

# Desempenho do Algoritmo PSO na Estimação ML de Modelos HMM para Erros em Surtos

N. Maciel, E. Marques e E. Pinto

**Resumo**—A aplicação do algoritmo PSO na estimação de máxima verossimilhança (ML) de modelos HMM para canais com erros em surtos é investigada neste trabalho. Uma comparação de desempenho com o algoritmo Baum-Welch (BW) é apresentada, focada na habilidade dos algoritmos em se aproximar da solução de máximo global e nos compromissos entre esforço computacional e desempenho em termos de verossimilhança. Os resultados numéricos aqui apresentados sugerem que o algoritmo PSO tem desempenho claramente superior ao algoritmo Baum-Welch.

**Palavras-Chave**—HMM, Erros em Surto, Baum-Welch, PSO.

**Abstract**—The application of PSO algorithm to perform maximum-likelihood (ML) estimation of HMM models of burst error channels is addressed in this work. A performance comparison with the Baum-Welch (BW) algorithm is presented, with focus on the ability to search the global ML solution and the balance between computational effort and likelihood performance. The numerical results here presented suggest that the PSO algorithm clearly outperforms the Baum-Welch algorithm.

**Keywords**—HMM, Burst Errors, Baum-Welch, PSO.

## I. INTRODUÇÃO

A ocorrência de erros em surtos é um problema frequentemente observado em sistemas de comunicações sem fio, tais como os sistemas via satélite e os sistemas que operam na faixa de HF [1], [2].

A modelagem matemática deste fenômeno e o levantamento e análise de seus parâmetros são os primeiros passos para a investigação do impacto dos erros em surtos sobre os serviços de comunicações finais. Modelos com erros independentes e identicamente distribuídos (IID) são inadequados para erros em surtos, pois estes possuem grau de correlação [3].

A busca por métodos que possibilitem a obtenção de modelos precisos e que possam ser gerados rapidamente, tem aplicação nos ambientes de simulações em laboratório, gerando rapidez na caracterização destes erros e na avaliação de seu efeito sobre protocolos de nível superior.

Os modelos matemáticos mais empregados para representar processos de erros com memória têm por base o emprego de cadeias Markov ocultas (HMM, de “Hidden Markov Models”) [2]. Na grande maioria dos casos, estes modelos são ajustados a dados empíricos ou de simulação através da estimação de máxima verossimilhança (ML, de “Maximum Likelihood”) de seus parâmetros, e o algoritmo Baum-Welch (BW) [4] é frequentemente utilizado para isso. Este algoritmo iterativo, apesar de garantir um maior valor da função de

verossimilhança a cada iteração, não é capaz de assegurar uma solução de máximo global [4]. Além disso, o esforço computacional despendido aumenta significativamente com o número de estados da cadeia de Markov, podendo em alguns casos inviabilizar a sua aplicação.

O estudo de algoritmos mais simples, com menor custo computacional e que tenham maiores possibilidades de encontrar uma solução de máxima verossimilhança global é de grande interesse nestes casos [5]. Um caminho que tem sido perseguido em vários trabalhos é a investigação do emprego de algoritmos de otimização bio-inspirados, e alguns trabalhos recentes têm destacado as potencialidades do algoritmo PSO (“Particle Swarm Optimization”) para esta aplicação [6]–[9]. Este algoritmo foi desenvolvido em 1995 por Eberhart e Kennedy [10], tendo por base o comportamento social e o movimento dinâmico dos insetos, pássaros e peixes, e desde então tem sido aplicado com sucesso a diferentes problemas de otimização não-linear [6]–[9], [11], [12].

No que diz respeito ao ajuste dos parâmetros de modelos HMM, o PSO é encontrado na literatura principalmente em aplicações ao problema de reconhecimento de voz [7], [13]. Em [7], os autores fazem uma comparação entre o desempenho dos algoritmos PSO e Baum-Welch na estimação dos parâmetros de um modelo HMM usado em reconhecimento de locutor. Seus resultados mostram que os obtidos com o algoritmo PSO apresentam maior valor da função de verossimilhança que os parâmetros estimados pelo algoritmo BW. No entanto, não é feita em [7] nenhuma avaliação de desempenho dos algoritmos em relação à quantidade de iterações e aos valores dos parâmetros do algoritmo PSO utilizados. Além disso, a comparação realizada não apresenta nenhum indicativo de esforço computacional.

