

Equalização Cega de Canais Lineares FIR empregando o Algoritmo EM Bayesiano

Claudio J. Bordin Jr. e Luiz A. Baccalá

Resumo—Neste trabalho descrevemos métodos para equalizar cegamente canais de comunicação lineares FIR através da versão Bayesiana do consagrado algoritmo EM (*Expectation-Maximization*). Os métodos desenvolvidos fornecem estimativas que maximizam localmente a densidade de probabilidade *a posteriori* dos parâmetros do canal de comunicação, estimando os símbolos transmitidos através do algoritmo BCJR. Para contornar dificuldades de convergência observadas sob baixos níveis de ruído, modificamos a proposta original ao considerar técnicas de *annealing*, que proporcionam um desempenho próximo ao ótimo (Viterbi) sem incorrer em aumento de complexidade computacional, conforme atestam simulações numéricas.

Palavras-Chave—Equalização cega, Algoritmo *Expectation-Maximization* Bayesiano, *Annealing*.

Abstract—This work describes blind equalization methods for linear FIR channels using a Bayesian version of the well-known EM (*Expectation-Maximization*) algorithm. The proposed methods produce estimates that locally maximize the posterior density of the channel parameters, leading to estimates of the transmitted bits via the BCJR algorithm. Without increasing the overall computational complexity, we next propose an enhanced scheme based on annealing techniques to circumvent bad convergence properties observed in the original algorithm when the noise level is low, leading to performances close to optimal (Viterbi for known channel parameters) as confirmed by numerical simulations.

Keywords—Blind Equalization, Bayesian *Expectation-Maximization* Algorithm, *Annealing*.

I. INTRODUÇÃO

A despeito dos avanços observados nas últimas décadas, a equalização cega [1] [2] de canais de comunicação ainda carece de soluções simultaneamente satisfatórias nos quesitos robustez, velocidade de convergência e complexidade computacional. Neste contexto, o algoritmo EM [3] oferece uma solução de compromisso para o problema da equalização cega ao prover estimativas aproximadamente ótimas para os parâmetros do canal de transmissão, embora a uma complexidade computacional relativamente elevada.

Introduzido em 1977 por Dempster *et al.* [4], o algoritmo EM é um método iterativo que aproxima soluções ótimas (nos sentidos ML ou MAP [5]) para problemas de estimação com dados latentes, ou seja, problemas nos quais a função de verossimilhança (a densidade *a posteriori*) somente é tratável se condicionada a um conjunto de dados não-observados.

Apesar de terem-se desenvolvido diversos métodos de obtenção de soluções ótimas no sentido da máxima verossimilhança (ML) para o problema da equalização cega utilizando o algoritmo EM ([6][7]), neste trabalho propomos

um novo método, desta vez de cunho Bayesiano. Nele obtêm-se aproximações para a estimativa MAP dos parâmetros do canal admitindo-se para este uma densidade *a priori* gaussiana. Em consequência à análise dos resultados obtidos com esta abordagem, propomos um segundo método para equalização cega empregando uma técnica de *annealing* que corrige deficiências de desempenho observadas quando a relação sinal-ruído disponível é alta.

O conteúdo deste trabalho está organizado da seguinte forma: na Sec. II apresentamos o algoritmo EM, descrevendo em seguida o modelo de sinal empregado (Sec. III). Na Sec. IV, descrevemos a aplicação do algoritmo EM à solução do problema da equalização cega, avaliando seu desempenho através de simulações numéricas. Na Sec. V apresentamos o algoritmo modificado e seu desempenho, conduzindo às conclusões formalizadas na Sec. VI.

II. O ALGORITMO EM

Dado que se deseje inferir um vetor aleatório θ a partir de um conjunto de observações A e considerando uma variável aleatória B latente (i.e. não-observada), de sorte que se possa descrever θ por meio de uma $p(\theta|A, B)$ analiticamente simples, o algoritmo EM prescreve como obter uma seqüência de estimativas $\theta^{(i)}$, $i \geq 0$, tais que $p(\theta^{(i+1)}|A) \geq p(\theta^{(i)}|A)$. Em outras palavras, o algoritmo EM permite obter valores $\theta^{(i)}$ que maximizam localmente a densidade *a posteriori* da variável inferida (veja o Apêndice para detalhes).

