



Notas Técnicas do LCD

Publicação para divulgação técnica do Laboratório de Comunicações Digitais do IME

Autorizada a utilização sem fins lucrativos

No. 02 - 24 de Outubro de 2011

Estimação de Parâmetros de Modelos HMM para Canais com Erros em Surtos

Nilson Maciel de Paiva Junior¹, Elaine Crespo Marques², Ernesto Leite Pinto³

¹ nilsonmpj@ime.eb.br, ² elaincecmарques@ime.eb.br, ³ ernesto@ime.eb.br

Resumo

Este artigo trata da aplicação de modelos HMM (“Hidden Markov Models”) a canais de comunicações com erros em surtos. Após uma breve discussão sobre a importância da geração de modelos para este tipo de erro, é feita uma descrição genérica do problema de obtenção de modelos ajustados a amostras de erros. Uma ênfase particular é dada à necessidade de métodos adequados para a estimação dos parâmetros dos modelos em questão, de modo a obter modelos úteis de forma eficiente. Alguns métodos de estimação são revistos brevemente.

1 Introdução

A ocorrência de erros em surtos é uma importante fonte de limitação de desempenho em muitos sistemas de comunicações de interesse atual, estando presente não apenas em sistemas que se baseiam na transmissão sem fio mas também em sistemas de transmissão em meio confinados, como os que se utilizam da rede de distribuição de eletricidade (PLC, de “Power Line Communications”).

Para se investigar em profundidade o impacto deste tipo de erros sobre o desempenho de protocolos de diferentes níveis [1,2], faz-se necessário o desenvolvimento de modelos matemáticos que reproduzam adequadamente as propriedades probabilísticas de amostras de erros colhidas por simulações dos enlaces físicos de comunicação ou em condições reais de operação [2].

Os modelos de Markov escondidos (HMM, de “Hidden Markov Model”) têm se mostrado de grande utilidade para representação eficiente dos mecanismos de memória presentes nos processos de erros em surtos [3–6].

Um modelo HMM é caracterizado pelo número de estados, pelas probabilidades de transição de estado, pela distribuição de probabilidades do estado em que o modelo é inicializado, e pelas distribuições condicionais da observação, dado cada um dos estados.

Uma questão importante para o estabelecimento de modelos HMM úteis é a do ajuste de seus parâmetros a amostras empíricas de erros, de modo que o modelo obtido reproduza satisfatoriamente propriedades probabilísticas desta amostra de erros. Um critério frequentemente adotado para este ajuste é o de máxima verossimilhança (ML, de “Maximum-Likelihood”), sendo bastante usual o emprego do algoritmo Baum-Welch (BW) para este fim.

O ajuste de parâmetros de modelos HMM a amostras de erros em surtos é o tema central da presente Nota Técnica, que está organizada em 6 seções. Uma revisão sobre os conceitos de erros em surtos pode ser encontrada em [3].

A seção 2 contém uma breve introdução aos modelos HMM, maiores informações podem ser obtidas em [3]. Na seção 3 se discute o problema de ajustar modelos HMM a amostras de erros em surtos. Na seção 4 são apresentados alguns algoritmos para estimação de parâmetros de modelos HMM. A seção 5 contém exemplos de aplicação destes algoritmos para modelagem de erros em surtos. Comentários finais são apresentados na seção 6.

2 Modelos de Markov Escondidos (HMM)

Os modelos HMM não se restringem apenas à modelagem de erros em surtos, estes modelos têm sido aplicados a uma vasta gama de problemas nela se incluindo reconhecimento de locutor [7], processamento de imagem e visão computacional [8], reconhecimento de texto [9], entre outros.

Um modelo desta classe descreve uma sequência de amostras indexadas por um parâmetro discreto como sendo gerada aleatoriamente a partir de estados não observados modelados como uma cadeia de Markov. Assim sendo, a cada valor do índice o modelo se encontra num estado pertencente a um conjunto finito e a probabilidade condicional de que o modelo assim se encontre, dada toda a sua evolução passada, é igual à probabilidade condicional de que se encontre no mesmo estado dada apenas a sua situação no valor do índice imediatamente anterior.

Na aplicação de modelos HMM é usual admitir que as probabilidades de transição de estado são invariantes no tempo, ou seja, que a cadeia de Markov associada ao modelo é homogênea.