Neste artigo se procura avaliar se o algoritmo PSO pode ser também vantajoso para a estimação de modelos de erros em surtos. Para isso, é feita uma comparação de desempenho, em termos de verossimilhança, entre os algoritmos PSO e BW na estimação dos parâmetros do modelo HMM para modelar canais com erros em surtos. Para isso, simulou-se um canal Rayleigh e um canal de satélite LEO. Uma análise do desempenho destes algoritmos com a variação da quantidade de iterações e dos parâmetros do algoritmo PSO utilizados também é realizada. Todas as comparações tem como base um mesmo esforço computacional.

O restante deste texto está organizado como se segue: a seção II apresenta alguns conceitos básicos sobre modelos HMM; as seções III e IV descrevem os algoritmos BW e PSO, respectivamente; a comparação dos algoritmos BW e PSO é feita na seção V e a seção VI conclui o trabalho.

N. Maciel, E. Marques e E. Pinto, Instituto Militar de Engenharia, Departamento de Engenharia Elétrica, Laboratório de Comunicações Digitais, Emails: {nilsonmpj,elaineckmarques,ernesto}@ime.eb.br. Este trabalho foi parcialmente apoiado pelo CNPQ.

## II. MODELOS HMM

Os modelos HMM são caracterizados por sua ordem, probabilidades condicionais de transição de estados e probabilidades condicionais de observações [9], [14]. Um modelo HMM de ordem  $N$  é composto pelos estados  $S = \{S_1, S_2, \dots, S_N\}$  e seus parâmetros são aqui representado por  $\lambda = \{A, B, \Pi\}$ , onde:

- $A = (a_{ij}), 1 \leq i, j, \leq N$  é a matriz das probabilidades condicionais de transição dos estados e  $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$  é a probabilidade condicional de que a cadeia de Markov se encontre no estado  $S_j$ , dado que se encontrava no estado  $S_i$  imediatamente antes. Os elementos da matriz  $A$  devem satisfazer as condições abaixo:

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

- $B = \{b_i(k)\}$  é a matriz de probabilidades condicionais de observação e  $b_i(k) = P(O_t = o_k | q_t = i)$  é a probabilidade condicional de que o  $k$ -ésimo elemento do alfabeto de observações tenha sido gerado pelo estado  $S_i$ . Os elementos da matriz  $B$  devem atender às condições abaixo:

$$0 \leq b_i(k) \leq 1; \sum_{k=1}^m b_i(k) = 1; \quad 0 \leq i \leq N; \quad (2)$$

- $\Pi = \{\pi_i\}$  é o vetor das probabilidades iniciais de cada estado e  $\pi_i = P(q_1 = i)$  é a probabilidade inicial do estado  $S_i$ . Os elementos do vetor  $\Pi$  devem respeitar as condições abaixo:

$$0 \leq \pi_i \leq 1; \sum_{i=1}^N \pi_i = 1; \quad 1 \leq i \leq N; \quad (3)$$

Para a modelagem de erros em surtos, há apenas duas observações: “1” e “0” que representam, respectivamente, presença e ausência de erro num dado bit recebido.

## III. ALGORITMO BAUM-WELCH (BW)

O algoritmo Baum-Welch é um procedimento iterativo para a estimação dos parâmetros  $\lambda$ , no qual, a cada iteração aumenta-se o valor da função de verossimilhança  $P(\bar{O} | \lambda)$  [4].

A cada iteração, o valor atual da trinca de parâmetros de  $\hat{\lambda}_k$  é usada para calcular as variáveis progressiva  $\alpha_{kt}$  (*forward*) e regressiva  $\beta_{kt}$  (*backward*) representadas nas equações (4) e (5), respectivamente. Em seguida é estimado  $\hat{\lambda}_{k+1}$  usando  $\alpha_{kt}$  e  $\beta_{kt}$ . Este procedimento é repetido até que se atinja o critério de parada desejado, por exemplo o número máximo de iterações determinado.