Dividido basicamente em dois passos, o algoritmo EM consiste primeiramente (passo E) em determinar a esperança da densidade *a posteriori* aumentada $p(\theta|A, B)$ dadas as observações A e uma estimativa atual $\theta^{(i)}$. A isto segue-se (passo M) o cálculo da nova estimativa de $\theta^{(i+1)}$, obtida ao maximizar-se a esperança determinada no passo E em função de θ , conforme descrito na Tabela I.

Como se pode observar na Tabela I, o algoritmo EM é um procedimento bastante geral. Como tal, tem aplicabilidade a diversos problemas de estimação, sendo sua praticidade determinada pela capacidade de se calcular e maximizar $Q(\theta, \theta^{(i)})$ de modo fácil. Cabe salientar que as estimativas obtidas não são necessariamente boas aproximações da estimativa MAP do vetor aleatório θ , já que a convergência do algoritmo só é garantida até máximos locais da densidade *a posteriori* [8], propriedade que demonstramos no Apêndice.

III. MODELO DE SINAL

Neste trabalho temos como principal interesse a equalização de canais de comunicação lineares com resposta ao impulso de duração finita. Assim, admitimos que a amostra y_k do sinal

Escola Politécnica, Universidade de São Paulo, São Paulo, Brasil, E-mails: {bordin,baccala}@lcs.poli.usp.br.

1) Passo E (*expectation*):

$$Q(\theta, \theta^{(i)}) = \int_B \log p(\theta|A, B) p(B|A, \theta^{(i)}) dB \quad (1)$$

2) Passo M (*maximization*):

$$\theta^{(i+1)} = \arg \max_{\theta} Q(\theta, \theta^{(i)}) \quad (2)$$

3) Itera-se até a convergência, ou seja, até que $\|\theta^{(i+1)} - \theta^{(i)}\|$ seja suficientemente pequeno.

TABELA I
ALGORITMO EM

recebido no instante k (tomada à taxa de símbolos) pode ser expressa como:

$$y_k = \sum_{i=0}^{L-1} h_i x_{k-i} + v_k, \quad (3)$$

em que L denota a duração da resposta ao impulso do canal (em intervalos de símbolo), x_k denota a seqüência de símbolos transmitida, h_k os termos da resposta impulsiva do canal e v_k o ruído aditivo, por hipótese gaussiano complexo circular com média nula e variância σ^2 conhecida.

Observe ainda que o modelo de sinal dado em (3) pode ser representado alternativamente como

$$y_k = h^H X_k + v_k, \quad (4)$$

ao fazer com que $X_k = [x_k \ x_{k-L+1}]^T \in \mathbb{C}^{L \times 1}$ seja um vetor que coleciona os valores dos últimos L símbolos transmitidos e $h = [h_0 \ h_{L-1}]^T \in \mathbb{C}^{L \times 1}$ seja o vetor que coleciona os coeficientes do canal.

IV. EQUALIZAÇÃO CEGA EMPREGANDO O ALGORITMO EM

Neste trabalho temos como objetivo obter estimativas MAP dos parâmetros do canal de transmissão h , ou seja, determinar o conjunto de parâmetros h que maximiza a densidade *a posteriori* $p(h|y_{0:n}, \sigma^2)$, $y_{0:n} \triangleq \{y_0, \dots, y_n\}$.

Pode-se mostrar que a determinação exata da densidade $p(h|y_{0:n}, \sigma^2)$ é impraticável [9], uma vez que exige um número de operações que cresce exponencialmente em função do tamanho do bloco de dados n . No entanto, conforme mostramos nesta seção, o modelo de sinal descrito por (4) permite que se obtenham diretamente estimativas MAP dos parâmetros h através do algoritmo EM.