Para que um modelo HMM deste tipo seja completamente especificado, é necessário portanto estabelecer os seguintes parâmetros:

- **N** - Número de estados;
- **m** - Número de símbolos do alfabeto de observações. Admite-se a partir deste ponto que a observação assume valores em um conjunto discreto e finito. Ou seja, que cada observação $o_t \in \{v_1, v_2, \dots, v_m\}$, que é o conjunto alfabeto de observações. As probabilidades de transição $a_{i,j}$ devem atender às condições apresentadas na Equação (2).
- **A** - Matriz das probabilidades de transição representada em (1), onde q_t representa o estado no tempo t .

$$A = \{a_{i,j}\}; \quad a_{i,j} = P(q_{t+1} = j | q_t = i); \quad 1 \leq i, j \leq N \quad (1)$$

$$0 \leq a_{i,j} \leq 1; \quad \sum_{j=1}^N a_{i,j} = 1; \quad 1 \leq i, j \leq N; \quad (2)$$

- **B** - Matriz das distribuições de probabilidades em cada estado representada na Equação (3), onde v_k representa o k -ésimo símbolo de observação no alfabeto e o_t a observação no tempo t . As

probabilidades b_j devem obedecer às condições apresentadas na Equação (4) .

$$B = \{b_j(k)\}; \text{ onde: } b_j(k) = P(o_t = v_k | q_t = j) \quad 0 \leq j \leq N; \quad 1 \leq k \leq M \quad (3)$$

$$0 \leq b_j(k) \leq 1; \quad \sum_{k=1}^m b_j(k) = 1; \quad 0 \leq j \leq N; \quad 1 \leq k \leq M \quad (4)$$

- Π - Vetor das probabilidades iniciais de cada estado apresentada na Equação (5). As probabilidades Π_i devem satisfazer às condições apresentadas na Equação (6) .

$$\Pi = \{\Pi_i\}; \quad \text{onde:} \quad \Pi_i = p(q_1 = i) \quad (5)$$

$$0 \leq \Pi_i \leq 1; \quad \sum_{i=1}^N \Pi_i = 1; \quad 1 \leq i \leq N \quad (6)$$

Assim sendo, representa-se um modelo HMM com observações discretas por $\lambda(A, B, \Pi)$. No caso específico da modelagem de canais com erros em surtos, é bastante comum se admitir o emprego de decisão abrupta, de modo que a sequência de observações é binária ($M = 2$), sendo usualmente representada por zeros (para decisões corretas) e uns (para ocorrências de erro).

3 Ajuste de Modelos HMM a Dados Empíricos

A Figura 1 apresenta uma visão geral do problema da escolha de um modelo HMM ajustado a uma determinada amostra de erros em surtos. O ponto de partida é um canal de comunicação com uma “fonte de erros” que produz uma “sequência de erros”. Esta sequência é composta por bits “0”, indicando decisões corretas e bits “1” que indicando a ocorrência de erros.

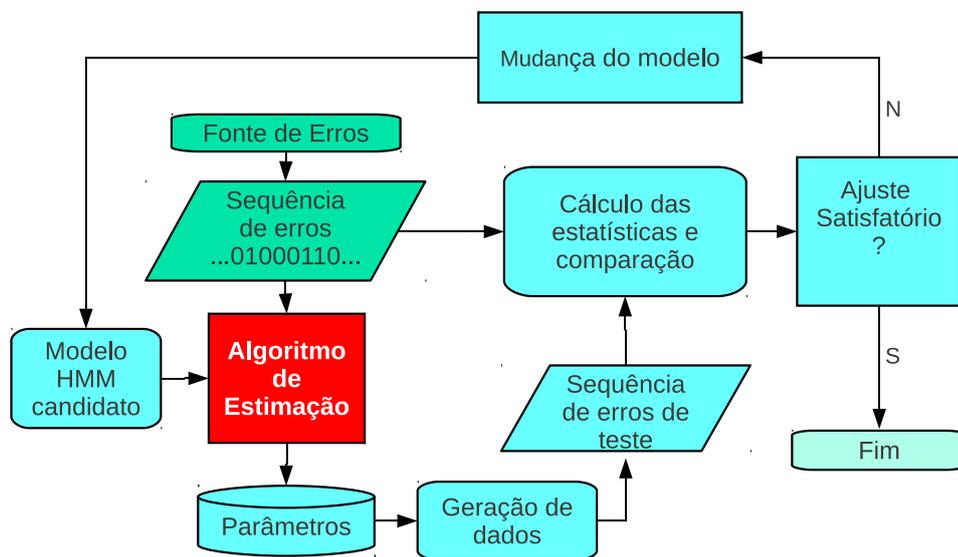


Figura 1: Apresentação do Problema de Modelagem de Erros em Surtos.