$$\alpha_{kt}(i) = P[O_1, O_2, \dots, O_t, s_t = i | \lambda] \quad (4)$$

$$\beta_{kt}(i) = P[O_{t+1}, O_{t+2}, \dots, O_T | s_t = i, \lambda] \quad (5)$$

## IV. PARTICLE SWARM OPTIMIZATION (PSO)

O PSO é um algoritmo de otimização de funções não-lineares que tem por base uma teoria sócio-cognitiva muito simples. Segundo ela, cada indivíduo de uma população é capaz de estimar a qualidade de sua experiência na busca de um determinado objetivo. Como os indivíduos são sociais, eles também possuem conhecimentos sobre como seus vizinhos se comportam tentando atingir o mesmo objetivo. A probabilidade de que um determinado indivíduo tome uma certa decisão será então função do seu desempenho no passado e do desempenho de seus vizinhos.

Cada solução consiste em um conjunto de parâmetros representando um ponto num espaço, em geral, multidimensional. A solução é chamada de “partícula” e o grupo de partículas é chamado de “enxame”.

Cada partícula  $i$  é representada por um vetor de posição  $\vec{x}_i(t)$  e tem uma velocidade instantânea  $\vec{v}_i(t)$  a ela associada.

Além disso, cada partícula guarda a posição  $\vec{p}_i$  na qual obteve o melhor valor da função custo. A cada iteração, é atualizado o valor de  $\vec{p}_g$ , correspondente à posição da partícula que obteve melhor valor para a função custo, e a velocidade de cada partícula é calculada como se segue:

$$\begin{aligned} \vec{v}_i(t+1) &= W \vec{v}_i(t) \\ &+ c_1 U_1 (\vec{p}_i(t) - \vec{x}_i(t)) \\ &+ c_2 U_2 (\vec{p}_g(t) - \vec{x}_i(t)) \end{aligned} \quad (6)$$

O parâmetro  $W$  é chamado peso de inércia.  $U_1$  e  $U_2$  são valores uniformemente distribuídos no intervalo  $[0 \ 1]$ . Os parâmetros  $c_1$  e  $c_2$  ponderam a influência da experiência própria passada e social, respectivamente.

A nova posição de cada partícula é dada por (7), onde  $\alpha$  é uma constante que pondera o efeito da velocidade na atualização da posição.

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \alpha \vec{v}_i(t+1) \quad (7)$$

As equações (6) e (7), assim como os valores de  $\vec{p}_i$  e  $\vec{p}_g$ , são calculados a cada iteração até que seja atingida uma condição de parada pré-estabelecida.

Quando o algoritmo PSO é utilizado na estimação dos parâmetros dos modelos HMM alguns cuidados devem ser tomados para atender às restrições apresentadas em (1), (2) e (3). Neste trabalho adotamos a modificação do algoritmo PSO apresentada em [15], na qual o valor de  $\alpha$  é escolhido de modo que os parâmetros dos modelos HMM atendam às suas restrições estocásticas.

## V. RESULTADOS

A função custo aqui empregada foi o logaritmo neperiano da função de verossimilhança ( $\ln P(\bar{O} | \lambda)$ ) de um modelo HMM com número de estados pré-estabelecido. Esta função será daqui por diante frequentemente denotada por LL (de “log-likelihood”).

Esta seção apresenta os resultados de avaliação de desempenho dos algoritmos BW e PSO, e está dividida em 5 sub-seções. Na primeira se procura avaliar as potencialidades dos algoritmos em obter soluções de máxima verossimilhança. Em

seguida, é feita uma avaliação da relação entre o desempenho do algoritmo PSO e o número de iterações requerido. Na subseção V-C compara-se os desempenhos dos dois algoritmos para um tempo de processamento pré-fixado. Na subseção seguinte é apresentada uma pequena avaliação do efeito dos parâmetros do algoritmo PSO sobre o seu desempenho. Em seguida, é apresentada uma nova comparação de desempenho dos algoritmos PSO e BW para uma outra aplicação.

#### A. Potencialidades dos algoritmos na obtenção de soluções ML

Para a investigação do desempenho dos algoritmos PSO e BW na estimação dos parâmetros dos modelos HMM utilizados em erros em surtos, gerou-se uma sequência de erros de tamanho  $10^6$  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, foi utilizado o 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. Especificamente, o experimento foi repetido com 50 e 30 iterações, sendo realizadas 2581 e 4324 inicializações, respectivamente.