Utilizando como variável latente a seqüência de vetores $X_{0:n}$, assumida independente *a priori* do vetor de parâmetros h , obtemos que a densidade *a posteriori* aumentada

$p(h|X_{0:n}, y_{0:n}, \sigma^2)$ pode ser expressa¹ como

$$p(h|X_{0:n}, y_{0:n}) = \frac{p(h, X_{0:n}, y_{0:n})}{\int_h p(h, X_{0:n}, y_{0:n}) dh} \quad (5)$$

$$\propto p(h, X_{0:n}, y_{0:n}) \quad (6)$$

$$= p(y_{0:n}|h, X_{0:n}) p(h) p(X_{0:n}) \quad (7)$$

$$= p(h) \prod_{j=0}^n p(y_j|h, X_j) p(x_j). \quad (8)$$

Aplicando-se o algoritmo EM para conseguir estimativas de h , obtemos a seguinte expressão para o “passo E”:

$$Q(h, h^{(i)}) = \sum_{X_{0:n}} \log p(h|X_{0:n}, y_{0:n}) p(X_{0:n}|h^{(i)}, y_{0:n}) \quad (9)$$

$$= E \left[\log p(h|X_{0:n}, y_{0:n}) | h^{(i)}, y_{0:n} \right]. \quad (10)$$

Assumindo que a densidade *a priori* dos parâmetros do canal é Gaussiana complexa, ou seja,

$$p(h) = \left(\frac{1}{\pi} \right)^N |\Sigma_0|^{-1} \exp \left\{ -\|h - h_0\|_{\Sigma_0}^2 \right\} \\ \triangleq \mathcal{N}_C(h|h_0; \Sigma_0), \quad (11)$$

e que os símbolos transmitidos são i.i.d equiprováveis, segue que a verossimilhança das observações é dada por

$$p(y_j|h, X_j) = \mathcal{N}_C(y_j|h^H X_j; \sigma^2). \quad (12)$$

O logaritmo da densidade *a posteriori* aumentada dada em (8) pode ser escrito então como

$$\log p(h|X_{0:n}, y_{0:n}) = \alpha - \|h - h_0\|_{\Sigma_0}^2 - \frac{1}{2} \sum_{j=0}^n \|y_j - h^H X_j\|^2. \quad (13)$$

sendo α uma constante de normalização independente de h . Calculando-se então a esperança condicional de (13), obtemos

$$Q(h, h^{(i)}) = -\frac{1}{2} \sum_{j=0}^n E \left[\|y_j - h^H X_j\|^2 | h^{(i)}, y_{0:n} \right] \\ - \|h - h_0\|_{\Sigma_0}^2 + \alpha' \quad (14)$$

$$= -\frac{1}{2} \sum_{j=0}^n \left(-2\Re \left\{ h^H E \left[X_j | h^{(i)}, y_{0:n} \right] y_j \right\} \right. \\ \left. + h^H E \left[X_j X_j^H | h^{(i)}, y_{0:n} \right] h \right) \\ - 2\Re \left\{ h^H \Sigma_0^{-1} h_0 \right\} + h^H \Sigma_0^{-1} h_0 + \alpha''. \quad (15)$$

em que α' e α'' são constantes que não dependem de h .

O segundo passo do algoritmo EM (passo “M”) consiste de determinar o valor de h que maximiza (15). Para isto, note inicialmente que

$$\frac{\partial}{\partial h} Q(h, h^{(i)}) = \\ - \left(\Sigma_0^{-1} + \frac{1}{2} \sum_{j=0}^n E \left[X_j X_j^H | h^{(i)}, y_{0:n} \right] \right) h \\ + \left(\Sigma_0^{-1} h_0 + \frac{1}{2} \sum_{j=0}^n E \left[X_j | h^{(i)}, y_{0:n} \right] y_j \right). \quad (16)$$

