Turbo Decoding

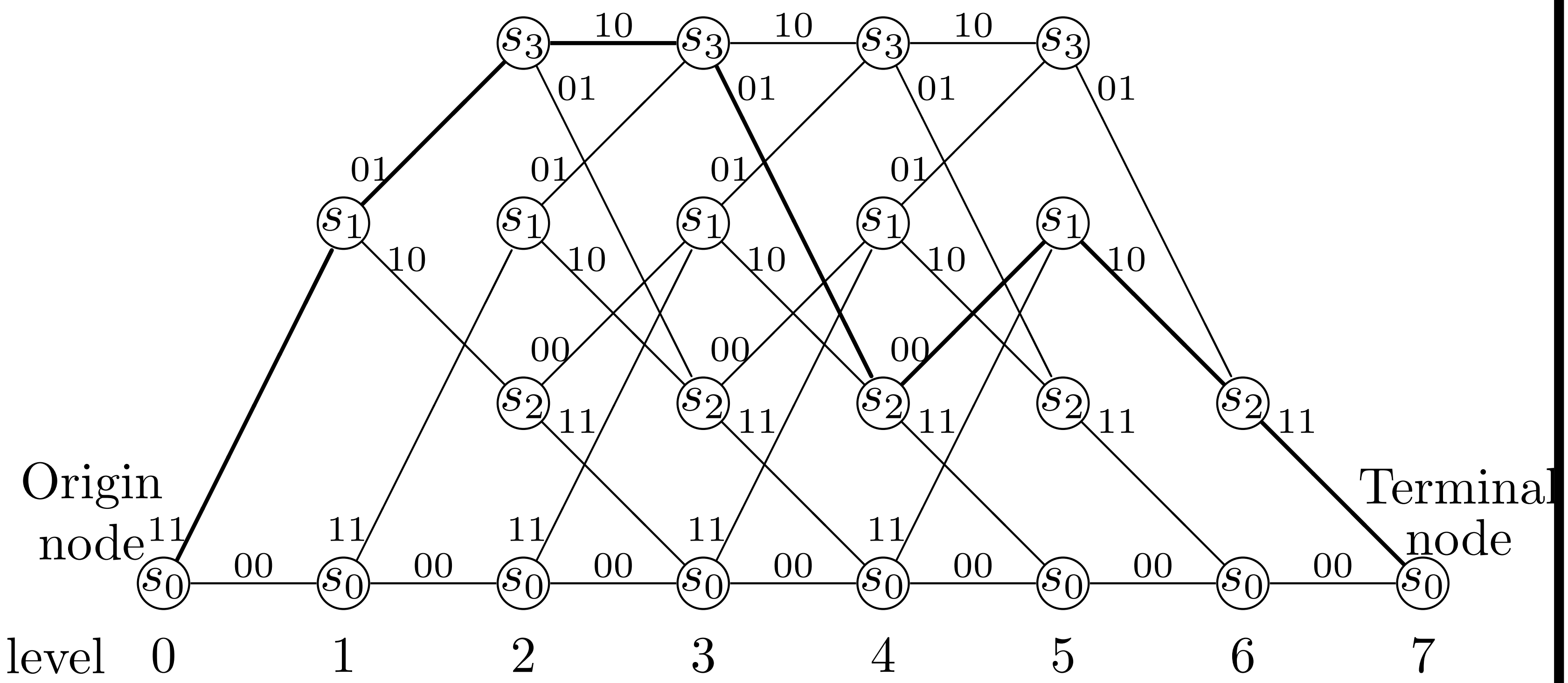
Yunghsiang S. Han

Department of Electrical Engineering
National Taiwan University of Science and Technology
Taiwan

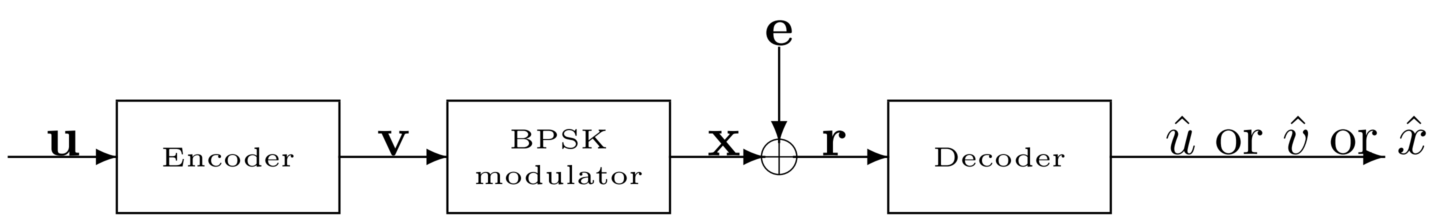
E-mail: yshan@mail.ntust.edu.tw

Trellises of Convolutional Codes

1. a code *trellis* as termed by Forney is a structure obtained from a code tree by merging those nodes in the same *state*.
2. Recall that the *state* associated with a node is determined by the associated shift-register contents.
3. For a binary (n, k, m) convolutional code, the number of states at levels m through L is 2^K, where $K = \sum_{j=1}^{k} K_j$ and K_j is the length of the jth shift register in the encoder; hence, there are 2^K nodes on these levels.
4. Due to node merging, only one terminal node remains in a trellis.
5. Analogous to a code tree, a path from the single origin node to the single terminal node in a trellis also mirrors a codeword.


