Turbo Decoding as an Instance of Expectation Maximization Algorithm

Saejoon Kim

Department of Computer Science
Sogang University
Seoul 121-742, Korea
saejoon@sogang.ac.kr

Abstract. The Baum-Welch algorithm is a technique for the maximum likelihood parameter estimation of probabilistic functions of Markov processes. We apply this technique to nonstationary Markov processes and explore a relationship between the Baum-Welch algorithm and the BCJR algorithm. Furthermore, we apply the Baum-Welch algorithm to two nonstationary Markov processes and obtain the turbo decoding algorithm.

1 Introduction

In 1962, Lloyd R. Welch discovered a heuristic algorithm for maximum likelihood parameter (re)estimation for probabilistic functions of Markov processes. In the late 1960's, the algorithm was published in a series of papers[2][3][4]. This algorithm has come to be called the *Baum-Welch algorithm*, and will be referred to as such in this paper.

In this paper, we apply the Baum-Welch algorithm to a class of nonstationary Markov processes and obtain the BCJR algorithm (an optimal symbol-by-symbol algorithm for the decoding of linear codes [1]) as a special case. We then apply the Baum-Welch algorithm to two nonstationary Markov processes which are related by the ordering of the sequence of inputs that cause the processes. For a given solution, the different ordering of the sequence of inputs causes a substantial increase in computational complexity. We show how one approach to overcoming this high complexity results in the turbo decoding algorithm [5][6][7].

2 Statistical Estimation of Probabilistic Functions of Markov Process

We consider a stationary Markov process $\{S_t\}$ which is generated by some $S \times S$ stochastic matrix $A = \{a(s', s)\}$, $0 \leq s', s \leq S-1$, with initial probability distribution $\{a(s)\}$. The process produces outputs $\{y_t\}$ with probability distribution $f(s', s)(y_t)$. $f(s', s)(y_t) = \Pr\{y_t | S_{t-1} = s', S_t = s\}$ is the output probability distribution for state transition $s' \to s$ and thus $\int f(s', s)(\cdot) d\mu(\cdot) = 1$. Three terms define the parameter, $\theta \in \Theta$, of the Markov process: $\theta = \{\{a(s)\}, \{a(s', s)\}, \{f(s', s)(\cdot)\}\}$.

I. King et al. (Eds.): ICONIP 2006, Part I, LNCS 4232, pp. 371–378, 2006.

Now, given θ, we shall define by $Y = \{Y_t\}$, a stochastic process, with the following density function

$$\Pr\{Y_1 = y_1, \cdots, Y_\tau = y_\tau\}(\theta) = \sum_{s_0, s_1, \cdots, s_\tau = 0}^{S-1} a(s_0) \prod_{t=1}^{\tau} a(s_{t-1}, s_t) f(s_{t-1}, s_t)(y_t) \quad (1)$$

This is a *probabilistic function of the Markov process* $\{S_t\}$ *given* θ. $\{S_t\}$ being stationary, $\{Y_t\}$ is also stationary. Note how the output probabilities $\{f(s_{t-1}, s_t)(y_t)\}$ "hide" the Markov process $\{S_t\}$. A probabilistic function of a Markov process is often called a *hidden Markov model*.

Let us denote the output sequence extending from time t_1 to t_2 by $\mathbf{y}_{t_1}^{t_2}$. The Baum-Welch algorithm is an iterative method for maximum likelihood estimation of the parameter θ for $\Pr\{\mathbf{y}_1^\tau\}(\theta)$. In other words, Baum-Welch algorithm finds θ' such that

$$\Pr\{\mathbf{y}_1^\tau\}(\theta') = \max_\theta \Pr\{\mathbf{y}_1^\tau\}(\theta) \quad (2)$$