¹Omitiremos a partir deste ponto a dependência das densidades condicionais em relação ao parâmetro σ^2 (variância do ruído aditivo), uma vez que este é assumido conhecido.

e que a matriz Hessiana $\partial^2 Q(h, h^{(i)})/\partial h^2$ é dada por

$$-\left(\Sigma_0^{-1} + \frac{1}{2} \sum_{j=0}^n E \left[X_j X_j^H | h^{(i)}, y_{0:n} \right]\right) < 0. \quad (17)$$

Igualando-se (16) a zero, segue que o máximo de $Q(h, h^{(i)})$ é dado por

$$h^{(i+1)} = \left(\Sigma_0^{-1} + \frac{1}{2} \sum_{j=0}^n E \left[X_j X_j^H | h^{(i)}, y_{0:n} \right] \right)^{-1} \left(\Sigma_0^{-1} h_0 + \frac{1}{2} \sum_{j=0}^n E \left[X_j | h^{(i)}, y_{0:n} \right] y_j \right) \quad (18)$$

As esperanças em (18) podem ser calculadas numericamente através do algoritmo BCJR [10] (vide Apêndice), uma vez que

$$E \left[X_j | h^{(i)}, y_{0:n} \right] = \sum_{i=1}^{N_s} \mathcal{X}_i \mathbb{P}(X_j = \mathcal{X}_i | h^{(i)}, y_{0:n}) \quad (19)$$

$$E \left[X_j X_j^H | h^{(i)}, y_{0:n} \right] = \sum_{i=1}^{N_s} \mathcal{X}_i \mathcal{X}_i^H \mathbb{P}(X_j = \mathcal{X}_i | h^{(i)}, y_{0:n}) \quad (20)$$

sendo N_s o número de vetores X_j distintos.

Na Tabela II resumimos o conteúdo desta seção, descrevendo o algoritmo proposto. Note que aproximações das estimativas² MAP/ML dos símbolos transmitidos podem ser obtidas a partir de $\mathbb{P}(X_j | h^{(i)}, y_{0:n})$, uma vez que a densidade *a posteriori* dos símbolos transmitidos é dada por

$$\mathbb{P}(x_j = a | h^{(i)}, y_{0:n}) = \sum_{\forall b | x_j = a} \mathbb{P}(X_j = b | h^{(i)}, y_{0:n}). \quad (21)$$

(Inicialização) Obtenha $h^{(0)} \sim \mathcal{N}_C(h|0; \Sigma_0)$
Para $i = 0$, itere:

- Calcule $\mathbb{P}(X_j | h^{(i)}, y_{0:n})$, $0 \leq j \leq n$ através do algoritmo BCJR.
- Determine $h^{(i+1)}$ através de (18) - (20).

TABELA II

ALGORITMO PARA EQUALIZAÇÃO CEGA BASEADO NO ALGORITMO EM

A. Simulações

Através de simulações numéricas avaliamos o desempenho do algoritmo segundo a taxa média de erro de bit (BER) obtida ao longo de 100 realizações independentes num sistema de comunicação que transmite blocos de 250 símbolos binários i.i.d. através do canal $h = [0, 41 - 0, 82 \ 0, 41]^T$ com ordem e potência de ruído conhecidas pelo receptor. A densidade *a priori* do canal foi admitida igual a $\mathcal{N}_C(h|0; I)$.

Na Fig. 1 mostramos o desempenho obtido em função da relação sinal-ruído e do número de iterações empregado pelo

²Observe que as estimativas obtidas através deste método são subótimas, uma vez que $h^{(i)}$ é uma estimativa dos parâmetros do canal.

algoritmo EM. Na mesma figura, para efeito de comparação, mostramos o desempenho obtido pelo algoritmo de Viterbi, construído a partir do canal conhecido e implementado considerando-se desconhecidos os bits das extremidades do bloco. Uma característica que salta aos olhos é o desempenho ruim do algoritmo da Sec. IV que piora a medida que a relação sinal ruído ultrapassa 6dB.