Para apresentar os resultados são utilizadas curvas em que se tem como abscissa o percentual das inicializações (denotado por INI) que levou a valores de logaritmo da função de verossimilhança superiores à ordenada (denotado por LL). Curvas deste tipo serão denominadas distribuições de verossimilhanças ou distribuições de valores de verossimilhanças. Estas curvas são obtidas após diversas execuções dos algoritmos com inicializações diferentes geradas aleatoriamente sobre uma mesma amostra.

Percebe-se na Figura 1 o aumento da quantidade de inicializações que atinge valores mais elevados de LL à medida em que se aumenta o número de iterações do BW. Nota-se também que mesmo com 100 iterações a maioria das estimações usando o algoritmo BW 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 [7].

A Figura 2 apresenta as distribuições de verossimilhanças do algoritmo PSO para valores de  $W=0,2$  e  $W=0,7$ . 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 o máximo global. Percebe-se também na Figura 2 a influência dos parâmetros

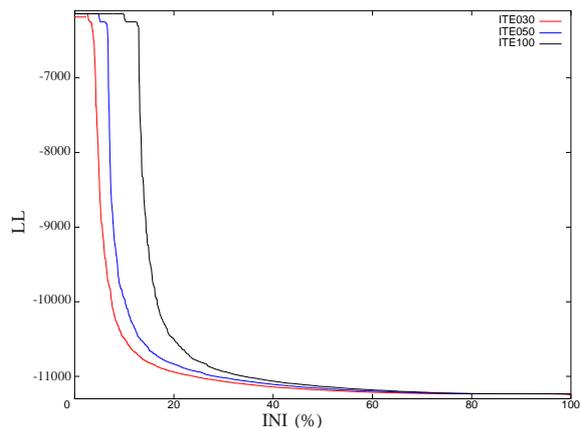


Fig. 1. Distribuições de verossimilhanças dos modelos HMM estimados pelo algoritmo BW, com 30 (ITE30), 50 (ITE50) e 100 (ITE100) iterações, e mesmo tempo de processamento.

do PSO nos valores de verossimilhança alcançados, tendo sido obtidos melhores valores de 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.

#### B. Influência da iteração no desempenho do PSO

Em seguida, avaliou-se a variação dos valores de LL de acordo com o número de iterações do algoritmo PSO com trinca de parâmetros  $(W \ c_1 \ c_2)$  fixada em  $(0,7 \ 2,0 \ 2,0)$ . Especificamente, foram feitas 472 inicializações e testados os valores de 200, 400, 600, 800 e 1000 iterações. As distribuições de valores de verossimilhanças são apresentadas na Figura 3.

Percebe-se nesta figura que com o aumento do número de iterações, aumenta-se a porcentagem de inicializações com altos valores de LL. Por exemplo, para 200 iterações (pior caso nesta figura), obteve-se pelo menos 32,2% das inicializações com valor de LL superior a  $-7000$  contra apenas 13% para as estimações usando o algoritmo BW.

Os maiores valores alcançados para o logaritmo da função de verossimilhança com o algoritmo BW foram  $-6118$ ,  $-6140$  e  $-6136$  para 100, 50 e 30 iterações, respectivamente. Com

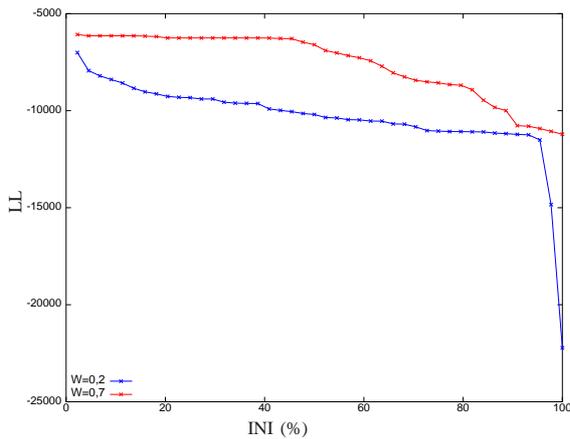


Fig. 2. Distribuições de valores de verossimilhanças dos modelos HMM estimados pelo algoritmo PSO com 30 partículas,  $c_1 = c_2 = 2, 0$  e valores de  $W$  iguais a 0, 2 e 0, 7.

apenas 110 iterações do PSO foi possível alcançar um valor de -6118 para o logaritmo da função de verossimilhança, e aumentando-se o número de iterações chegou-se a um valor de -6036.