De modo a modelar a sequência de erros assume-se que a estrutura do “modelo HMM candidato” é inicialmente escolhida. Em seguida, estima-se os parâmetros do modelo HMM através de algum “algoritmo de estimação”. Depois disso, o modelo ajustado é utilizado para gerar uma “sequência de erros de teste” a partir da qual são realizados os cálculos dos diversos parâmetros estatísticos utilizados em erros em surtos. Por fim, esses parâmetros são comparados aos da sequência de erros original. Se ocorrer um ajuste satisfatório entre estas estatísticas, o modelo candidato é aceito. Caso contrário, há

uma mudança na estrutura do modelo HMM candidato, como por exemplo alteração da quantidade de estados do modelo, e o novo modelo HMM candidato é avaliado de forma semelhante. Este ciclo deve ser repetido até que um ajuste adequado da sequência de erros seja alcançado.

Vale ressaltar que podem ser necessárias várias tentativas de modelos HMM candidatos a fim de se encontrar um modelo satisfatório. Destacamos que o algoritmo de estimação empregado para ajustar os parâmetros do modelo HMM desempenha um papel crucial na busca por um bom modelo, tendo impacto significativo na sua complexidade computacional e tempo de processamento.

4 Estimação de Parâmetros

O critério de estimação mais frequentemente empregado no ajuste de modelos HMM a amostras de erros em surtos é o de máxima verossimilhança (ML). Este é de fato um método clássico de estimação de parâmetros, com ampla utilização em diversos problemas atuais de muitas áreas de conhecimento e investigação.

A ideia básica da estimação ML é buscar os valores dos parâmetros para os quais a probabilidade de observação de uma dada amostra seja máxima. Esta probabilidade, vista como função dos parâmetros, é denominada *função de verossimilhança*.

No caso dos modelos HMM aqui adotados, supondo uma amostra dada por $\{o_1, o_2, \dots, o_{N_a}\}$, a estimativa de máxima verossimilhança dos parâmetros λ fica dada por: $\hat{\lambda}_{ML} = \underset{\lambda}{\operatorname{argmax}} P(o_1, o_2, \dots, o_{N_a} | \lambda)$.

Cabe aqui registrar que em geral pode ser conveniente implementar o critério ML através de uma forma equivalente que consiste na maximização do logaritmo neperiano da função verossimilhança acima apresentada. Neste caso, tem-se: $\hat{\lambda}_{ML} = \underset{\lambda}{\operatorname{argmax}} \ln[P(o_1, o_2, \dots, o_{N_a} | \lambda)]$.

A aplicação deste critério a modelos HMM requer o emprego de um algoritmo de otimização e de fato a escolha deste algoritmo é um objeto de pesquisas atuais, uma vez que a função de verossimilhança e o seu logaritmo se caracterizam por apresentar máximos locais, o que dificulta sobremaneira a obtenção da solução ML propriamente dita.

Apresenta-se a seguir uma breve descrição de alguns algoritmos corretamente empregados ou investigados neste contexto.

4.1 Algoritmo Baum-Welch (BW)

O algoritmo Baum-Welch [10] tem sido a ferramenta mais utilizada para a estimação de parâmetros de modelos HMM. É um caso particular do bem conhecido algoritmo EM para a estimação ML e pode ser descrito pelos seguintes passos:

Passo 0: Inicializar a estimativa de parâmetros $\hat{\lambda} = \{\hat{A}, \hat{B}, \hat{\Pi}\}$.

Passo 1: Calcular as variáveis progressivas e regressivas, utilizando $\hat{\lambda}$.

$$\alpha_t(i) = P[O_1, O_2, \dots, O_t, s_t = i | \hat{\lambda}] \quad \beta_t(i) = P[O_{t+1}, O_{t+2}, \dots, O_T | s_t = i, \hat{\lambda}] \quad (7)$$

Passo 2: Atualizar $\hat{\lambda}$, usando α_t e β_t obtidos no Passo 1. As expressões para esta atualização podem ser obtidas em [10].