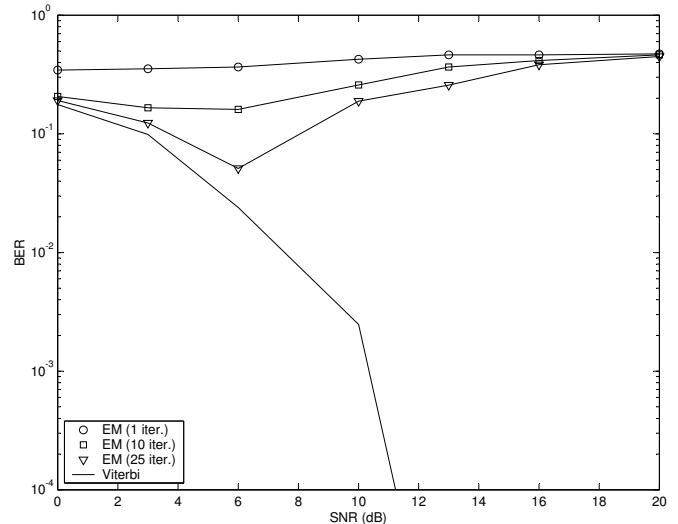


Fig. 1. Desempenho médio ao longo de 100 realizações independentes do algoritmo para equalização cega descrito na Sec. IV em função da relação sinal-ruído(SNR) e o número de iterações.

V. ALGORITMO MELHORADO

Analisando mais detidamente o funcionamento do algoritmo da Sec. IV observamos que, sob baixos níveis de ruído, as densidades $p(X_k | h^{(i)}, y_{0:n})$ estimadas (através do algoritmo BCJR) assumem valores muito próximos de 0 ou 1, num comportamento que parece contribuir para o aparecimento de algo semelhante a um ciclo-limite³, que impede a convergência do algoritmo para uma solução fixa.

Para contornar este problema, fizemos uma alteração de caráter heurístico no algoritmo da Sec. IV inspirada em técnicas de *simulated annealing*. Estas técnicas [11], amplamente empregadas na solução de problemas de otimização não-linear, baseiam-se na observação de fenômenos naturais ligados à cristalização de metais, que sugerem que a convergência para o mínimo global num problema de minimização iterativa pode ser obtida pela introdução de perturbações lentamente decrescentes (nas suas soluções intermediárias).

Para evitar o aparecimento de oscilações cíclicas como as observadas para o algoritmo anteriormente descrito, substituímos a densidade $p(X_k | h^{(i)}, y_{0:n}, \delta(i))$ (calculada através do algoritmo BCJR), necessária para a determinação de (20), pela densidade $\bar{p}_i(X_k | h^{(i)}, y_{0:n}, \delta(i)) = p(X_k | h^{(i)}, y_{0:n}, \delta(i) + \delta(i))$, com $\delta(i) > 0$ é uma função decrescente de i .

Note que, dado o maior nível de ruído informado ao algoritmo BCJR, a densidade $\bar{p}_i(X_k | h^{(i)}, y_{0:n}, \delta(i))$ exibe uma

³Note que o algoritmo EM opera de forma determinística, e que os valores de $h^{(i)}$ e $y_{0:n}$ numa iteração definem completamente o comportamento do algoritmo nas iterações posteriores.

menor tendência de assumir valores extremos (0 ou 1) do que a densidade *a posteriori* verdadeira. Isto parece contribuir para a eliminação de oscilações cíclicas. Em nossas simulações, testamos diversas formas para a função $\delta(i)$, obtendo melhores resultados para funções $\delta(i)$ da forma exponencial.

A. Simulações

Nas Figs. 2 e 3 mostramos o desempenho obtido pela versão modificada do algoritmo da Sec. IV, utilizando $\delta(i) = 10^{-0,1i}$ e $\delta(i) = 10^{-0,05i}$, respectivamente, empregando as mesmas hipóteses adotadas na Sec. IV-A.