Origin
node
Terminal
node
level 0 1 2 3 4 5 6 7
s_0 s_1 s_2 s_3
00 11 01 10

System Model[a]



For an $(n, 1, m)$ convolutional code:

- $\mathbf{u} = (u_0, u_1, \ldots, u_{L-1}) \in \{0, 1\}^L$ is the information bit sequence.
- $\mathbf{v} = (v_0, v_1, \ldots, v_{N-1}) \in \{0, 1\}^N$ is the code bit sequence.
- $\mathbf{x} = (x_0, x_1, \ldots, x_{N-1}) \in \{-1, 1\}^N$ is the BPSK-modulated code bit sequence, where $x_j = (-1)^{v_j}$.
- $\mathbf{e} = (e_0, e_1, \ldots, e_{N-1}) \in \Re^N$ is a zero-mean i.i.d Gaussian

[a] All material of MAP algorithm and turbo codes are adapted from the lecture note by Prof. Po-Ning Chen

vector (AWGN).

- $\mathbf{r} = (r_0, r_1, \ldots, r_{N-1}) \in \Re^N$ is the received vector, where $r_j = x_j + e_j$.
- $\hat{\mathbf{u}} = (\hat{u}_0, \hat{u}_1, \ldots, \hat{u}_{L-1}) \in \{0, 1\}^L$ is reconstructed information bit sequence.

BCJR (MAP) Algorithm (1)

1.

$$
\begin{aligned}
\Pr\{u_i = 0|\mathbf{r}\} &= \sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \Pr\left\{ S_s^i, S_{\bar{s}}^{i+1} \middle| \mathbf{r} \right\} \\
&= \sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \frac{\Pr\left\{ S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r} \right\}}{\Pr\left\{ \mathbf{r} \right\}},
\end{aligned}
$$

where $\mathcal{B}_i^{(0)}$ is the set of transition from nodes at level i to nodes at level $i+1$, which corresponds to input $u_i = 0$.

$$\begin{aligned}\Pr\{u_i = 1|\mathbf{r}\} &= \sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \Pr\left\{ S_s^i, S_{\bar{s}}^{i+1} \middle| \mathbf{r} \right\} \\ &= \sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \frac{\Pr\left\{S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}\right\}}{\Pr\left\{\mathbf{r}\right\}},\end{aligned}$$

where $\mathcal{B}_i^{(1)}$ is the set of transition from nodes at level i to nodes at level $i+1$, which corresponds to input $u_i = 1$. Therefore,

$$\Lambda(i) = \log \frac{\displaystyle\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \Pr\left\{S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}\right\}}{\displaystyle\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \Pr\left\{S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}\right\}}$$

2. Define

$$
\begin{aligned}
\alpha(S_{\bar{s}}^{i}) &\stackrel{\triangle}{=} \Pr\{S_{\bar{s}}^{i}, \mathbf{r}_0^{in-1}\} \\
\beta(S_{\bar{s}}^{i}) &\stackrel{\triangle}{=} \Pr\left\{ \mathbf{r}_{in}^{N-1} \middle| S_{\bar{s}}^{i} \right\} \\
\gamma(u, S_s^i, S_{\bar{s}}^{i+1}) &\stackrel{\triangle}{=} \Pr\left\{ u_i = u, S_{\bar{s}}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \middle| S_s^i \right\}, \quad u = 0, 1,
\end{aligned}
$$

where $\mathbf{r}_a^b = (r_a, r_{a+1}, \ldots, r_b)$.

$$
\begin{aligned}
& \Pr\left\{S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}\right\} \\
= \quad & \Pr\left\{S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}_0^{in-1}, \mathbf{r}_{in}^{(i+1)n-1}, \mathbf{r}_{(i+1)n}^{N-1}\right\} \\
= \quad & \Pr\left\{\mathbf{r}_{(i+1)n}^{N-1} \middle| S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}_0^{in-1}, \mathbf{r}_{in}^{(i+1)n-1}\right\} \\
& \times \Pr\left\{S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}_0^{in-1}, \mathbf{r}_{in}^{(i+1)n-1}\right\} \\
= \quad & \Pr\left\{\mathbf{r}_{(i+1)n}^{N-1} \middle| S_{\bar{s}}^{i+1}\right\} \Pr\left\{S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}_0^{in-1}, \mathbf{r}_{in}^{(i+1)n-1}\right\} \\
= \quad & \beta(S_{\bar{s}}^{i+1}) \cdot \Pr\left\{S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}_0^{in-1}, \mathbf{r}_{in}^{(i+1)n-1}\right\} \\
= \quad & \beta(S_{\bar{s}}^{i+1}) \cdot \Pr\left\{S_{\bar{s}}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \middle| S_s^i, \mathbf{r}_0^{in-1}\right\} \Pr\left\{S_s^i, \mathbf{r}_0^{in-1}\right\} \\
= \quad & \Pr\left\{S_s^i, \mathbf{r}_0^{in-1}\right\} \beta(S_{\bar{s}}^{i+1}) \Pr\left\{S_{\bar{s}}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \middle| S_s^i, \mathbf{r}_0^{in-1}\right\} \\
= \quad & \Pr\left\{S_s^i, \mathbf{r}_0^{in-1}\right\} \beta(S_{\bar{s}}^{i+1}) \Pr\left\{S_{\bar{s}}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \middle| S_s^i\right\} \\
= \quad & \alpha(S_s^i) \beta(S_{\bar{s}}^{i+1}) \sum_{u=0}^{1} \gamma(u, S_s^i, S_{\bar{s}}^{i+1})
\end{aligned}
$$