We assume that for a fixed $\mathbf{y}_1^\tau$, $\Pr\{\mathbf{y}_1^\tau\}(\theta)$ is a smooth function of θ. We shall abbreviate this by $P(\theta)$ when it is clear from the context. Under a mild hypothesis on f, we can define a continuous transformation $\mathcal{T} : \Theta \to \Theta$ such that

$$P(\mathcal{T}(\theta)) \geq P(\theta) \quad (3)$$

with equality if and only if $\mathcal{T}(\theta) = \theta$. Now, let $P(\theta^{(l)}) = \sum_{\mathbf{s} \in \mathbf{S}} p(\mathbf{s}, \theta^{(l)})$, where

$$p(\mathbf{s}, \theta^{(l)}) = a_0^{(l)}(s_0) \prod_{t=1}^{\tau} a^{(l)}(s_{t-1}, s_t) f^{(l)}(s_{t-1}, s_t)(y_t).$$

Here we have let $\mathbf{S} = \{0, 1, \cdots, S-1\}^\tau$ and $\mathbf{s} = \{s_0, s_1, \cdots, s_\tau\} \in \mathbf{S}$. Moreover, let us define an auxiliary function $Q(\theta, \theta')$ such that $Q(\theta, \theta') = \sum_{\mathbf{s} \in \mathbf{S}} p(\mathbf{s}, \theta) \log p(\mathbf{s}, \theta')$. Then, it is proved in [4] that if $Q(\theta, \theta') \geq Q(\theta, \theta)$ then $P(\theta') \geq P(\theta)$ with equality if and only if $p(\mathbf{s}, \theta') = p(\mathbf{s}, \theta)$. This leads us to define

$$\mathcal{T}(\theta) = \{\theta' \in \Theta | Q(\theta, \theta') = \max_{\bar{\theta} \in \Theta} Q(\theta, \bar{\theta})\} \quad (4)$$

To summarize, Baum-Welch algorithm finds the transformation $\mathcal{T}(\cdot)$ such that given any initial distribution $\theta^{(0)}$,

$$\lim_{l \to \infty} \Pr\{\mathbf{y}_1^\tau\}(\mathcal{T}^l(\theta^{(0)})) = \max_\theta \Pr\{\mathbf{y}_1^\tau\}(\theta)$$

where $\mathcal{T}^l(\cdot) = \mathcal{T}(\mathcal{T}^{l-1}(\cdot))$.

Let us now find this $\mathcal{T}(\cdot)$ more explicitly. For notational clarity, let $\mathcal{T}^l(\theta^{(0)}) = \theta^{(l)}$, where $\theta^{(l)} = \{\{a^{(l)}(s)\}, \{a^{(l)}(s', s)\}, \{f^{(l)}(s', s)(\cdot)\}\}$. Now, suppose we already know $\{f(s', s)(\cdot)\}$. From here on, we shall assume that this is true and thus suppress its superscript (l). Let

$$\begin{aligned}
\gamma_t^{(l)}(s', s) &= \Pr\{S_t = s, y_t | S_{t-1} = s'\}(\theta^{(l)}) \\
\alpha_t^{(l)}(s) &= \Pr\{S_t = s, \mathbf{y}_1^t\}(\theta^{(l)}) \\
\beta_t^{(l)}(s) &= \Pr\{\mathbf{y}_{t+1}^\tau | S_t = s\}(\theta^{(l)})
\end{aligned}$$

where $\gamma_t^{(l)}(s',s) = a^{(l)}(s',s)f^{(l)}(s',s)(y_t)$, and $\alpha_t^{(l)}(s)$ and $\beta_t^{(l)}(s)$ are computed by the forward-backward (BCJR) algorithm [1]. Let

$$\begin{aligned} \lambda_t^{(l)}(s) &= \Pr\{S_t = s, \mathbf{y}_1^\tau\}(\theta^{(l)}) \\ &= \alpha_t^{(l)}(s)\beta_t^{(l)}(s) \\ \sigma_t^{(l)}(s',s) &= \Pr\{S_{t-1} = s', S_t = s, \mathbf{y}_1^\tau\}(\theta^{(l)}) \\ &= \alpha_{t-1}^{(l)}(s')\gamma_t^{(l)}(s',s)\beta_t^{(l)}(s) \end{aligned} \tag{5}$$

The Baum-Welch (re)estimate algorithm determines the following.

$$a^{(l)}(s) = \frac{\lambda_0^{(l-1)}(s)}{\Pr\{\mathbf{y}_1^\tau\}(\theta^{(l-1)})} \tag{6}$$

$$a^{(l)}(s',s) = \frac{\sum_{t=1}^{\tau} \sigma_t^{(l-1)}(s',s)}{\sum_{t=0}^{\tau-1} \lambda_t^{(l-1)}(s')} \tag{7}$$

In essence, Baum-Welch (re)estimate is the *a posteriori* (re)estimate of the parameter given the *a priori* parameter and the output, *i.e.*, $\Pr\{\cdot\}(\theta^{(l)}) = \Pr\{\cdot|\mathbf{y}_1^\tau\}(\theta^{(l-1)})$.