Nota-se que, além de o algoritmo PSO ter alcançado maior valor de verossimilhança, a estimação dos parâmetros dos modelos HMM utilizando este algoritmo se caracteriza por realizar a “busca” pelo máximo global com menor sensibilidade à inicialização. Neste sentido, o algoritmo PSO parece ser, portanto, uma ferramenta melhor para realizar a estimação ML do que o algoritmo BW.

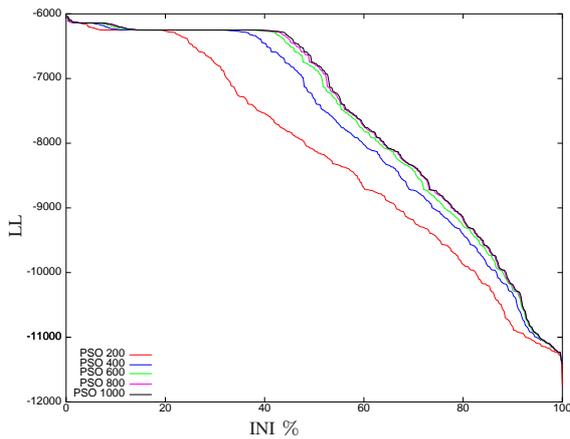


Fig. 3. Distribuições de verossimilhanças do algoritmo PSO com 30 partículas,  $W = 0, 7$  e  $c_1 = c_2 = 2, 0$  para 200, 400, 600, 800 e 1000 iterações.

C. Desempenho x Esforço computacional

Em seguida foi realizada uma comparação de desempenho para tempos de processamento pré fixados. Apesar das limitações da avaliação da carga de processamento do tempo

TABELA I

PERCENTUAL DE CASOS EM QUE O PSO PRODUZIU LL MAIOR QUE O BW.

| 2 minutos de simulação |      |       | 4 minutos de simulação |      |       |
|------------------------|------|-------|------------------------|------|-------|
| I_PSO                  | I_BW | % PSO | I_PSO                  | I_BW | % PSO |
| 10                     | 30   | 72,9  | 10                     | 30   | 70,1  |
| 10                     | 50   | 72,9  | 10                     | 50   | 70,1  |
| 10                     | 100  | 75,4  | 10                     | 100  | 70,1  |
| 20                     | 50   | 74,2  | 20                     | 50   | 70,3  |
| 20                     | 100  | 75,8  | 20                     | 100  | 72,9  |
| 40                     | 100  | 72,9  | 40                     | 100  | 72,9  |

TABELA II

PARÂMETROS TESTADOS.

| W     | c1    | c2    | W     | c1    | c2    |
|-------|-------|-------|-------|-------|-------|
| 0,200 | 1,400 | 1,400 | 0,700 | 2,000 | 2,000 |
| 0,200 | 2,000 | 2,000 | 0,729 | 1,494 | 1,494 |
| 0,700 | 1,200 | 2,000 | 1,000 | 1,200 | 2,000 |
| 0,700 | 1,400 | 1,400 | 1,000 | 2,000 | 1,200 |
| 0,700 | 2,000 | 1,200 | 1,000 | 2,000 | 2,000 |

de execução de um algoritmo, optou-se por utilizá-la como uma primeira abordagem para esta questão, tomando todos os cuidados possíveis na preparação dos programas de simulação para evitar distorções na avaliação comparativa entre os algoritmos PSO e BW.

Foram testadas diferentes quantidades de iterações dos algoritmos PSO ( $I\_PSO$ ) e BW ( $I\_BW$ ). Neste caso, cada experimento de simulação é composto por  $x$  inicializações do algoritmo BW e  $y$  inicializações do algoritmo PSO, sendo  $x$  e  $y$  escolhidos de modo a levar ao mesmo esforço computacional. A Tabela I mostra os resultados para 2 e 4 minutos de simulação apresentando os números de iterações utilizados, assim como o percentual das inicializações em que o PSO produziu maior valor de verossimilhança do que o algoritmo BW (% PSO).