3. **Derivation of** α: $\alpha(S_0^0)=1$; $\alpha(S_1^0)=\alpha(S_2^0)=\alpha(S_3^0)=0$.
For $i=1,\ldots,L$,

$$\begin{aligned}
\alpha(S_{\bar{s}}^i) &\stackrel{\triangle}{=} \Pr\left\{S_{\bar{s}}^i, \mathbf{r}_0^{in-1}\right\} \\
&= \sum_{s=0}^{3} \Pr\left\{S_s^{i-1}, S_{\bar{s}}^i, \mathbf{r}_0^{in-1}\right\} \\
&= \sum_{s=0}^{3} \Pr\left\{S_s^{i-1}, S_{\bar{s}}^i, \mathbf{r}_0^{(i-1)n-1}, \mathbf{r}_{(i-1)n}^{in-1}\right\} \\
&= \sum_{s=0}^{3} \Pr\left\{S_s^{i-1}, \mathbf{r}_0^{(i-1)n-1}\right\} \Pr\left\{ S_{\bar{s}}^i, \mathbf{r}_{(i-1)n}^{in-1} \,\middle|\, S_s^{i-1}, \mathbf{r}_0^{(i-1)n-1}\right\} \\
&= \sum_{s=0}^{3} \alpha(S_s^{i-1}) \Pr\left\{ S_{\bar{s}}^i, \mathbf{r}_{(i-1)n}^{in-1} \,\middle|\, S_s^{i-1}, \mathbf{r}_0^{(i-1)n-1}\right\} \\
&= \sum_{s=0}^{3} \alpha(S_s^{i-1}) \Pr\left\{ S_{\bar{s}}^i, \mathbf{r}_{(i-1)n}^{in-1} \,\middle|\, S_s^{i-1}\right\} \\
&= \sum_{s=0}^{3} \alpha(S_s^{i-1}) \sum_{u=0}^{1} \gamma(u, S_s^{i-1}, S_{\bar{s}}^i).
\end{aligned}$$

4. **Derivation of** β: $\beta(S_0^L)=1$; $\beta(S_1^L)=\beta(S_2^L)=\beta(S_3^L)=0$.

For $i = L-1, \ldots, 0$,

$$
\begin{aligned}
\beta(S_{\bar{s}}^{i}) &\triangleq \Pr\left\{ \mathbf{r}_{in}^{N-1} \,\middle|\, S_{\bar{s}}^{i} \right\} \\
&= \sum_{s=0}^{3} \Pr\left\{ S_{s}^{i+1}, \mathbf{r}_{in}^{N-1} \,\middle|\, S_{\bar{s}}^{i} \right\} \\
&= \sum_{s=0}^{3} \frac{\Pr\left\{ S_{\bar{s}}^{i}, S_{s}^{i+1}, \mathbf{r}_{in}^{N-1} \right\}}{\Pr\left\{ S_{\bar{s}}^{i} \right\}} \\
&= \sum_{s=0}^{3} \frac{\Pr\left\{ S_{\bar{s}}^{i}, S_{s}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1}, \mathbf{r}_{(i+1)n}^{N-1} \right\}}{\Pr\left\{ S_{\bar{s}}^{i} \right\}} \\
&= \sum_{s=0}^{3} \frac{\Pr\left\{ \mathbf{r}_{(i+1)n}^{N-1} \,\middle|\, S_{\bar{s}}^{i}, S_{s}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \right\} \Pr\left\{ S_{\bar{s}}^{i}, S_{s}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \right\}}{\Pr\left\{ S_{\bar{s}}^{i} \right\}} \\
&= \sum_{s=0}^{3} \frac{\Pr\left\{ \mathbf{r}_{(i+1)n}^{N-1} \,\middle|\, S_{s}^{i+1} \right\} \Pr\left\{ S_{\bar{s}}^{i}, S_{s}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \right\}}{\Pr\left\{ S_{\bar{s}}^{i} \right\}}
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{s=0}^{3} \frac{\beta(S_s^{i+1}) \Pr\left\{S_{\bar{s}}^{i}, S_s^{i+1}, \mathbf{r}_{in}^{(i+1)n-1}\right\}}{\Pr\left\{S_{\bar{s}}^{i}\right\}} \\
&= \sum_{s=0}^{3} \frac{\beta(S_s^{i+1}) \Pr\left\{ S_s^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \middle| S_{\bar{s}}^{i}\right\} \Pr\left\{S_{\bar{s}}^{i}\right\}}{\Pr\left\{S_{\bar{s}}^{i}\right\}} \\
&= \sum_{s=0}^{3} \beta(S_s^{i+1}) \Pr\left\{ S_s^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \middle| S_{\bar{s}}^{i}\right\} \\
&= \sum_{s=0}^{3} \beta(S_s^{i+1}) \sum_{u=0}^{1} \Pr\left\{ u, S_s^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \middle| S_{\bar{s}}^{i}\right\} \\
&= \sum_{s=0}^{3} \beta(S_s^{i+1}) \sum_{u=0}^{1} \gamma(u, S_{\bar{s}}^{i}, S_s^{i+1}).
\end{aligned}
$$

5. **Derivation of** γ: $\beta(S_0^L) = 1$; $\beta(S_1^L) = \alpha(S_2^L) = \alpha(S_3^L) = 0$.