Let us consider now the nonstationary Markov process case. Let a nonstationary Markov process $\{S_t\}$ be generated by a series of $S \times S$ stochastic matrices $\{A_t\}$ where each stochastic matrix $A_t = \{a_t(s',s)\}$ with initial distribution $\{a_t(s)\}$ generates some stationary Markov process. Thus $\theta = \{\theta_t\}$ and $\theta_t = \{\{a_t(s)\},\{a_t(s',s)\},\{f_t(s',s)(\cdot)\}\}$. For the nonstationary case, it makes sense to consider

$$a_t^{(l)}(s) = \frac{\lambda_t^{(l-1)}(s)}{\Pr\{\mathbf{y}_1^\tau\}(\theta^{(l-1)})} \tag{8}$$

$$a_t^{(l)}(s',s) = \frac{\sigma_t^{(l-1)}(s',s)}{\lambda_{t-1}^{(l-1)}(s')} \tag{9}$$

so that $\Pr\{\cdot\}(\theta^{(l)}) = \Pr\{\cdot|\mathbf{y}_1^\tau\}(\theta^{(l-1)})$. Note that we only need to find $\{a_0^{(l)}(s)\}$ and $\{a_t^{(l)}(s',s)\}$ since $a_t^{(l)}(s) = \sum_{s'} a_{t-1}^{(l)}(s')a_t^{(l)}(s',s)$. This statistical estimation of probabilistic function of nonstationary Markov process includes the optimal symbol-by-symbol decoding of linear codes. Let us be more precise. The symbol-by-symbol MAP algorithm[1] finds $\Pr\{U_t = u_t|\mathbf{y}_1^\tau\}(\theta^{(0)}) = \Pr\{U_t = u_t\}(\theta^{(1)})$ where

$$\begin{aligned} \Pr\{U_t = a|\mathbf{y}_1^\tau\}(\theta^{(0)}) &= \alpha \textstyle\sum_{s'\to s: U_t = a} \sigma_t^{(0)}(s',s) \\ &= \textstyle\sum_{s'\to s: U_t = a} a_{t-1}^{(1)}(s')a_t^{(1)}(s',s) \end{aligned}$$

with the boundary condition that $a_0^{(0)}(0) = 1$ and $a_0^{(0)}(s) = 0,\ \forall s \neq 0$, and the remaining $\theta^{(0)}$ is uniformly distributed. Thus, it remains to find $\{a_t^{(1)}(s',s)\}$.

The sequence of random variables $\{U_t\}$ corresponds to the sequence of stochastic matrices $\{A_t\}$ and thus generates a Markov process. Assuming a Gaussian noise with variance σ^2, we have $f_t(s', s)(y_t) = C \exp \frac{(y_t - a)^2}{2\sigma^2}$ if $s' \to s$ causes the coded bit $x_t = a$. Now, for any $a_t^{(l)}(s', s)$, $\sum_s a_t^{(l)}(s', s) = 1$ where all but 2 terms are equal to 0 for all l. Let us denote these two nonzero terms by ρ and $1 - \rho$. Then, for all l

$$\arg \max_{0 \le \rho \le 1} \frac{\partial \Pr\{\mathbf{y}_1^{\mathcal{T}}\}(\theta^{(l)})}{\partial \rho}$$

is either $\rho = 0$ or $\rho = 1$. Thus the problem of finding the maximum likelihood parameter (re)estimation of nonstationary Markov process has reduced to that of finding $\{a_t(s', s)\}$ where $a_t(s', s) = 0$ or 1 for all t, s' and s. ¿From the monotonicity of $\Pr\{y_t\}(\mathcal{T}^l(\cdot))$ of Equation (3) and the uniform distribution of $\theta^{(0)}$,

