APPENDIX C

DECODING T-TCM WITH NON-BINARY SYMBOL-BY-SYMBOL MAP ALGORITHM

As stated in Section about decoding of T-TCM has to rely on symbol-by-symbol MAP decoding of component recursive systematic TCM code. In this section, the symbol-by-symbol MAP algorithm for a nonbinary binary trellis is presented first as an extension of the MAP algorithm for binary trellis. Then the utilization of this algorithm in T-TCM decoder is described. The discussion follows (**Robertson and Woerz,** 1996, 1997).

The MAP Algorithm for Non-binary Trellis

Consider the MAP decoding for a classical TCM scheme. The trellis diagram of TCM encoder is essentially the same as one for a binary trellis. The main difference is that there are 2^{m-1} transitions stemming and merging at each state corresponding to one of 2^{m-1} different groups of information bits u_k. The output at each branch is an M-ary symbol. The MAP algorithm takes the *a priori* information on each group of information bits u_k as its input. Similar to MAP algorithm for binary trellis in Appendix A, the branch transition probability at time k is,

$$\gamma_k^i(s_{k-1}, s_k) = p(\mathbf{y}_k | u_k = i, s_k, s_{k-1}) \cdot q(u_k = i | s_k, s_{k-1}) \cdot P(s_k | s_{k-1}) \qquad \text{(C-1)}$$

where $q(u_k = i | s_k, s_{k-1})$ is equal to zero or one depending on whether the encoder input $u_k=i$, $i=1,2, . . ., 2^{m-1}$ is associated with the transition from state s_{k-1} to s_k or not. The received vector $\boldsymbol{y} = \{y_1, y_2, . . ., y_N\}$ is the TCM encoder output sequence that has been disturbed by AWGN with variance σ_N^2. The last component in (C-1) is the *a priori* information if $q(u_k = i | s_k, s_{k-1})=1$.

$$P(s_k | s_{k-1}) = P(u_k = i) \qquad \text{(C-2)}$$

If there does not exist a i such that $q(u_k = i | s_k, s_{k-1})=1$ then $P(s_k | s_{k-1})=0$.

The forward and backward recursion to calculate $\alpha_k(s_k)$ and $\beta_k(s_k)$ are essentially the same as in (A-3) and (A-4). Then the output of the MAP algorithm is,

$$P(u_k = i | \mathbf{y}) = C \cdot \sum_{s_k} \sum_{s_{k-1}} \gamma_k^i(s_{k-1}, s_k) \cdot \alpha_{k-1}(s_{k-1}) \cdot \beta_k(s_k), \quad i = 1, . . ., 2^{m-1} \qquad \text{(C-3)}$$

The constant C can be eliminated by normalization. Similarly as in Appendix A, the logarithm of probability in (C-3) can be approximated via logarithms of probabilities $\gamma_k^i(s_{k-1}, s_k)$, $\alpha_k(s_k)$, $\beta_k(s_k)$ as,

$$LogP(u_k = i | \mathbf{y}) = \max_{s_k, s_{k-1}} \left(\overline{\gamma}_k^i(s_{k-1}, s_k) + \overline{\alpha}_{k-1}(s_{k-1}) + \overline{\beta}_k(s_k) \right) \quad i = 1, . . ., 2^{m-1} \qquad \text{(C-4)}$$

The term in (C-4) comprises of *a priori*, *systematic* and *extrinsic* (*e&s*) components. The second component is passed to the next decoder in which it is used as *a priori* information. The second component can be found by subtracting the $LogP(u_k = i)$ from the logarithm in (C-4) as follows.

$$L_{e\&s}(u_k = i) = LogP(u_k = i|\mathbf{y}) - LogP(u_k = i) \quad i = 1, \; . \; . \; ., \; 2^{m-1} \tag{C-5}$$

The term in (C-5) is independent from the *a priori* information $LogP(u_k = i)$ since the term $P(u_k = i)$ is a factor in (C-1) that does not depend on s_{k-1} and s_k and can be put outside of the summations in (C-3).

C.2 T-TCM Decoder with the MAP Algorithm for Non-binary Trellis

The complete decoder of T-TCM was illustrated in Fig. 4.20 and is reproduced in Fig. C.1 for ease of reference. The upper decoder see a punctured symbol, the corresponding symbol from the upper encoder was not transmitted. The upper decoder now ignores this symbol as far as the direct channel input is concerned. In (C-1) we set $Logp(\mathbf{y}_k|u_k = i, s_k, s_{k-1}) = 0$. The only input for this step in the trellis is *a prirori* information **a** from the other decoder and this contain the systematic information ***s***. At this point the output of the MAP is the sum of this *a priori* information **a** and newly computed extrinsic information **e**. Then the *a priori* information is subtracted and the extrinsic information **e** is given to the second decoder as its *a priori* information. The second decoder, however sees a symbol that was transmitted by its encoder, and it can compute $Logp(\mathbf{y}_k|u_k = i, s_k, s_{k-1})$ for each i and $L_{e\&s}(u_k = i)$ which is used as the *a priori* input of the upper decoder in the next iteration.