Passo 3: Verificar a condição de parada. Se esta não for atendida, retornar para o Passo 1 com a atualização de $\hat{\lambda}$.

4.2 Treinamento baseado no Algoritmo de Viterbi (VT)

Este método possibilita estimar recursivamente as matrizes A e B do modelo¹ e consiste basicamente em obter a sequência de estados com máxima probabilidade a posteriori (MAP) associada às estimativas atuais de A e B , e usá-la para atualizar estas estimativas. O VT pode ser descrito pelos seguintes passos:

Passo 0: Inicializar as estimativas \hat{A} e \hat{B} .

Passo 1: Estimar a sequência de estados de máxima probabilidade a posteriori (MAP), usando o algoritmo de Viterbi.

Passo 2: Atualizar as estimativas \hat{A} e \hat{B} pelo cálculo de frequências relativas de transição de estados e de frequências relativas de observações a partir de cada estado, usando a sequência MAP.

Passo 3: Verificar a condição de parada e, se necessário, retornar para o Passo 1 com os novos valores de \hat{A} e \hat{B} .

4.3 Otimização por Enxame de Partículas (PSO)

O PSO é um algoritmo de otimização não linear baseado numa analogia sócio-cognitiva simples, segundo a qual cada indivíduo de uma população é capaz de quantificar a qualidade de sua experiência na busca de um determinado objetivo, e também tem conhecimento dos resultados prévios obtidos por seus vizinhos. Admite-se ainda que um determinado indivíduo decide sobre como prosseguir nesta busca com base nos melhores resultados obtidos previamente por ele mesmo e pelo conjunto de seus vizinhos [11].

Com base nesta analogia, a cada iteração o algoritmo gera e atualiza uma quantidade pré-fixada de amostras dos parâmetros da função a ser otimizada. Cada uma destas amostras é usualmente denominada “partícula” e o seu conjunto é chamado de “enxame”. Uma dada partícula i é representada por um vetor de posição $\vec{x}_i(t)$ e por um vetor de velocidade instantânea $\vec{v}_i(t)$. Além disso, o algoritmo retém a posição \vec{p}_i em que cada partícula obteve o melhor valor da função custo, assim como a posição \vec{p}_g correspondente ao melhor resultado obtido por todo o enxame.

A velocidade e a posição de uma dada partícula são atualizadas segundo as equações a seguir:

$$\vec{v}_i(t+1) = W\vec{v}_i(t) + c_1U_1(\vec{p}_i(t) - \vec{x}_i(t)) + c_2U_2(\vec{p}_g(t) - \vec{x}_i(t)) \quad (8)$$

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \alpha\vec{v}_i(t+1) \quad (9)$$

onde o parâmetro W , chamado de peso de inércia, regula a influência do último valor de velocidade sobre o atual, U_1 e U_2 são valores uniformemente distribuídos no intervalo $[0 \ 1]$, as constantes c_1 e c_2 ponderam a influência da experiência própria passada e social, respectivamente, e α pondera o efeito da velocidade na atualização da posição.

As Equações (8) e (9), assim como os valores de \vec{p}_i e \vec{p}_g , são calculados a cada iteração até que seja satisfeito o critério de parada adotado (por exemplo, um número máximo de iterações permitidas).

5 Exemplos de Aplicação

Em seguida são apresentados alguns resultados de aplicações dos algoritmos acima descritos à estimação ML de modelos HMM para erros em surtos. Longe de querer tratar em profundidade o assunto, pretende-se apenas ilustrar alguns aspectos deste problema, que é na realidade um tema de investigação atual [12, 13].

¹A estimativa do vetor Π é calculada a partir da estimativa de A , admitindo-se que o modelo é iniciado em regime estacionário.

Considera-se nestes exemplos que a função a ser maximizada é o logaritmo neperiano da função verossimilhança ($\ln P(O|\lambda)$) de um modelo HMM com número de estados pré-estabelecido. Esta função será daqui por diante frequentemente denotada por LL (de “log-likelihood”).

5.1 Algoritmo BW e Algoritmo VT

Uma sequência de erros com 20000 bits foi gerada através de um modelo HMM de 3 estados. Foram feitas 120 inicializações para cada quantidade de estados escolhida (2, 3 e 4 estados do modelo HMM estimado).