$$a_t^{(1)}(s', s) > \frac{1}{2} \Rightarrow \; a'_t(s', s) = 1$$

and likewise for the $a_t^{(1)}(s', s) < \frac{1}{2}$ case for all t, s' and s. This result and the boundary condition gives $a'_t(s) = 0$ or 1 for all t and s. Thus, in decoding of linear codes problem, $\theta^{(0)}$ being the *correct a priori* knowledge, the *a posteriori* estimate of $\theta^{(0)}$ given the observation $\mathbf{y}_1^{\mathcal{T}}$, $\theta^{(1)}$, is the maximum likelihood value, θ', after the hard decision.

3 On Statistical Estimation of Two Probabilistic Functions of Markov Processes

In this section, we consider two probabilistic functions of Markov processes. Let $Y = \{Y_t\}$ be a probabilistic function of an S state Markov process $\{S_t\}$ given θ and let $\tilde{Y} = \{\tilde{Y}_t\}$ be a probabilistic function of an S state Markov process $\{\tilde{S}_t\}$ given $\tilde{\theta}$. Let a sequence of random variables $\{U_t\}$ define the S state Markov process $\{S_t\}$. Similarly, let a reordered version of the sequence of random variables $\mathcal{P}(\{u_t\}) = \{u_{t'}\}$ define the S state Markov process $\{\tilde{S}_t\}$. The relationship is shown in Fig. 1.

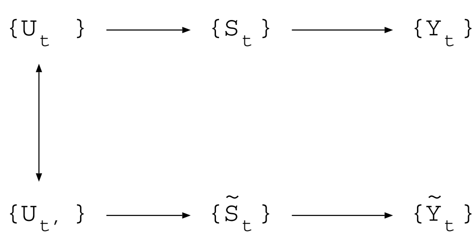

Fig. 1.

The sequence of hidden random variables $\{U_t\}$ is observed through the outputs of Markov processes $\{S_t\}$ and $\{\tilde{S}_t\}$. Our problem is to estimate the sequence of hidden variables $\{u_t\}$ given the outputs $\{y_t\}$ and $\{\tilde{y}_t\}$. To this end, let us proceed as follows. For $l \geq 0$, let

$$\Pr\{U_t = a|\mathbf{y}_1^\tau, \tilde{\mathbf{y}}_1^\tau\}(\theta^{(2l)}) = \alpha \sum_{s' \to s : U_t = a} \sigma_t^{(l)}(s', s) \tag{10}$$

$$\Pr\{U_{t'} = a|\mathbf{y}_1^\tau, \tilde{\mathbf{y}}_1^\tau\}(\theta^{(2l+1)}) = \alpha \sum_{s' \to s : U_t = a} \tilde{\sigma}_t^{(l)}(s', s) \tag{11}$$

where $\sigma_t^{(l)}(s', s) = \alpha_{t-1}^{(l)}(s') a_t^{(l)}(s', s) f_t(s', s)(y_t) \beta_t^{(l)}(s)$ and similarly for $\tilde{\sigma}_t^{(l)}(s', s)$. Thus we need to find $\{a_t^{(l)}(s', s)\}$ and $\{\tilde{a}_t^{(l)}(s', s)\}$.

Now, if $\{u_t\} = \{u_{t'}\}$, then

$$a_t^{(l)}(s', s) = \frac{\tilde{\sigma}_t^{(l-1)}(s', s)}{\tilde{\lambda}_{t-1}^{(l-1)}(s')} \text{ and } \tilde{a}_t^{(l)}(s', s) = \frac{\sigma_t^{(l)}(s', s)}{\lambda_{t-1}^{(l)}(s')} \tag{12}$$

Due to the reordering, the computational complexity of $a_t^{(l)}(s', s)$ and $\tilde{a}_t^{(l)}(s', s)$ is high. A suboptimal way to estimate $\{a_t^{(l)}(s', s)\}$ and $\{\tilde{a}_t^{(l)}(s', s)\}$ is via the following.