The discussion above applies only to the decoding process where the *a priori* information for the upper decoder is already available. The situation in the very first decoding step is different. Before the first decoding of the upper decoder, it is needed to set the *a priori* information to contain the systematic information for the "*" transitions, where the transmitted symbol was determined by the information group u_k and by the unknown parity bit $b_k^* \in \{0,1\}$ of the other decoder. By using the Bayes' rule, the *a priori* information is set as follows.

$$\begin{aligned} P(u_k = i) \leftarrow P(u_k = i|y_k) &= const \cdot p(y_k|u_k = i) \\ &= const \cdot \sum_{j\in\{0,1\}} p(y_k, b_k^* = j|u_k = i) \\ &= \frac{const}{2} \cdot \sum_{j\in\{0,1\}} p(y_k|u_k = i, b_k^* = j) \end{aligned} \tag{C-6}$$

where it is assumed that $P(b_k^* = j|u_k) = P(b_k^* = j) = \frac{1}{2}$ and the initial *a priori* probability of u_k prior to any decoding is constant for all i. If the upper decoder is not at a "*" positions, then we simply set $P(u_k = i) = \frac{1}{2^{m-1}}$. The "metric *s*" in Fig. 4.20 (or Fig. C.1) means the evaluation of (C-6).

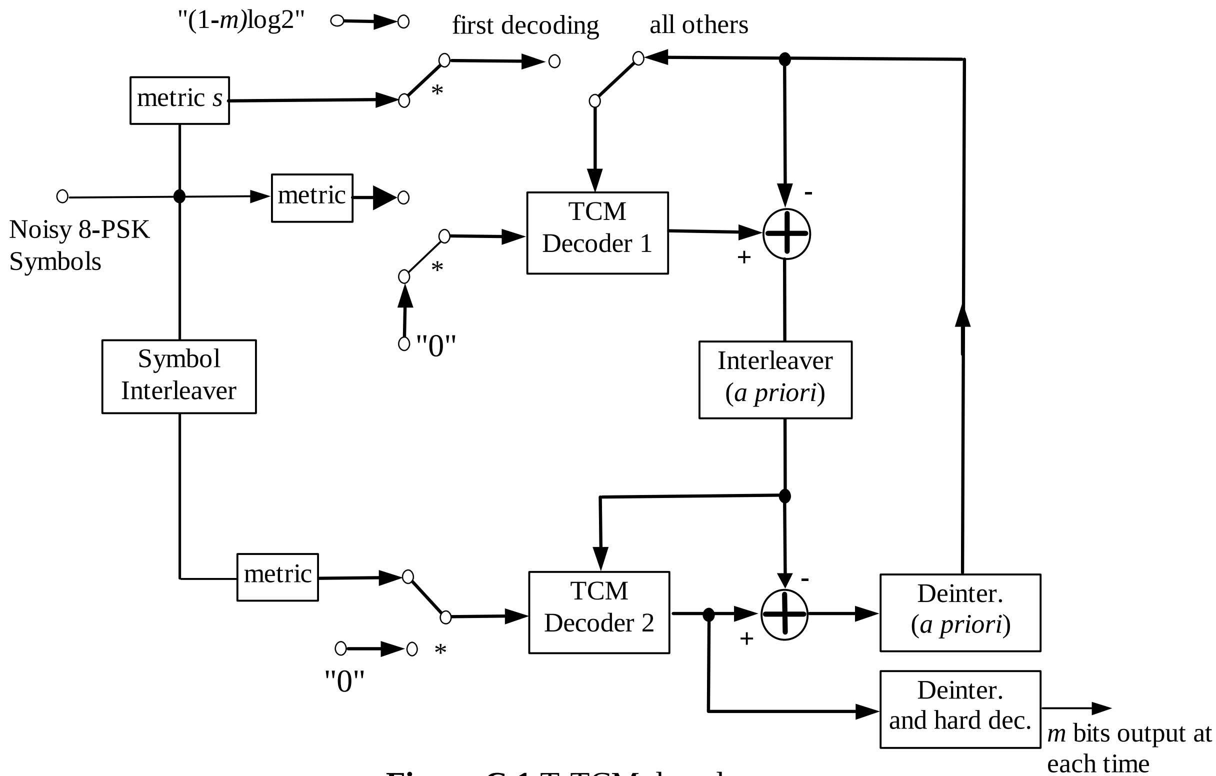

Figure C.1 T-TCM decoder