Para cada inicialização, os modelos HMM eram estimados usando os algoritmos BW e VT e seus tempos de processamento e logaritmos da função verossimilhança alcançados eram armazenados. O número de iterações foi fixado em 50.

Como o tempo de processamento do VT é inferior ao do BW, em seguida, fizemos novas inicializações com o VT até que o mesmo alcançasse o mesmo valor LL alcançado por BW ou então alcançasse o tempo de processamento do BW.

A Figura 2 apresenta os tempos das 120 inicializações (INI) estimadas pelos algoritmos BW e VT considerando um modelo HMM com 3 estados. Pode-se notar que para quase todos os casos, o tempo de processamento do VT foi muito inferior ao do BW.

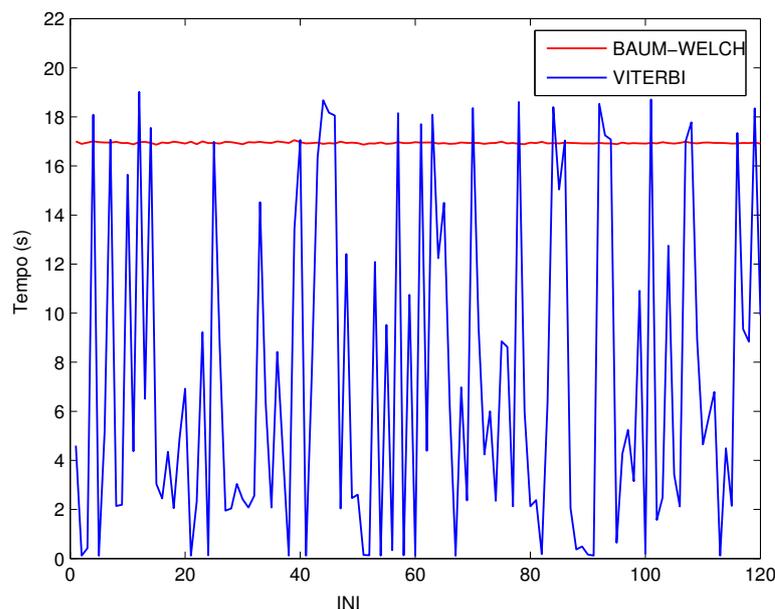


Figura 2: Tempos para a estimação dos modelos HMM com 3 estados.

A Figura 3 apresenta os valores de LL para as 120 inicializações (INI) estimando modelos HMM com 2 estados através dos algoritmos BW e VT. Percebe-se que para a maioria das inicializações a estimação através do VT resultou em valores de LL muito próximos aos dos alcançados pelo BW.

Estes resultados apontam para a possibilidade de se obter com o algoritmo VT estimativas de parâmetros úteis com esforço computacional (tempo de processamento) bem menor do que seria gasto com o algoritmo Baum-Welch.

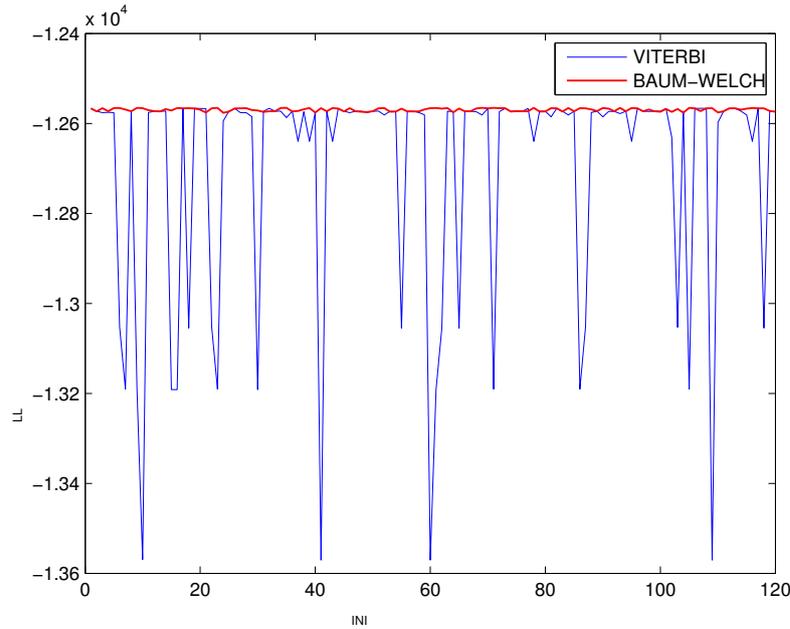


Figura 3: Log-verossimilhança dos modelos HMM com 2 estados.