$$\begin{aligned} &\text{For all } s' \to s \text{ that corresponds to } U_t = a, \\ &\quad \text{let } a_t^{(l)}(s', s) = \mathcal{P}^{-1}\big(\textstyle\sum_{s' \to s : a} \tilde{\sigma}_t^{(l-1)}(s', s)\big) \\ &\text{and similarly,} \\ &\text{for all } s' \to s \text{ that corresponds to } U_{t'} = a, \\ &\quad \text{let } \tilde{a}_t^{(l)}(s', s) = \mathcal{P}\big(\textstyle\sum_{s' \to s : a} \sigma_t^{(l)}(s', s)\big). \end{aligned} \tag{13}$$

Note that this process may violate Equation (4) and thus (3). Thus this deviation from the Baum-Welch algorithm does not guarantee that the algorithm will find the maximum likelihood parameter. We shall say more about this point later in the paper. Also note that $\sum_{s' \to s : a} \tilde{\sigma}_t^{(l-1)}(s', s)$ is the *weighted* average, and $\sum_{s' \to s : a} \frac{\tilde{\sigma}_t^{(l-1)}(s', s)}{\tilde{\lambda}_{t-1}^{(l-1)}(s')}$ is just the average of $\Pr\{\tilde{S}_t = s|\tilde{S}_{t-1} = s', \tilde{\mathbf{y}}_1^\tau\}(\theta^{(2l-1)})$, respectively, and thus $a_t^{(l)}(s', s) = \mathcal{P}^{-1}(\sum_{s' \to s : a} \tilde{\sigma}_t^{(l-1)}(s', s))$ is a better estimate than

$$a_t^{(l)}(s', s) = \mathcal{P}^{-1}\Big(\sum_{s' \to s : U_t = a} \frac{\tilde{\sigma}_t^{(l-1)}(s', s)}{\tilde{\lambda}_{t-1}^{(l-1)}(s')}\Big).$$

Now for $l \geq 1$

$$\begin{aligned} \sigma_t^{(l)}(s', s) &= \alpha_{t-1}^{(l)}(s') \gamma_t^{(l)}(s', s) \beta_t^{(l)}(s) \\ &= \alpha_{t-1}^{(l)}(s') a_t^{(l)}(s', s) f_t(s', s)(y_t) \beta_t^{(l)}(s) \\ &= \alpha_{t-1}^{(l)}(s') \mathcal{P}^{-1}\Big(\sum_{s' \to s : a} \tilde{\sigma}_t^{(l-1)}(s', s)\Big) f_t(s', s)(y_t) \beta_t^{(l)}(s). \end{aligned}$$

And similarly for $l \geq 0$,

$$\tilde{\sigma}_t^{(l)}(s', s) = \tilde{\alpha}_{t-1}^{(l)}(s')\mathcal{P}\Big(\sum_{s' \to s: U_t = a} \sigma_t^{(l)}(s', s)\Big)\tilde{f}_t(s', s)(\tilde{y}_t)\tilde{\beta}_t^{(l)}(s).$$

Thus Equation (10) is now

$$\begin{aligned}&\Pr\{U_t = a|\mathbf{y}_1^\tau, \tilde{\mathbf{y}}_1^\tau\}(\theta^{(2l)}) \\ &= \mathcal{P}^{-1}\Big(\sum_{s' \to s: U_t = a} \tilde{\sigma}_t^{(l-1)}(s', s)\Big) \sum_{s' \to s: U_t = a} \alpha_{t-1}^{(l)}(s') f_t(s', s)(y_t)\beta_t^{(l)}(s)\end{aligned} \tag{14}$$

and Equation (11) is

$$\begin{aligned}&\Pr\{U_{t'} = a|\mathbf{y}_1^\tau, \tilde{\mathbf{y}}_1^\tau\}(\theta^{(2l+1)}) \\ &= \mathcal{P}\Big(\sum_{s' \to s: U_t = a} \sigma_t^{(l)}(s', s)\Big) \sum_{s' \to s: U_t = a} \tilde{\alpha}_{t-1}^{(l)}(s') \tilde{f}_t(s', s)(\tilde{y}_t)\tilde{\beta}_t^{(l)}(s).\end{aligned} \tag{15}$$

Thus, the *a posteriori* estimate of $\Pr\{U_t = a\}$ for a given iteration step becomes the *a priori* value of $\Pr\{U_t = a\}$ for the next iteration step.