$$
\begin{aligned}
\gamma(u, S_s^i, S_{\bar{s}}^{i+1}) &\triangleq \Pr\left\{ u, S_{\bar{s}}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1} \,\middle|\, S_s^i \right\} \\
&= \frac{\Pr\left\{u, S_s^i, S_{\bar{s}}^{i+1}, \mathbf{r}_{in}^{(i+1)n-1}\right\}}{\Pr\{S_s^i\}} \\
&= \frac{\Pr\left\{ \mathbf{r}_{in}^{(i+1)n-1} \,\middle|\, u, S_s^i, S_{\bar{s}}^{i+1} \right\} \Pr\left\{u, S_s^i, S_{\bar{s}}^{i+1}\right\}}{\Pr\{S_s^i\}} \\
&= \frac{\Pr\left\{ \mathbf{r}_{in}^{(i+1)n-1} \,\middle|\, u, S_s^i \right\} \Pr\left\{u, S_s^i, S_{\bar{s}}^{i+1}\right\}}{\Pr\{S_s^i\}} \\
&= \frac{\Pr\left\{ \mathbf{r}_{in}^{(i+1)n-1} \,\middle|\, \mathbf{x}_{in}^{(i+1)n-1} \right\} \Pr\left\{u, S_s^i, S_{\bar{s}}^{i+1}\right\}}{\Pr\{S_s^i\}} \\
&= \frac{\Pr\left\{ \mathbf{r}_{in}^{(i+1)n-1} \,\middle|\, \mathbf{x}_{in}^{(i+1)n-1} \right\} \Pr\left\{u \,\middle|\, S_s^i, S_{\bar{s}}^{i+1} \right\} \Pr\left\{S_s^i, S_{\bar{s}}^{i+1}\right\}}{\Pr\{S_s^i\}}
\end{aligned}
$$

$$
\begin{aligned}
&= \Pr\left\{ \mathbf{r}_{in}^{(i+1)n-1} \middle| \mathbf{x}_{in}^{(i+1)n-1} \right\} \Pr\left\{ u \middle| S_s^i, S_{\bar{s}}^{i+1} \right\} \Pr\left\{ S_{\bar{s}}^{i+1} \middle| S_s^i \right\} \\
&= \begin{cases} \Pr\left\{ S_{\bar{s}}^{i+1} \middle| S_s^i \right\} \Pr\left\{ \mathbf{r}_{in}^{(i+1)n-1} \middle| \mathbf{x}_{in}^{(i+1)n-1} \right\}, & (S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(u)}; \\ 0, & \text{otherwise}; \end{cases} \\
&= \begin{cases} \dfrac{\Pr\left\{ S_{\bar{s}}^{i+1} \middle| S_s^i \right\}}{\sqrt{2\pi}\sigma} \exp\left\{ -\dfrac{\sum_{j=in}^{(i+1)n-1} (r_j - x_j(u))^2}{2\sigma^2} \right\}, & (S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(u)}; \\ 0, & \text{otherwise}; \end{cases} \\
&= \begin{cases} \dfrac{\Pr\left\{ u_i = u \right\}}{\sqrt{2\pi}\sigma} \exp\left\{ -\dfrac{\sum_{j=in}^{(i+1)n-1} (r_j - x_j(u))^2}{2\sigma^2} \right\}, & (S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(u)}; \\ 0, & \text{otherwise}. \end{cases}
\end{aligned}
$$

Note:

- x_j is indeed a function of u (or u_i) and S_s^i. For notation simplification, we denote it by $x_j(u)$.
- $\Pr\left\{ u \middle| S_s^i, S_{\bar{s}}^{i+1} \right\}$ is either 1 or 0.
- The initial value of $\Pr\left\{ S_{\bar{s}}^{i+1} \middle| S_s^i \right\}$ can be chosen as $1/K$, where K is the number of edges on trellis, which starts from node S_s^i and ends at a node at level $i+1$.

BCJR (MAP) Algorithm (2)

step 1: Forward recursion

step 1-1:

- $\alpha(S_0^0) = 1$; $\alpha(S_1^0) = \alpha(S_2^0) = \alpha(S_3^0) = 0$.

step 1-2:

- For $i = 0, \ldots, L-1$, $s = 0, 1, 2, 3$, $\bar{s} = 0, 1, 2, 3$, $u = 0, 1$, compute

$$\gamma(u, S_s^i, S_{\bar{s}}^{i+1}) = \begin{cases} \dfrac{\Pr\{u_i = u\}}{\sqrt{2\pi}\sigma} e^{-\frac{\sum_{j=in}^{(i+1)n-1}(r_j - x_j(u))^2}{2\sigma^2}}, & (S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(u)} \\ 0, & \text{otherwise.} \end{cases}$$

step 1-3:

- For $i = 1, \ldots, L$ and $\bar{s} = 0, 1, 2, 3,$

$$\alpha(S_{\bar{s}}^{i}) = \sum_{s=0}^{3} \alpha(S_{s}^{i-1}) \sum_{u=0}^{1} \gamma(u, S_{s}^{i-1}, S_{\bar{s}}^{i}).$$

step 2: Backward recursion

step 2-1:

- $\beta(S_0^L) = 1$; $\beta(S_1^L) = \beta(S_2^L) = \beta(S_3^L) = 0.$

step 2-2:

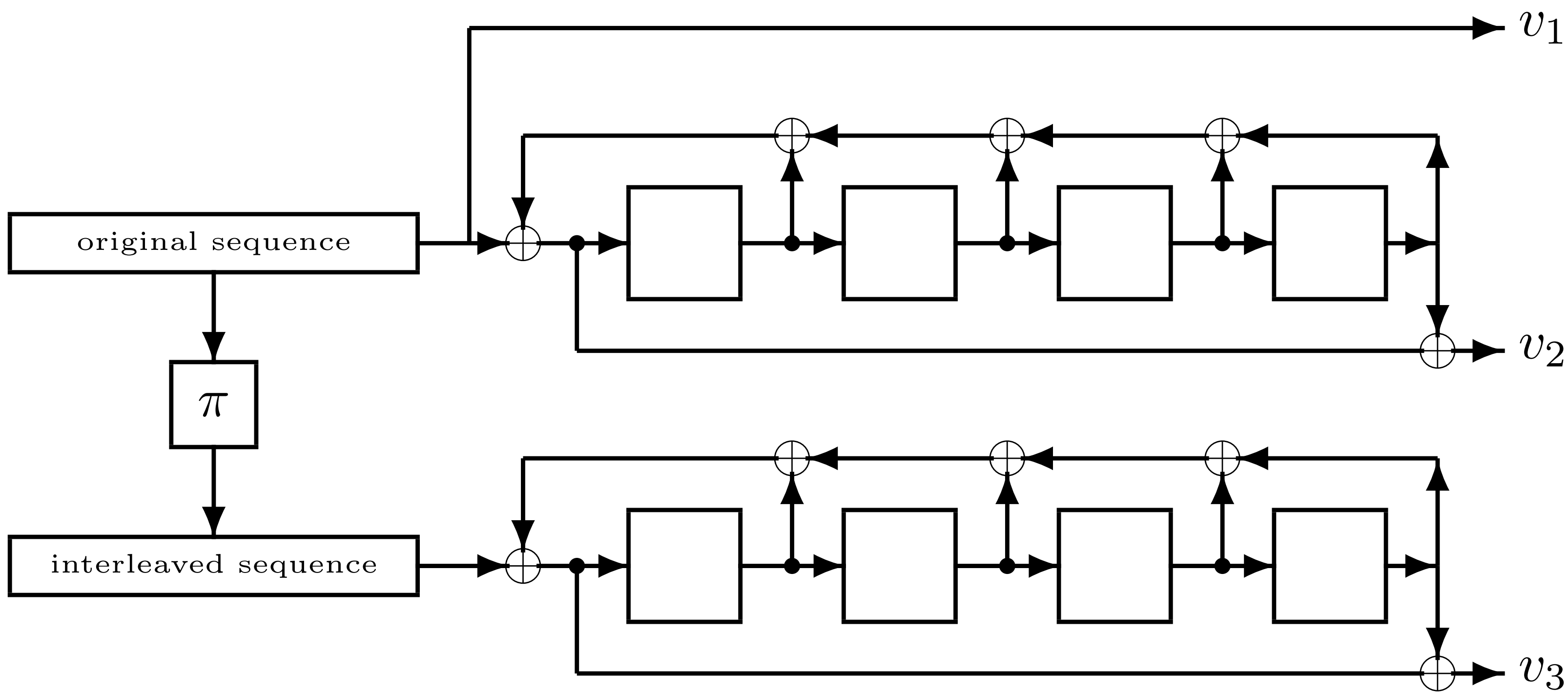
- For $i = L - 1, \ldots, 0$ and $\bar{s} = 0, 1, 2, 3,$ compute

$$\beta(S_{\bar{s}}^{i}) = \sum_{s=0}^{3} \beta(S_{s}^{i+1}) \sum_{u=0}^{1} \gamma(u, S_{\bar{s}}^{i}, S_{s}^{i+1}).$$

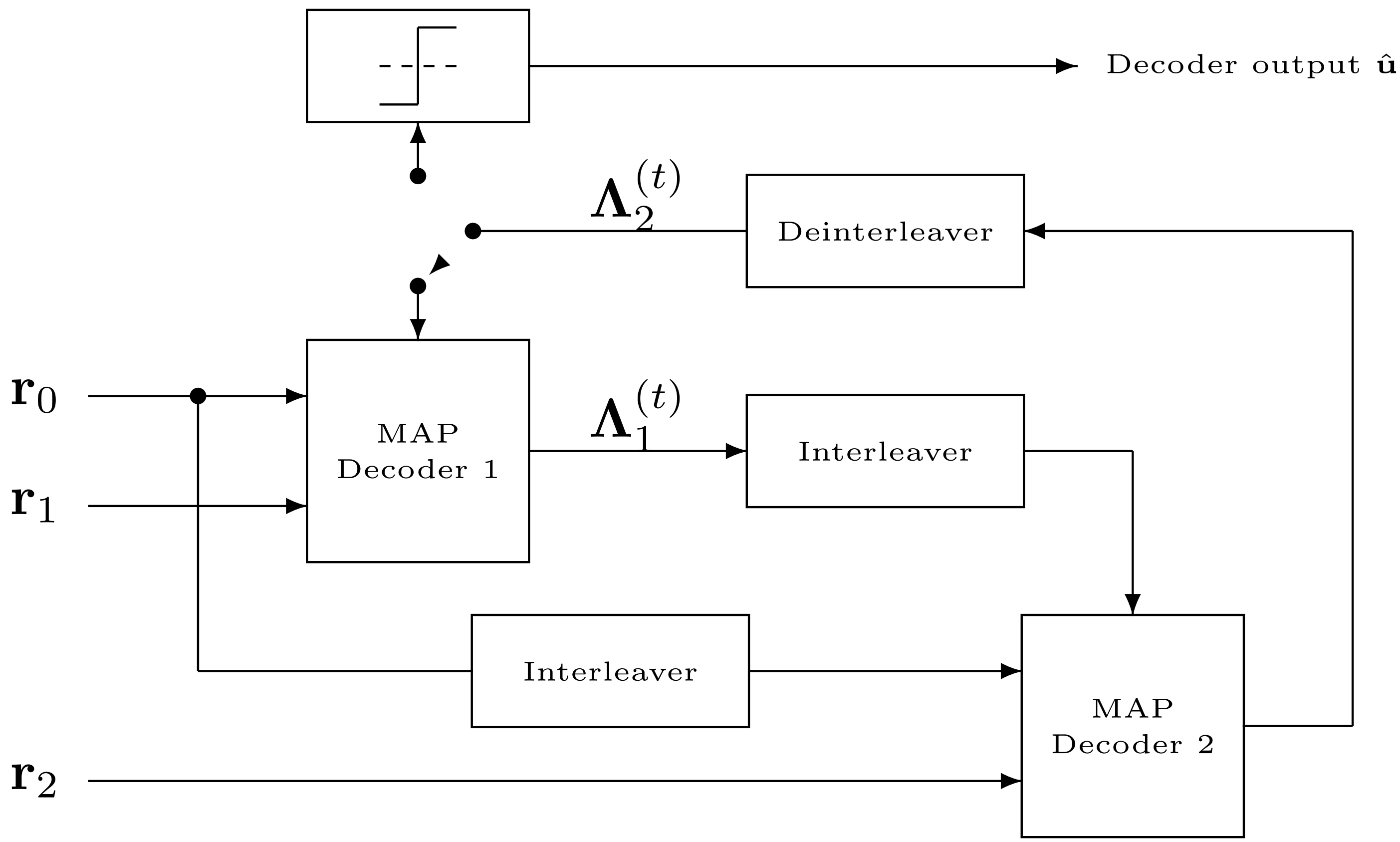
step 3: Soft decision. For $i = 0, \ldots, L-1$,