Percebe-se nesta tabela que para todas as combinações de quantidades de iterações ( $I\_PSO$  e  $I\_BW$ ), o algoritmo PSO produziu maiores valores de verossimilhança, em percentuais bastante elevados das inicializações, o que evidencia a sua vantagem sobre o algoritmo BW.

D. Efeito da escolha dos parâmetros no desempenho do PSO

A fim de analisar os valores de verossimilhança com a mudança das trincas ( $W$   $c_1$   $c_2$ ) e da quantidade de partículas utilizada no algoritmo PSO, realizou-se 970 simulações utilizando-se as 10 trincas apresentadas na Tabela II, nas quais foram feitas 44, 33 e 20 inicializações independentes com 30, 60 e 100 partículas, respectivamente, para cada trinca. Estas quantidades de inicializações foram escolhidas de modo a manter o tempo de processamento inalterado.

A Tabela III apresenta os cinco melhores valores do logaritmo da função de verossimilhança (LL) alcançados, juntamente com os valores dos parâmetros e as quantidades de partículas utilizadas para obtê-los.

Nota-se nesta tabela que, neste caso, foram encontrados valores maiores de LL do que os obtidos com ( $W$   $c_1$   $c_2$ ) = (0,7 2,0 2,0), utilizado anteriormente.

Destaca-se que para aproximadamente 52% dos casos obteve-se valores maiores do que -7000 para o logaritmo da

TABELA III

 MELHORES TRINCAS ( $W, c_1, c_2$ ) COM AS QUANTIDADES DE PARTÍCULAS E LL CORRESPONDENTES.

|   | $W$ | $c_1$ | $c_2$ | P   | LL       |
|---|-----|-------|-------|-----|----------|
| 1 | 1.0 | 1.2   | 2.0   | 30  | -5861,84 |
| 2 | 1.0 | 2.0   | 2.0   | 60  | -5951,23 |
| 3 | 1.0 | 2.0   | 1.2   | 100 | -5966,83 |
| 4 | 1.0 | 2.0   | 2.0   | 100 | -5975,06 |
| 5 | 0.7 | 2.0   | 2.0   | 30  | -6036,80 |

TABELA IV

PERCENTUAL DE CASOS EM QUE O PSO PRODUZIU LL MAIOR QUE O BW.

| 2 minutos de simulação |      |       | 4 minutos de simulação |      |       |
|------------------------|------|-------|------------------------|------|-------|
| I_PSO                  | I_BW | % PSO | I_PSO                  | I_BW | % PSO |
| 10                     | 30   | 76,3  | 10                     | 30   | 83,0  |
| 10                     | 50   | 74,7  | 10                     | 50   | 79,7  |
| 10                     | 100  | 76,5  | 10                     | 100  | 78,9  |
| 20                     | 50   | 78,0  | 20                     | 50   | 83,8  |
| 20                     | 100  | 78,7  | 20                     | 100  | 82,5  |
| 40                     | 100  | 74,0  | 40                     | 100  | 81,9  |

função de verossimilhança, e em 92% os valores de LL obtidos foram superiores a -11000.

Esta tabela ilustra a importância de uma escolha criteriosa da trinca ( $W, c_1, c_2$ ) a fim de se obter uma melhor estimação dos parâmetros dos modelos HMM.

#### E. Aplicação a um Canal de Satélite LEO (Low Earth Orbit)

Nesta subseção é apresentada uma comparação de desempenho dos algoritmos PSO e BW, para tempos de processamento pré fixados, na estimação dos parâmetros do modelo HMM representando um canal de um satélite LEO. Para isso, gerou-se uma sequência de tamanho  $10^6$  produzida pelo modelo HMM de Fritchmann utilizado para representar um canal de um satélite LEO com ângulo de elevação igual a  $68^\circ$ , apresentado em [16]. Os parâmetros utilizados para o algoritmo PSO foram:  $W = 0,7$  e  $c_1 = c_2 = 2,0$ . Foram estimados os parâmetros para um modelo HMM com 3 estados.

Esta comparação é feita da mesma maneira como apresentada na subseção V-C. A Tabela IV mostra os resultados para 2 e 4 minutos de simulação apresentando os números de iterações utilizados, assim como o percentual das inicializações em que o PSO produziu maior valor de verossimilhança do que o algoritmo BW (%PSO).