Now, for a special class of two probabilistic functions of Markov processes, this process deserves a second look. Let $\{y_t\} = \{y_{1t}, y_{2t}\}$ where $\{y_{1t}\}$ and $\{y_{2t}\}$ are mutually independent given $\{u_t\}$. Repeat for $\{\tilde{y}_t\}$. We can factor $f_t(s', s)(y_t)$ into $f_{1t}(s', s)(y_{1t}) f_{2t}(s', s)(y_{2t})$. Moreover, let $\{y_{1t}\}$ be independent of the state transition given $\{u_t\}$ and thus let $f_{1t}(s', s)(y_{1t}) = f_{1t}(u_t)(y_{1t})$.

Now for all $s' \to s$ that corresponds to $U_t = a$, let

$$\begin{aligned} a_t^{(l)}(s', s) &= \mathcal{P}^{-1}\Big(\frac{\sum_{s' \to s: U_t = a} \tilde{\sigma}_t^{(l-1)}(s', s)}{\tilde{a}_t^{(l-1)}(s', s)\tilde{f}_{1t}(a)(\tilde{y}_{1t})}\Big) \\ &= \mathcal{P}^{-1}\Big(\sum_{s' \to s: U_t = a} \tilde{\alpha}_{t-1}^{(l-1)}(s')\tilde{f}_{2t}(s', s)(\tilde{y}_{2t})\tilde{\beta}_t^{(l-1)}(s)\Big)\end{aligned} \tag{16}$$

and similarly

$$\begin{aligned} \tilde{a}_t^{(l)}(s', s) &= \mathcal{P}\Big(\frac{\sum_{s' \to s: U_t = a} \sigma_t^{(l)}(s', s)}{a_t^{(l)}(s', s) f_{1t}(a)(y_{1t})}\Big) \\ &= \mathcal{P}\Big(\sum_{s' \to s: U_t = a} \alpha_{t-1}^{(l)}(s') f_{2t}(s', s)(y_{2t})\beta_t^{(l)}(s)\Big).\end{aligned} \tag{17}$$

Then,

$$\begin{aligned}&\Pr\{U_t = a|\mathbf{y}_1^\tau, \tilde{\mathbf{y}}_1^\tau\}(\theta^{(2l)}) \\ &\quad= \mathcal{P}^{-1}\Big(\sum_{s' \to s: U_t = a} \frac{\tilde{\sigma}_t^{(l-1)}(s', s)}{\tilde{a}_t^{(l-1)}(s', s)\tilde{f}_{1t}(s', s)(\tilde{y}_{1t})}\Big) \\ &\qquad \cdot f_{1t}(a)(y_{1t}) \cdot \sum_{s' \to s: U_t = a} \alpha_{t-1}^{(l)}(s') f_{2t}(s', s)(y_{2t})\beta_t^{(l)}(s)\end{aligned} \tag{18}$$

and

$$\begin{aligned}&\Pr\{U_{t'} = a|\mathbf{y}_1^\tau, \tilde{\mathbf{y}}_1^\tau\}(\theta^{(2l+1)}) \\ &\quad= \mathcal{P}\Big(\sum_{s' \to s: U_t = a} \frac{\sigma_t^{(l)}(s', s)}{a_t^{(l)}(s', s) f_{1t}(s', s)(y_{1t})}\Big) \\ &\qquad \cdot \tilde{f}_{1t}(a)(\tilde{y}_{1t}) \cdot \sum_{s' \to s: U_t = a} \tilde{\alpha}_{t-1}^{(l)}(s')\tilde{f}_{2t}(s', s)(\tilde{y}_{2t})\tilde{\beta}_t^{(l)}(s).\end{aligned} \tag{19}$$

Equations (18) and (19) are the turbo decoding algorithm [5][6][7].