$$\Lambda(i) = \log \frac{\displaystyle\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) \sum_{u=0}^{1} \gamma(u, S_s^i, S_{\bar{s}}^{i+1})}{\displaystyle\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) \sum_{u=0}^{1} \gamma(u, S_s^i, S_{\bar{s}}^{i+1})}.$$

Turbo Encoder



Iterative decoder



MAP algorithm revisited

1.
$$\Lambda(i) = \log \frac{\Pr\{u_i = 1|\mathbf{r}\}}{\Pr\{u_i = 0|\mathbf{r}\}}$$
does not (exactly) provide the information of $\Pr\{u_i = 1\} = 1 - \Pr\{u_i = 0\}$, which is the *information* requires for each component decoder of the iterative decoder.

2. Recall for $(3,1)$ code (i.e., $n = 3$):

$$\begin{aligned}
\gamma(u, S_s^i, S_{\bar{s}}^{i+1}) &= \begin{cases} \frac{p_i(u)}{\sqrt{2\pi}\sigma} e^{-\frac{\sum_{j=3i}^{3(i+1)-1}(r_j - x_j(u))^2}{2\sigma^2}}, & (S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(u)}; \\ 0, & \text{otherwise}, \end{cases} \\
&= \begin{cases} \frac{p_i(u)}{\sqrt{2\pi}\sigma} e^{-\frac{\sum_{j=in}^{(i+1)n-1}(r_j - x_j(u))^2}{2\sigma^2}}, & (S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(u)}; \\ 0, & \text{otherwise}, \end{cases}
\end{aligned}$$

where, for convenience, denote $\Pr\{u_i = u\}$ by $p_i(u)$.

MAP algorithm revisited

$$
\begin{aligned}
\Lambda(i) &= \log \frac{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) \sum_{u=0}^{1} \gamma(u, S_s^i, S_{\bar{s}}^{i+1})}{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) \sum_{u=0}^{1} \gamma(u, S_s^i, S_{\bar{s}}^{i+1})} \\
&= \log \frac{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) p_i(1) \exp\left\{-\frac{\sum_{j=3i}^{3(i+1)-1}(r_j - x_j(1))^2}{2\sigma^2}\right\}}{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) p_i(0) \exp\left\{-\frac{\sum_{j=3i}^{3(i+1)-1}(r_j - x_j(0))^2}{2\sigma^2}\right\}}.
\end{aligned}
$$

$$= \log \frac{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) p_i(1) \exp\left\{ -\frac{(r_{3i} - x_{3i}(1))^2 + \sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(1))^2}{2\sigma^2} \right\}}{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) p_i(0) \exp\left\{ -\frac{(r_{3i} - x_{3i}(0))^2 + \sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(0))^2}{2\sigma^2} \right\}}$$

$$= \log \frac{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) p_i(1) \exp\left\{ -\frac{(r_{3i} + (-1)^1)^2 + \sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(1))^2}{2\sigma^2} \right\}}{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) p_i(0) \exp\left\{ -\frac{(r_{3i} + (-1)^0)^2 + \sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(0))^2}{2\sigma^2} \right\}}$$

$$= \frac{2}{\sigma^2} r_{3i} + \log \frac{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) p_i(1) \exp\left\{ -\frac{\sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(1))^2}{2\sigma^2} \right\}}{\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) p_i(0) \exp\left\{ -\frac{\sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(0))^2}{2\sigma^2} \right\}}$$

$$
\begin{aligned}
&= \log\frac{p_i(1)}{p_i(0)} + \frac{2}{\sigma^2} r_{3i} \\
&\quad + \log \frac{\displaystyle\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) \exp\left\{ -\frac{\sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(1))^2}{2\sigma^2} \right\}}{\displaystyle\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) \exp\left\{ -\frac{\sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(0))^2}{2\sigma^2} \right\}} \\
&= \log\frac{p_i(1)}{p_i(0)} + \frac{2}{\sigma^2} r_{3i} + \Lambda_e(i),
\end{aligned}
$$