5.2 Algoritmo BW e Algoritmo PSO

Apresenta-se a seguir um resultado de uma pequena comparação do desempenho dos algoritmos PSO e BW na estimação dos parâmetros dos modelos HMM utilizados em erros em surtos. Para isso, gerou-se uma sequência de erros de tamanho 1000000 produzida por um sistema de transmissão QPSK num canal Rayleigh com espectro Doppler de Jakes e espalhamento Doppler normalizado $f_D T = 10^{-4}$. Assumiu-se sincronização perfeita e a razão E_b/N_0 na entrada do receptor foi fixada em $20dB$.

Estimou-se os parâmetros de modelos HMM com 3 estados a fim de modelar a sequência de erros gerada. Primeiramente, esta sequência foi estimada pelo algoritmo BW com 1300 inicializações aleatórias dos parâmetros, registrando-se o valor de verossimilhança alcançado ao fim de 100 iterações. Em seguida, repetiu-se o mesmo tipo de experimento variando-se o número de iterações e ajustando a quantidade de inicializações de modo a manter o tempo de processamento inalterado.

Nas curvas mostradas na Figura 4, o eixo horizontal corresponde ao percentual das inicializações (INI) em que se obteve valores do logaritmo da função verossimilhança inferiores ao valor de LL indicado no eixo vertical. Percebe-se nesta figura o aumento do percentual de inicializações que levaram a valores mais elevados de LL à medida em que se aumenta o número de iterações. Vê-se também que mesmo com 100 iterações a maioria das estimações resultou em pequenos valores de verossimilhança. Este comportamento pode ser atribuído, pelo menos em parte, à característica bem conhecida do algoritmo BW, que é a de convergir para uma solução de máximo local.

Em seguida, modelou-se a sequência de erros usando o algoritmo PSO com 30 partículas e trincas de parâmetros (W c_1 c_2) iguais a $(0, 2 \ 2, 0 \ 2, 0)$ e $(0, 7 \ 2, 0 \ 2, 0)$, as quais foram utilizadas em [14].

Na Figura 5 apresenta-se (no eixo horizontal) o percentual das inicializações (INI) que levou a valores do logaritmo da função verossimilhança superiores ao valor de LL correspondente (eixo vertical). Observa-se que, para as duas escolhas de parâmetros do algoritmo PSO, a maioria das inicializações levou a valores elevados de verossimilhança. Isto pode ser atribuído à característica do algoritmo PSO de “procurar” soluções que representem máximo global. Percebe-se também na Figura 5 a influência dos parâmetros do PSO nos valores de verossimilhança, tendo sido obtidos melhores valores de

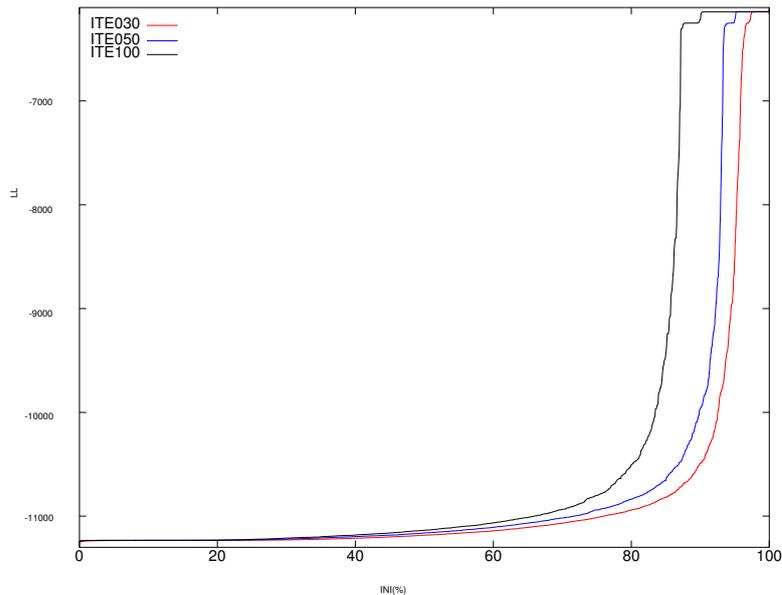


Figura 4: Log-verossimilhança dos modelos HMM estimados pelo algoritmo BW, com 30 (ITE30), 50 (ITE50) e 100 (ITE100) iterações, e mesmo tempo de processamento.

verossimilhança com $W=0,7$ do que com $W=0,2$.