Como se pode observar, ao fazer $\delta(i) = 10^{-0,1i}$ nota-se ainda uma degradação de desempenho para SNR superiores a 12 dB (Fig. 2). Diminuindo-se o nível de ruído adicional mais lentamente ($\delta(i) = 10^{-0,05i}$), no entanto, o desempenho do algoritmo modificado aproxima-se bastante daquele do detector ótimo MLSE (Viterbi) para 25 iterações, como se pode observar na Fig. 3.

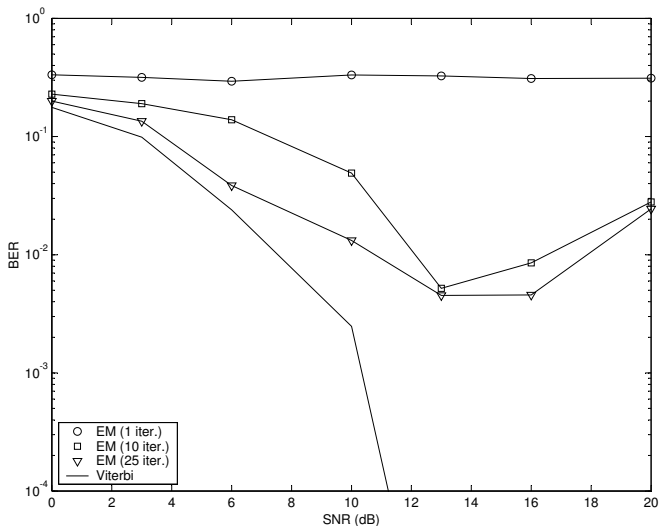


Fig. 2. Desempenho médio ao longo de 100 realizações independentes da versão modificada do algoritmo para equalização cega descrito na Sec. V ($\delta(i) = 10^{-0,1i}$) em função da relação sinal-ruído(SNR) e o número de iterações.

VI. CONCLUSÕES

Neste trabalho descrevemos métodos para a equalização cega de canais de comunicação lineares FIR baseados na versão Bayesiana do algoritmo EM (*Expectation-Maximization*). Verificamos que o método básico, obtido através da aplicação direta do algoritmo EM ao problema tratado, apresenta um desempenho insatisfatório sob baixos níveis de ruído. Este resultado nos levou a propor um segundo esquema, empregando uma modificação heurística inspirada nas técnicas de *simulated annealing*, o qual, sem incorrer num aumento da complexidade computacional, proporciona um desempenho próximo ao ótimo, conforme verificado através simulações numéricas.

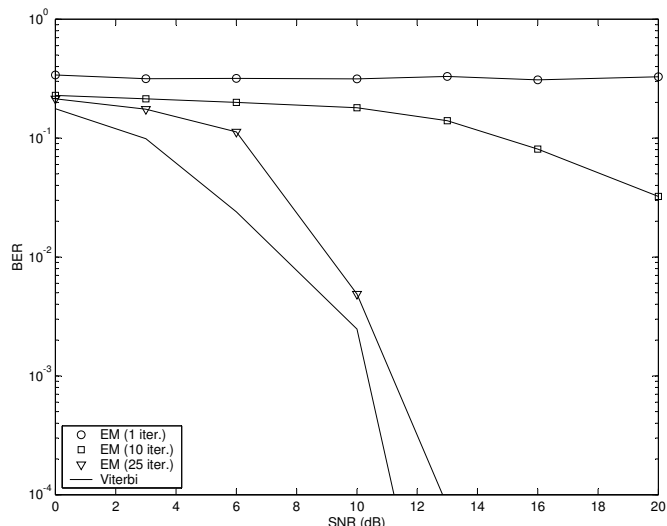


Fig. 3. Desempenho médio ao longo de 100 realizações independentes da versão modificada do algoritmo para equalização cega descrito na Sec. V ($\delta(i) = 10^{-0,05i}$) em função da relação sinal-ruído(SNR) e o número de iterações.