where

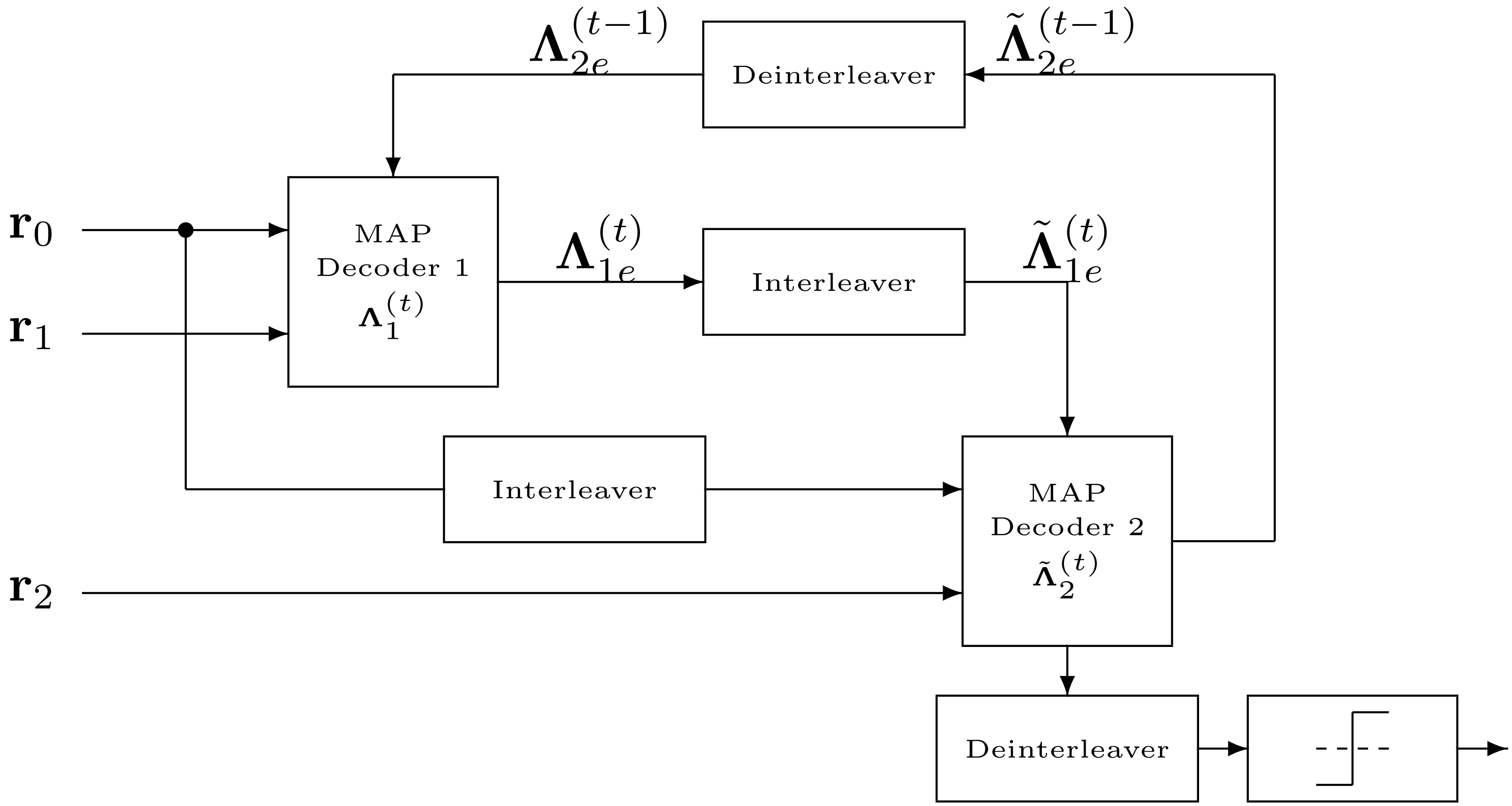
$$
\Lambda_e(i) \triangleq \log \frac{\displaystyle\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(1)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) \exp\left\{ -\frac{\sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(1))^2}{2\sigma^2} \right\}}{\displaystyle\sum_{(S_s^i, S_{\bar{s}}^{i+1}) \in \mathcal{B}_i^{(0)}} \alpha(S_s^i)\beta(S_{\bar{s}}^{i+1}) \exp\left\{ -\frac{\sum_{j=3i+1}^{3(i+1)-1} (r_j - x_j(0))^2}{2\sigma^2} \right\}}.
$$

1. $\Lambda_e(i)$ is called the *extrinsic information*, and is (together with $2r_{3i}/\sigma^2$) used to improve the *a priori probability estimate* $\log[p_i(1)/p_i(0)]$ for the next decoding stage.

As a consequence,

$$\log \frac{p_i(1)}{p_i(0)} = \Lambda(i) - \frac{2}{\sigma^2} r_{3i} - \Lambda_e(i).$$

Iterative MAP decoder



$$
\begin{aligned}
\mathbf{\Lambda}_1^{(t)} &= \mathbf{\Lambda}_{1e}^{(t)} + \frac{2}{\sigma^2}\mathbf{r}_1 + \mathbf{\Lambda}_{2e}^{(t-1)} \\
\tilde{\mathbf{\Lambda}}_2^{(t)} &= \tilde{\mathbf{\Lambda}}_{2e}^{(t)} + \frac{2}{\sigma^2}\mathbf{r}_2 + \tilde{\mathbf{\Lambda}}_{1e}^{(t-1)}
\end{aligned}
$$

where $\tilde{\mathbf{\Lambda}}$ is the interleaved version of $\mathbf{\Lambda}$.