Cabe observar também que os percentuais de inicializações que produziram valores elevados de LL com o algoritmo PSO são significativamente maiores que os melhores percentuais produzidos pelo algoritmo BW. Por exemplo, com o algoritmo BW em 100 iterações aproximadamente 66% das inicializações levaram a valores de LL inferiores a -11000 , enquanto que com o algoritmo PSO e $(W \ c_1 \ c_2) = (0,7 \ 2,0 \ 2,0)$ apenas 5% das inicializações resultaram em LL no mesmo intervalo. Por outro lado, aproximadamente 13% e 55% das inicializações para os algoritmos BW e PSO, respectivamente, geraram valores de LL superiores a -7000 . Estas diferenças de comportamento podem ser atribuídas, pelo menos parcialmente, à maior facilidade do algoritmo PSO em se aproximar do máximo global da função de verossimilhança e da relativa facilidade do algoritmo BW ficar “preso” num máximo local.

Por fim, deve-se lembrar que cada um dos algoritmos acima tem características vantajosas e limitações próprias, de modo que a rigor nenhum deles pode ser considerado superior aos demais. De fato, a investigação de novos algoritmos para este propósito é um tema de pesquisa atual, dado o interesse por este problema de modelagem e a dificuldade de se maximizar a função de verossimilhança neste contexto. Uma possibilidade para a obtenção de novos métodos de estimação é a busca de soluções híbridas, que reúnam características vantajosas de algoritmos distintos. Este é um tema no qual os autores deste texto têm trabalhado recentemente.

6 Conclusões

Discutiu-se aqui a modelagem HMM de erros em surtos, com ênfase na escolha de um algoritmo para estimação de parâmetros deste modelo, usando o critério de máxima verossimilhança. Alguns algoritmos correntemente utilizados ou investigados para esta aplicação foram brevemente apresentados, e foram também mostrados alguns resultados ilustrativos de suas aplicações.

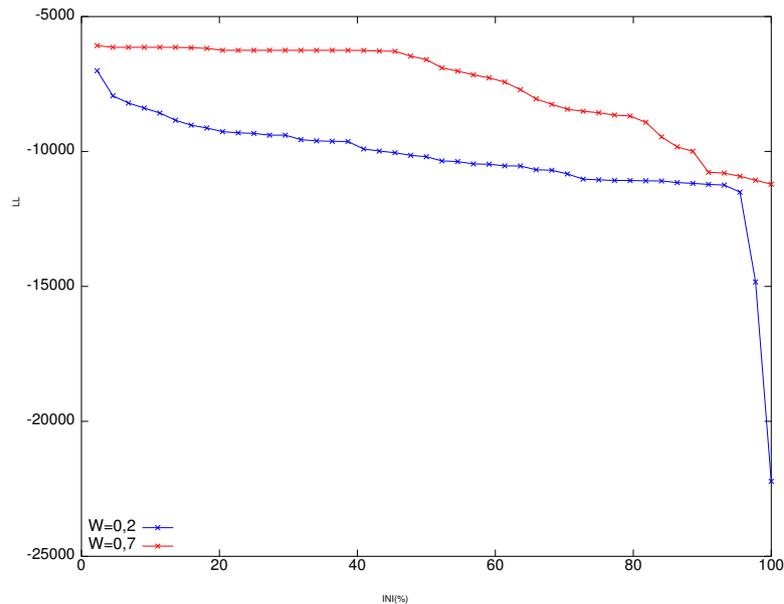


Figura 5: Log-verossimilhança dos modelos HMM estimados pelo algoritmo PSO.