Assim como na Tabela I, na Tabela IV também obteve-se maiores valores de verossimilhança para o algoritmo PSO, em percentuais bem mais elevados das inicializações, o que evidencia mais uma vez a vantagem do algoritmo PSO sobre o algoritmo BW.

## VI. CONCLUSÕES

Neste artigo se investigou o desempenho do algoritmo PSO na estimação de máxima verossimilhança dos parâmetros de modelos HMM aplicados a canais com erros em surtos.

Foi realizada uma comparação entre os desempenhos dos algoritmos PSO e BW na estimação de modelos HMM em

que se constatou que o algoritmo PSO produz resultados de verossimilhança significativamente melhores do que o algoritmo BW.

Além disso, com os resultados apresentados, ficou ilustrada a importância da escolha dos parâmetros do algoritmo PSO para se explorar adequadamente o seu potencial na otimização de máxima verossimilhança dos modelos HMM para erros em surtos. Em trabalhos futuros pretende-se atacar em maior profundidade esta questão.

Pretende-se também desenvolver algoritmos híbridos de estimação ML, nos quais a técnica PSO seja usada para “guiar” a busca do máximo global de um algoritmo computacionalmente simples com tendência a produzir soluções de máximos locais.

## REFERÊNCIAS

- [1] S. Tsai. Markov Characterization of the HF Channel. *IEEE Transactions on Communication Technology*, 17(1):24–32, 1969.
- [2] 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.
- [3] 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.
- [4] J. Bilmes. A Gentle Tutorial on the EM Algorithm Including Gaussian Mixtures and Baum-Welch. In *ICSI Technical Report TR-97-021*, 1997.
- [5] Fengqin Yang, Changhai Zhang, and Ge Bai. A Novel Genetic Algorithm Based on Tabu Search for HMM Optimization. In *Fourth International Conference on Natural Computation, 2008. ICNC '08.*, volume 4, pages 57–61, 2008.
- [6] S. Phon-Amnuaisuk. Estimating HMM Parameters Using Particle Swarm Optimisation. In *Proceedings of the EvoWorkshops 2009 on Applications of Evolutionary Computing*, 2009.
- [7] 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.
- [8] M. Maca, D. Novak, and L. Lhotska. Constraints in Particle Swarm Optimization of Hidden Markov Models. In *Intelligent Data Engineering and Automated Learning (IDEAL)*, volume 4224, pages 1399–1406, 2006.
- [9] M.S. Fernandes, E.L. Pinto, and M.A. Grivet. HMM Modelling of Burst Error Channels by Particle Swarm Optimization of the Likelihood Function. In *2010 International Telecommunications Symposium - ITS 2010*, volume 1, pages 1–5, 2010.
- [10] J. Kennedy and R. Eberhart. Particle Swarm Optimization. In *Proceedings IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4, 1995.
- [11] H.P.H. Anh and N.H. Phuc. Inverse neural mimo narx model identification of nonlinear system optimized with pso. In *Electronic Design, Test and Application, 2010. DELTA '10. Fifth IEEE International Symposium on*, pages 144–149, jan. 2010.
- [12] B. Majhi, G. Panda, and B. Mulgrew. Distributed identification of nonlinear processes using incremental and diffusion type pso algorithms. In *Evolutionary Computation, 2009. CEC '09. IEEE Congress on*, pages 2076–2082, may 2009.
- [13] S.M. Mirrezaie, K. Faez, A. Asnaashari, and A. Ziaei. A particle swarm optimization-based approach to speaker segmentation based on independent component analysis on gsm digital speech. In *Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE International Symposium on*, pages 502–507, dec. 2008.
- [14] 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.
- [15] E. Marques, N. Maciel, and E. Pinto. Tratamento de Restricoes Estocasticas no Algoritmo PSO para Estimacao ML de Modelos HMM. *Simpósio Brasileiro de Telecomunicacoes, SBTr'11*, 2011.
- [16] V. Y. Y. Chu and P. Sweeney. Channel modelling and error control strategies for the leo satellite channel. *Dept. of Electrical Engineering, The University of Surrey. Disponível:  $\frac{1}{2}$  vel: <http://www.ee.surrey.ac.uk/CCSR/Software/OPNET/chu-sweeney.pdf>, 2002.*