REFERÊNCIAS

- [1] S. Haykin, ed., *Blind Deconvolution*. Englewood Cliffs, NJ: Prentice Hall, 1994.
- [2] Z. Ding and Y. Li, eds., *Blind Equalization and Identification*. New York: Marcel Dekker, 2001.
- [3] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. Springer Verlag, 1999.
- [4] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society*, vol. 39, pp. 1–38, 1977.
- [5] L. Scharf, *Statistical signal processing: detection, estimation, and time series analysis*. Addison-Wesley Pub. Co., 1991.
- [6] K. Dogancay and V. Krishnamurthy, "Blind identification of fractionally spaced communication channels with markov inputs," *Signal Processing*, vol. 77, pp. 275–288, Setembro 1999.
- [7] M. Nissilä and S. Pasupathy, "Adaptive bayesian and em-based detectors for frequency-selective fading channels," *IEEE Transactions on Communications*, vol. 51, pp. 1325–1336, Agosto 2003.
- [8] M. A. Tanner, *Tools for Statistical Inference*. Springer Verlag, 3 ed., 1996.
- [9] A. Doucet, C. Andrieu, and R. Urien, "Bayesian blind and semi-blind equalization of channels with markov inputs," *IEE Proceedings-F Image & Signal Processing*, vol. 148, no. 4, pp. 269–274, 2001.
- [10] L. R. 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, Março 1974.
- [11] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.
- [12] G. D. Forney, "The {V}iterbi algorithm," *Proceedings of IEEE*, vol. 61, no. 3, pp. 268–278, Março 1973.

APÊNDICE

A. Convergência do Algoritmo EM

Proposição: Iterando-se o algoritmo da Tabela I tem-se que $p(\theta^{(i+1)}|A) \rightarrow p(\theta^{(i)}|A)$.

Prova: Inicialmente, note que como uma consequência lei de Bayes

$$p(\theta|A, B) = \frac{p(\theta|A)p(B|\theta, A)}{p(B|A)}, \quad (22)$$

o que implica que

$$\log[p(\theta|A)] = \log[p(\theta|A, B)] + \log[p(B|A)] - \log[p(B|\theta, A)]. \quad (23)$$

Integrando-se cada uma destas densidades com respeito a $p(B|\theta, A)$:

$$\begin{aligned} \log[p(\theta|A)] &= \int_B \log[p(\theta|A, B)] p(B|\theta, A) dB \\ &+ \int_B \log[p(B|A)] p(B|\theta, A) dB \\ &- \int_B \log[p(B|\theta, A)] p(B|\theta, A) dB \\ &\triangleq Q(\theta, \theta) + K(\theta, \theta) - H(\theta, \theta) \end{aligned} \quad (24)$$

Fazendo-se $\theta = \theta^{(i)}$ obtemos

$$\begin{aligned} \log[p(\theta^{(i+1)}|A)] - \log[p(\theta^{(i)}|A)] &= \\ &[Q(\theta^{(i+1)}, \theta^{(i)}) - Q(\theta^{(i)}, \theta^{(i)})] \\ &+ [H(\theta^{(i+1)}, \theta^{(i)}) - H(\theta^{(i)}, \theta^{(i)})] \\ &- [K(\theta^{(i+1)}, \theta^{(i)}) - K(\theta^{(i)}, \theta^{(i)})] \end{aligned} \quad (25)$$

Como, por definição, $\theta^{(i+1)}$ é escolhido de modo a maximizar $Q(\theta^{(i+1)}, \theta^{(i)})$, o primeiro termo em (25) é positivo (ou nulo se $\theta^{(i+1)} = \theta^{(i)}$). Pode-se verificar também que o segundo termo é nulo, já que $H(\theta^{(i+1)}, \theta^{(i)})$ não é função de $\theta^{(i+1)}$. Quanto ao terceiro termo, note que

$$\begin{aligned} -[K(\theta^{(i+1)}, \theta^{(i)}) - K(\theta^{(i)}, \theta^{(i)})] &\triangleq \\ \int_B \log \left[\frac{p(B|\theta^{(i)}, A)}{p(B|\theta^{(i+1)}, A)} \right] p(B|\theta^{(i)}, A) dB \end{aligned} \quad (26)$$