Referências

- [1] M. Zorzi and R.R. Rao. Perspectives on the impact of error statistics on protocols for wireless networks. *IEEE Personal Communications*, 6(5):32–40, oct 1999.
- [2] C. X. Wang and Wen Xu. A new class of generative models for burst-error characterization in digital wireless channels. *IEEE Transactions on Communications*, 55:453–462, March 2007.
- [3] Elaine Marques, Nilson Maciel, and Ernesto Pinto. Modelagem hmm de canais com erros em surtos. In *Notas Técnicas do LCD*, 2011.
- [4] Michele ZORZI, A. Chockalingam, and Ramesh R. Rao. Throughput analysis of tcp on channels with memory. 18(7):1289–1300, July 2000.
- [5] Heissler, R. Jeffrey, Barsoum, A. Yosry, and Richard Condello. An analysis of the viterbi decoder error statistics for atm and tcp/ip over satellite communication. 1999.
- [6] O.S. Salih, C.-X. Wang, and D.I. Laurenson. Three layered hidden markov models for binary digital wireless channels. In *IEEE International Conference on Communications, 2009. ICC '09.*, pages 1–5, 2009.
- [7] Jing Huang and K. Visweswariah. Improved decision trees for multi-stream hmm-based audio-visual continuous speech recognition. In *IEEE Workshop on Automatic Speech Recognition Understanding*, 13 2009.
- [8] Yuexian Zou, Guangyi Shi, Hang Shi, and Yiyan Wang. Image sequences based traffic incident detection for signaled intersections using hmm. In *Ninth International Conference on Hybrid Intelligent Systems*, volume 1, pages 257–261, 2009.
- [9] G.A. Fink, S. Vajda, U. Bhattacharya, S. K. Parui, and B. B. Chaudhuri. Online bangla word recognition using sub-stroke level features and hidden markov models. *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 393–398, 2010.

- [10] J. Bilmes. A gentle tutorial on the em algorithm including gaussian mixtures and baum-welch. In *ICSI Technical Report TR-97-021*, 1997.
- [11] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. pages 39–43, oct 1995.
- [12] N. Maciel, E. Marques, and E. Pinto. Desempenho do Algoritmo PSO na Estimativa ML de Modelos HMM para Erros em Surtos. *Simposio Brasileiro de Telecomunicacoes, SBrT'11*, 2011.
- [13] E. Marques, N. Maciel, and E. Pinto. Tratamento de Restricoes Estocasticas no Algoritmo PSO para Estimacao ML de Modelos HMM. *Simposio Brasileiro de Telecomunicacoes, SBrT'11*, 2011.
- [14] L. Xue, J. Yin, Z. Ji, and L. Jiang. A particle swarm optimization for hidden markov model training. In *8th International Conference on Signal Processing*, 16 2006.

Autores

Nilson Maciel é aluno concludente do curso de Engenharia Eletrônica do Instituto Militar de Engenharia. Fez estágio acadêmico na École Nationale Supérieure des Télécommunications concluído em 2010. Suas principais áreas de interesse são: sistema de comunicações digitais, algoritmos de estimação e dispositivos programáveis. É autor de alguns trabalhos publicados em anais de congressos e revistas de sociedades científicas nacionais e internacionais. É estudante sócio da Sociedade Brasileira de Telecomunicações.

Elaine Crespo Marques é aluna concludente do curso de Engenharia Eletrônica do Instituto Militar de Engenharia. Fez estágio acadêmico na École Nationale Supérieure des Télécommunications concluído em 2010. Suas principais áreas de interesse são: sistema de comunicações digitais e algoritmos de estimação. É autora de alguns trabalhos publicados em anais de congressos e revistas de sociedades científicas nacionais e internacionais. É estudante sócia da Sociedade Brasileira de Telecomunicações.

Ernesto Leite Pinto graduou-se em Engenharia Elétrica pela Universidade Federal da Paraíba, em 1983, tem mestrado e doutorado na área de sistemas de comunicações pela PUC/RJ, concluídos em 1986 e 1998, respectivamente. É professor associado 3 do Departamento de Engenharia Elétrica do IME, onde trabalha desde 1987. Tem ministrado diversas disciplinas de graduação e de pós-graduação, nas áreas de teoria de comunicações, detecção e estimação e transmissão digital. Suas principais áreas de pesquisa são: avaliação de desempenho de sistemas de transmissão digital e processamento digital de sinais aplicado a sistemas de comunicações digitais. É autor de diversos trabalhos publicados em anais de congressos e revistas de sociedades científicas nacionais e internacionais. É bolsista de produtividade em pesquisa do CNPq, de nível II. É sócio e já fez parte do Conselho Deliberativo da Sociedade Brasileira de Telecomunicações.