \( J \in \mathbb{N} \) each having \( 2^J \) substates and \( 2^K \cdot 2^J \) branches. A certain assignment of states to hyperstates is called a partitioning \([10]\). 

Instead of having \( 2^K \) arriving branches at each of the \( Z \) MLSE states we set a set of \( 2^K \cdot 2^J \) branches at each of the \( Z_R \) hyperstates. The total number of available branches remains \( 2^K \cdot Z \). However, when using RSSE only \( 2^K \) branches are possible (i.e., enabled) from each state, at a given time instant. The availability of branches is determined by the path registers, and thus form a decision-feedback.

The metric computation in this case can be implemented as depicted in Algorithm 1. Note that with line 6 only \( M \) branches are activated. Thus, the VA has to decide between \( M \) survivor branches at each state giving an estimate for \( \log_2(M) \) bits.

**Algorithm 1** Metric calculations for RSSE

1. \( \hat{L} \leftarrow \log_2(\text{nr. hyperstates})/\log_2(M) \)
2. \( \ell \leftarrow \log_2(\text{nr. substates})/\log_2(M) \)
3. for all \( s \in S \) do
   4. for all \( u \in A \cdot K \) do
      5. for all \( k = 1 \rightarrow \ell \) do
         6. \( \zeta[k] = p_{\text{reg}}(\text{end} - \hat{L} + k, s) \) \( \triangleright \) active branches
      7. \( \lambda(s, u) \leftarrow |g[k] - h(\zeta, u)|^2 \) \( \triangleright \) branch metric
      9. \( \lambda(s, u) \leftarrow \lambda(s, u) + \Gamma(s) \) \( \triangleright \) acc. path metric
   10. end for
5. end for
11. end for

For time-variant trellises some modifications to the underlying VA are necessary, which are described in \([5]\).

The main difference to MLSE is, that we decide for a surviving path prematurely resulting in a truncation of error events. A loss in Euclidean distance appears if an error event with minimum Euclidean distance gets truncated. Therefore the performance of RSSE strongly depends on the partitioning of the states into hyperstates. Instead of exhaustively search for the optimum state partitioning, which maximizes the intra-hyperstate distance \([10]\), we exploit the minimum phase characteristics of the ISI-channel which is, as described above, fully integrated into our trellis.

For a minimum phase channel impulse response the prior channel input symbols are weighted less than more recent ones and, thus, affect the metric less. The elder the symbols, the further back it is stored in the vector presentation of a particular trellis state. Hence, the intra-hyperstate distance is maximized when states are combined with respect to elder positions in the state number. This particular partitioning is equivalent to DFSE for ISI-channels (which is the optimum partitioning for equalization of minimum phase ISI-channels \([10]\)) and will in the latter be called DFSE partitioning. As the minimum phase ISI-channel is the last element to affect the transmitted symbols and is also fully integrated into the trellis, we can apply the DFSE partitioning to use RSSE for P-TCM over ISI-channels. An implementation of this set partitioning for the \( J \)th level exploiting the minimum phase characteristics is shown in Algorithm 2. The columns in the resulting matrix \( p(z, i) \) define the states that have to be grouped into hyperstates.

**Algorithm 2** DFSE Partitioning for RSSE

1. \( i, z \leftarrow 1 \)
2. for \( \ell = 1 \rightarrow Z \) do
   3. if \( \ell \mod J = 0 \) then
      4. \( z \leftarrow z + 1 \)
   5. end if
   6. \( p(z, i) = \ell \)
   7. \( i \leftarrow i + 1 \)
8. end for

In the following we will focus on our state design and two possibilities to apply DFSE partitioning to time-variant trellises.

**C. State Design**

In Fig. 5 a single trellis state in the VA is depicted as a FIFO. Input values to the FSM are represented by the branches at the left-hand side, whereas values that drop off the FIFO are stored within the state-specific path register \( p_{\text{reg}}(k, s) \) at the right-hand side. During decoding when entering a trellis state that is split, \( i.e., \), has an increased number of states, the two possibilities for the punctured bit are tracked using an additional delay element, indicated by the hatched block \( \mathbb{Z} \).

When DFSE partitioning is performed, the states can be reduced as shown in Fig. 5. There, fewer FIFO elements are used to define the trellis state, while the remaining ones are used as feedback to enable branches for the next trellis step for that particular state.

An implementation of this algorithm needs to ensure that when entering a split trellis segment, \( i.e., \), increased number of states, the right path register is chosen as source for the decision feedback.

**D. State Partitioning**

As already mentioned, a DFSE partitioning of the first order, \( i.e., \), reducing the number of states in each trellis segment by a factor of two, combining states that differ in the eldest position, into hyperstates. Hence, when applied to a punctured TCM, each trellis segment undergoes the same partitioning. A resulting reduced-state trellis is depicted in Fig. 6. For \( J = 1 \) and \( J = 2 \). As a consequence, non-existing states from the original trellis are also partitioned (\( cf., \) first trellis segment).
Due to the reduction of non-existing states, and hence reduced minimum Euclidean distance, we propose to partition those segments first that are split and keeping the others unpartitioned. As can be seen from the first-level state reduction in Fig. 6 (right-hand side, above), the first segment contains two transitions in each state, and thus is not partitioned, whereas in segment two and three, each state has four transitions, due to the state partitioning. Apparently, the first segment may be handled with a full state VA, while the other two segments need to be decoded via RSSE using the path register in each state.