4 Results

For completeness, let us define $a_t^{(l)}(s', s)$ in two other possible ways for this special class of two probabilistic functions of Markov processes. Namely for all $s' \to s$ that corresponds to $U_t = a$, let

$$a_t^{(l)}(s', s) = \mathcal{P}^{-1}\left(\frac{\sum_{s' \to s: U_t = a} \tilde{\sigma}_t^{(l-1)}(s', s)}{\tilde{a}_t^{(l-1)}(s', s)}\right) \tag{20}$$

and similarly for $\tilde{a}_t^{(l)}(s', s)$.

Or, let

$$a_t^{(l)}(s', s) = \mathcal{P}^{-1}\left(\frac{\sum_{s' \to s: U_t = a} \tilde{\sigma}_t^{(l-1)}(s', s)}{\tilde{f}_{1t}(a)(\tilde{y}_{1t})}\right) \tag{21}$$

and similarly for $\tilde{a}_t^{(l)}(s', s)$.

We ran simulations of iterative decoders based on the four different (re)estimates of $\{a_t^{(l)}(s', s)\}$ (Equations (13), (16 and 17), (20), and (21)). We used a rate-$\frac{1}{2}$ parallel concatenated $(37, 21)$ RSC encoder with a nonuniform 63×31 block interleaver. Figure 2 shows the performance of the four iterative decoding methods after 0, 2, and 8 iterations. Curves connected by $*$ correspond to the iterative decoding via Equation (13). Curves connected by $\triangle$ and by $\circ$ correspond to the iterative decodings via Equations (20) and (21), respectively. Performance for turbo decoding algorithm is represented by pure solid curves. Results show that the turbo decoding algorithm outperforms all other known

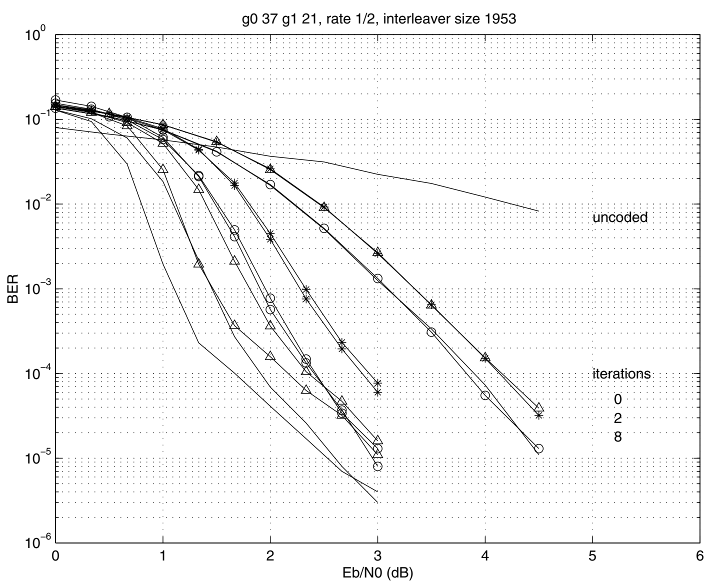

Fig. 2.

iterative decoding schemes of the same order of complexity for an RSC-type encoders. Moreover, the results show that getting rid of the *old a priori* value has more weight than getting rid of the *systematic* value.

References

[1] L.R. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284-287, Mar. 1974.
[2] L.E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *Ann. Math. Stat.* **37** (1966), pp. 1554-1563.
[3] L.E. Baum and J.A. Eagon, "An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology," *Bull. Amer. Math. Soc.* **73** (1967), pp. 360-363.
[4] L.E. Baum, T. Petrie, G. Soules and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Stat.* **41** (1970), pp. 164-171.
[5] C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes(1)," in *Proc., IEEE Int. Conf. on Communications* (Geneva, Switzerland, May 1993), pp. 1064-1070.
[6] S. Kim and S.B. Wicker, "A Connection between the Baum-Welch Algorithm and Turbo Decoding," in *Proc. 1998 IEEE Inform. Theory Workshop*, Killarney, Ireland, June 22-26, 1998. (Invited)
[7] S. Kim and S.B. Wicker, "An Exploration of the Relationship between the Baum-Welch Algorithm and Turbo Decoding," in *Proc. 1998 IEEE Int. Symp. on Inform. Theory*, Boston, USA, August 16-21, 1998.