ou seja, a diferença equivale à distância de Kullback-Leibler entre as densidades $p(B|\theta^{(i)}, A)$ e $p(B|\theta^{(i+1)}, A)$, sendo portanto positiva (ou nula de as duas densidades forem iguais “almost everywhere”). Assim, conclui-se que $\log[p(\theta^{(i+1)}|A)] - \log[p(\theta^{(i)}|A)] \geq 0$ concluindo a demonstração. \square

B. O Algoritmo BCJR

O Algoritmo BCJR [10], desenvolvido por Bahl *et al.* em 1974 descreve um método numérico para o cálculo das probabilidades de ocupação dos estados de um HMM (*Hidden Markov Model*). O método é implementado através de duas recursões (para trás e para frente), sendo a primeira delas bastante similar à descrita pelo Algoritmo de Viterbi [12].

Para se calcular a distribuição *a posteriori* da variável de estado X_k (Eq. 20), o algoritmo BCJR explora o fato de que $y_{0:n}$ são observações de um HMM, obtendo que

$$\begin{aligned} p(X_{k+1} = b | y_{0:n}) &= \sum_a p(X_k = a, X_{k+1} = b | y_{0:n}, h) \\ &\propto \sum_a \alpha_k(a) \gamma_k(a, b) \kappa_{k+1}(b), \end{aligned} \quad (27)$$

onde

$$\alpha_k(a) = p(X_k = a, y_{0:k-1}) \quad (28)$$

$$\kappa_{k+1}(b) = p(y_{k+1:n} | X_{k+1} = b) \quad (29)$$

$$\gamma_k(a, b) = p(X_{k+1} = b, y_k | X_k = a). \quad (30)$$

Sob as hipóteses adotadas para o modelo de sinal (Eq. 3), os termos $\gamma_k(a, b)$ podem ser calculados diretamente, obtendo-se que

$$\gamma_k(a, b) = \frac{1}{2} \exp \left[-\frac{1}{2} |y_k - h^H X_k|^2 \right] p(x_k = s^{(a,b)}), \quad (31)$$

onde $s^{(a,b)}$ denota o valor assumido pela v.a. x_k que provoca a transição do estado $X_k = a$ para o estado $X_{k+1} = b$.

Os demais termos em (30) somente podem ser determinados recursivamente. Para isto, introduzimos a seguinte notação vetorial

$$\alpha_k = [\alpha_k(1) \dots \alpha_k(N_s)]^T \in \mathbb{R}^{|N_s| \times 1} \quad (32)$$

$$\kappa_k = [\kappa_k(1) \dots \kappa_k(N_s)]^T \in \mathbb{R}^{|N_s| \times 1}, \quad (33)$$

e a matriz

$$[\mathbf{\Gamma}_k]_{ij} = \gamma_k(i, j), \quad (34)$$

onde N_s é o número de valores distintos que o vetor aleatório X_k pode assumir.

Utilizando estas definições, obtém-se que os demais termos podem ser calculados através das recursões [10]

$$\alpha_{k+1} = \mathbf{\Gamma}_k^T \alpha_k \quad (35)$$

$$\kappa_k = \mathbf{\Gamma}_k \kappa_{k+1}, \quad (36)$$

iniciadas com $\kappa_k(a) = p(y_k | X_k = a)$ e $\alpha_0(a) = p(X_0 = a)$.

Numa formulação geral, a aplicação do algoritmo BCJR a um bloco de n amostras exige um total de operações da ordem de nN_s^2 . Ao aplicar-se o algoritmo BCJR num sistema de comunicações digitais binário, no entanto, verifica-se que a matriz $\mathbf{\Gamma}_k$ somente possui $2N_s$ elementos não-nulos, onde $N_s = 2^L$, sendo a complexidade total uma função linear de N_s .