The right-hand side of Fig. 6 show the segment-dependent partitioning and state reduction for \( J = 1 \) and \( J = 2 \). The segments for \( J = 1 \) show four, and eight transitions per state, respectively. At this point RSSE has to consider a different amount of information in the path register for each segment.

The resulting trellis shows a less decreased minimum Euclidean distance when compared to the uniform partitioning but has a slightly higher computational complexity because of the extra states. Thus, the segment-dependent partitioning technique enables an even more flexible way to trade between complexity and performance.

V. NUMERICAL RESULTS

In this section we will give numerical simulation results and investigate several ISI-channels for punctured TCM. We analyse the decoder complexity as \textit{number of branch metric calculations per information bit} and show that this approach enables a flexible trade-off between computational complexity and performance.

Due to the minimum phase characteristics of the ISI-channel the partitioning of the trellis states into hyperstates leads to the smallest possible loss in Euclidean distance. Hence, if, for instance, the ISI-channel is an equal tap delay line, the loss in Euclidean distance is higher than for an exponentially decaying channel because of the premature decisions for a surviving path.

To see this effect we conducted simulations over different ISI-channels of unit energy and plotted the complexity number over the required \( E_{\text{INR}} \) to achieve a bit error probability of less than \( 10^{-3} \). The unit energy channels are defined as:

\[
\begin{align*}
    h_{\text{exp}}[\kappa] &= \frac{1}{\sqrt{\sum_{\kappa} |h_{\text{exp}}[\kappa]|^2}} \exp\left(-\kappa/\kappa_0\right) \quad \text{for } 0 \leq \kappa \leq L \\
    h_{\text{lin}}[\kappa] &= \frac{1}{\sqrt{\sum_{\kappa} |h_{\text{lin}}[\kappa]|^2}} \frac{L - \kappa + 1}{L + 1} \quad \text{for } 0 \leq \kappa \leq L
\end{align*}
\]

The results can be seen in Fig. 7. As should be clear to the reader, the loss in Euclidean distance is smallest for an exponentially decaying ISI-channel.

VI. CONCLUSION

It has been shown that TCM can be extended by puncturing. Furthermore, an efficient MLSE decoder for P-TCM over ISI-channels was proposed and investigated.

The numerical simulation results clearly show that we can achieve a soft trade-off between spectral and power efficiency easier and more flexibly than by means of traditional TCM.

REFERENCES

[1] G. Ungerboeck, “Trellis-coded modulation with redundant signal sets; part i: Introduction; part ii: State of the art,” Communications Magazine, IEEE, vol. 25, no. 2, pp. 5 –21, February 1987.
[2] T. Hehn and J. B. Huber, “LDPC Codes and Convolutional Codes with Equal Structural Delay: A Comparison,” IEEE Transactions on Communications, vol. 57, no. 6, pp. 1683–1692, June 2009.
[3] E. Zehavi, “8-PSK Trellis Codes for a Rayleigh Channel,” IEEE Transactions on Communications, vol. 40, no. 5, pp. 873 –884, may 1992.
[4] F. Schuh, A. Schenk, and J. Huber, “Reduced complexity Super-Trellis decoding for convolutionally encoded transmission over ISI-Channels,” in 2013 International Conference on Computing, Networking and Communications, Signal Processing for Communications Symposium (ICNC’13 - SPC), San Diego, USA, Jan. 2013.
[5] ——, “Matched decoding for punctured convolutional encoded transmission over ISI-Channels,” in 9th International ITG Conference on Systems, Communications and Coding 2013 (SCC’2013), Munich, Germany, Jan. 2013.
[6] T. Woerz and R. Schweikert, “Performance of punctured pragmatic codes,” in Global Telecommunications Conference, 1995. GLOBECOM ’95., IEEE, vol. 1, nov 1995, pp. 664 –669 vol.1.
[7] F. Schuh, A. Schenk, and J. Huber, “Punctured Trellis-Coded Modulation,” in submitted to ICC 2014, Jun. 2014.
[8] M. Eyuboglu and S. Qureshi, “Reduced-State Sequence Estimation With Set Partitioning and Decision Feedback,” IEEE Trans. Commun., vol. 36, no. 1, pp. 13–20, Jan. 1988.
[9] A. Duel-Hallen and C. Heegard, “Delayed Decision-Feedback Sequence Estimation,” IEEE Trans. Commun., vol. 37, no. 5, pp. 428–436, May 1989.
[10] B. Spinnen and J. Huber, “Design of Hyper States for Reduced-State Sequence Estimation,” in Proc. IEEE Int. Conf. Communications ICC ’95 Seattle, vol. 1, 1995, pp. 1–6.