Iterative MAP decoder

step 1: $\mathbf{\Lambda}_{2e}^{(0)} = \mathbf{0}$.

step 2: For $t = 1, 2, \ldots, T$, where T is the total number of iterations:

- calculate $\mathbf{\Lambda}_1^{(t)}$ (based on the prior provided by $\mathbf{\Lambda}_{2e}^{(t-1)}$:

$$\Pr\{u_i = 1\} = \frac{e^{\Lambda_{2e}^{(t-1)}(i)}}{1 + e^{\Lambda_{2e}^{(t-1)}(i)}} \quad \text{and} \quad \Pr\{u_i = 0\} = \frac{1}{1 + e^{\Lambda_{2e}^{(t-1)}(i)}})$$

- output (by the first MAP decoder)

$$\mathbf{\Lambda}_{1e}^{(t)} = \mathbf{\Lambda}_1^{(t)} - \frac{2}{\sigma^2}\mathbf{r}_1 - \mathbf{\Lambda}_{2e}^{(t-1)}.$$

- calculate $\tilde{\mathbf{\Lambda}}_2^{(t)}$ (based on the prior provided by $\tilde{\mathbf{\Lambda}}_{1e}^{(t)}$)
- output (by the second MAP decoder)

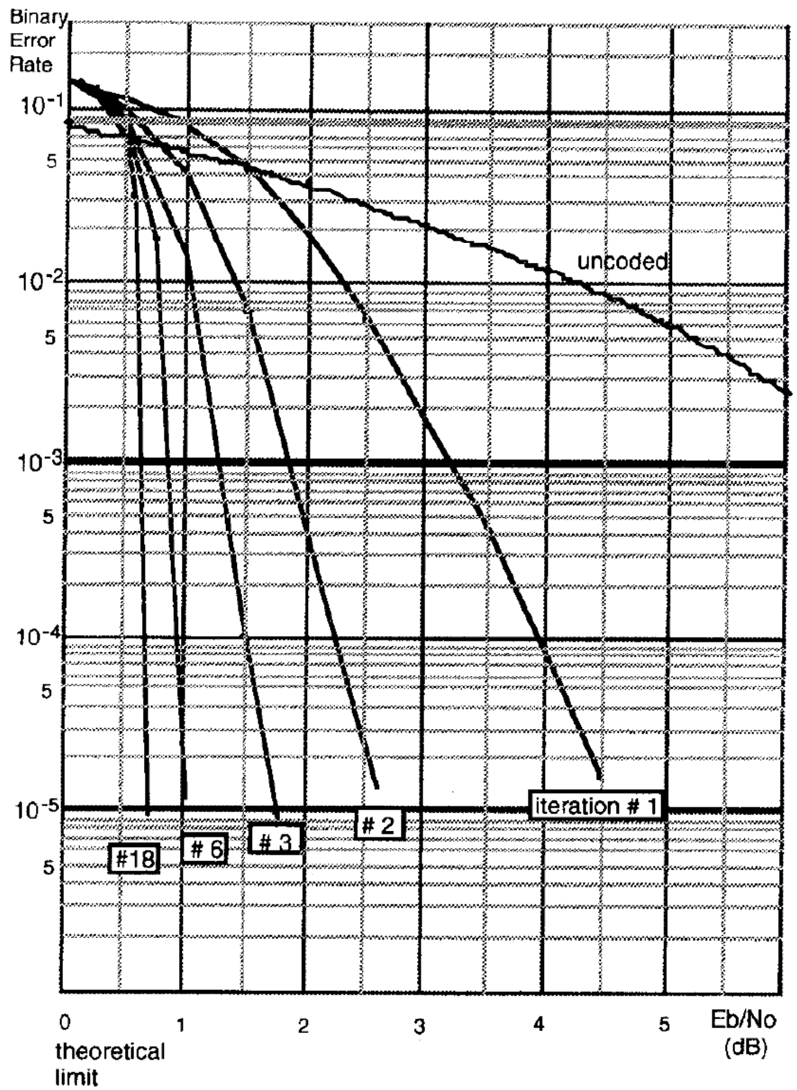
$$\tilde{\mathbf{\Lambda}}_{2e}^{(t)} = \mathbf{\Lambda}_2^{(t)} - \frac{2}{\sigma^2}\mathbf{r_1} - \mathbf{\Lambda}_{1e}^{(t)}.$$

step 3: Make the final (hard) decision on $\mathbf{u}$ based on $\boldsymbol{\Lambda}_2^{(T)}$.

Performance of iterative MAP decoder

1. The interleaver is the Berrou-Glavieux interleaver with size 256×256 ($k = 8$).
2. For 18 iterations, the E_b/N_0 is around 0.7 dB for BER=10^{-5}, which is around 0.5dB from the Shannon limit.

Performance of iterative MAP decoder



Some Important Codes

1. Algebraic codes– cyclic codes, Bose-Chaudhuri-Hocquenghem codes (BCH codes), Reed-Solomon codes (RS codes), Algebraic-Geometric codes (AG codes).
2. Codes for bandwidth-limited channels- Ungerboeck codes.
3. Codes approaching Shannon bound- turbo codes, low-density-parity-check codes (LDPC codes).
4. Codes for multi-input (multiple transmit antennas) multi-output (multiple receive antennas) channels- Space-time codes.