

ALGORITMO IPNLMS COM OTIMIZAÇÃO DO PARÂMETRO DE PROPORCIONALIDADE BASEADA EM BUSCA TABU PARA IDENTIFICAÇÃO DE PLANTAS ESPARSAS

CÉSAR A. S. C. BRANCO, FRANCISCO DAS C. DE SOUZA

Laboratório de Sistemas Adaptativos e Processamento de Sinais, Departamento de Engenharia Elétrica,
Universidade Federal do Maranhão

Av. dos Portugueses, 1966, CEP 65080-805, Bacanga, São Luís, MA, Brasil

E-mails: santana_castelo@hotmail.com, francisco.souza@ufma.br

Abstract— This work proposes a methodology to optimize the choice of the proportionality parameter of the IPNLMS (improved proportionate normalized least-mean-square) adaptive algorithm. The proposed approach uses a tabu search procedure to determine the optimal proportionality parameter for each iteration of the adaptation process of the IPNLMS algorithm. The objective function for the tabu search is based on the *a posteriori* output estimation error. Simulation results show that the convergence speed of the proposed methodology is faster than the IPNLMS algorithm for the identification of plants with high sparseness degree.

Keywords— IPNLMS algorithm, optimization, tabu search, sparse plant identification.

Resumo— Este trabalho propõe uma metodologia para otimizar a escolha do parâmetro de proporcionalidade do algoritmo adaptativo LMS normalizado proporcional melhorado (IPNLMS – *improved proportionate normalized least-mean-square*). A abordagem proposta usa um procedimento baseado em busca tabu para determinar o parâmetro de proporcionalidade ótimo em cada iteração do processo de adaptação do algoritmo IPNLMS. A função objetivo adotada para a busca tabu é baseada no erro de estimação de saída *a posteriori*. Resultados de simulações mostram que a velocidade de convergência da metodologia proposta é mais rápida do que a do algoritmo IPNLMS para a identificação de plantas com alto grau de esparsidade.

Palavras-chave— algoritmo IPNLMS, otimização, busca tabu, identificação de plantas esparsas.

1 Introdução

Filtros adaptativos são ferramentas fundamentais na resolução de vários problemas de processamento de sinais (Nascimento & Silva, 2014). Para ilustrar a importância destes filtros, a Figura 1(a) mostra um filtro adaptativo aplicado em um problema de identificação de sistemas (Chen, Gu & Hero III, 2009) e a Figura 1(b) mostra outro filtro adaptativo desta vez aplicado em um problema de cancelamento de eco em telecomunicações (Khong & Naylor, 2006). Os vetores $\mathbf{p}(n)$ e $\mathbf{w}(n)$ são as respostas impulsivas da planta e do filtro adaptativo, respectivamente. As variáveis $x(n)$, $d(n)$, $y(n)$, $e(n)$ e $z(n)$ são o sinal de entrada, as saídas da planta e do filtro, o sinal de erro e o ruído de medição, respectivamente. O filtro adaptativo usa o algoritmo adaptativo, A, para modificar $\mathbf{w}(n)$ a cada instante n . O objetivo dessas modificações é aproximar $e(n)$ de zero e, conseqüentemente, tornar $y(n)$ uma boa estimativa de $d(n)$.

Note que o cancelamento de eco também pode ser visto como um problema de identificação de sistemas (Farhang-Boroujeny, 2013). Neste caso, $x(n)$ é um sinal de voz proveniente de um usuário de longa distância do sistema de telecomunicações. Este sinal de voz é então recebido por um usuário local do sistema através de um dispositivo tal como um telefone celular, por exemplo. Então, o autofalante (AF) reproduz $x(n)$ que se propaga pelo ambiente onde o usuário local está, gerando eco. Este ambiente também é chamado de “caminho de eco” e pode ser modelado por uma planta cuja resposta impulsiva é $\mathbf{p}(n)$. Então, o eco $d(n)$ é captado e transmitido pelo microfone (MIC). Este eco deve ser cancelado para evitar que o usuário de longa distância escute sua própria voz enquanto estiver conversando com o usuário local. Note que o microfone também capta o ruído $z(n)$ do ambiente onde o usuário local está. O filtro adaptativo $\mathbf{w}(n)$ é usado para modelar a res-

posta impulsiva $\mathbf{p}(n)$ do ambiente que gera $d(n)$. Uma réplica $y(n)$ de $d(n)$ é então obtida e subtraída deste último. Resultados de aplicações mostram que apenas uma pequena parcela de $\mathbf{p}(n)$ é formada por pesos não-nulos (Loganathan, Khong & Naylor, 2008). Quando isto acontece, diz-se que a planta é esparsa.

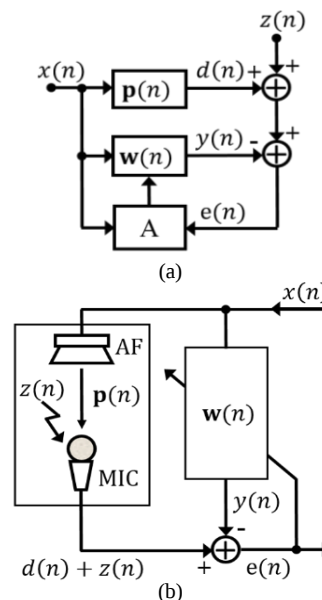


Figura 1. Exemplos de aplicação de um filtro adaptativo. (a) Identificação de sistemas. (b) Cancelamento de eco em telecomunicações.

As plantas esparsas são encontradas em diversas aplicações de engenharia, tais como estimação de harmônicas em sistemas elétricos de potência (Abdollahi, Zhang & Xue, 2013), processos químicos e sísmicos (Khong & Naylor, 2006), etc. Os algoritmos adaptativos *least-mean-square* (LMS) e LMS normalizado (NLMS – *normalized LMS*) apresentam desempenho pobre, em termos de velocidade de convergência, para ambientes esparsos (Widrow & Stearns, 1985) e (Huang, Benesty & Chen, 2006). Para solucionar este problema, foi desenvolvido o algo-

ritmo NLMS proporcional (PNLMS – *proportionate NLMS*). Este algoritmo distribui os ganhos de adaptação entre os pesos do filtro proporcionalmente à magnitude dos mesmos (Duttweiler, 2000), resultando em uma maior velocidade de convergência para plantas com alto grau de esparsidade. No entanto, o desempenho do PNLMS decai para plantas de esparsidade baixa (Benesty & Gay, 2002). Além disso, a rápida velocidade inicial de convergência deste algoritmo não é mantida em todo o processo de adaptação (Deng & Doroslovsky, 2006).

Versões aprimoradas do PNLMS, os algoritmos PNLMS++ (Gay, 1998) e PNLMS melhorado (IPNLMS – *improved PNLMS*) (Benesty & Gay, 2002) apresentam-se como opções para identificação de plantas de esparsidade média. Os algoritmos SC-PNLMS (Loganathan, Khong & Naylor, 2008) e SC-IPNLMS (Khong & Naylor, 2006) empregam estimativas da esparsidade da planta em suas respectivas regras de adaptação e são opções para plantas de esparsidade variável. Os algoritmos LMS com atração para zero (ZA-LMS) (Chen, Gu & Hero III, 2009) e LMS normalizado com detecção estocástica de taps ativos (ST-NLMS) (Li, Gu & Tang, 2006) são outros exemplos de algoritmos adaptativos utilizados para identificação de plantas esparsas.

O algoritmo IPNLMS foi criado com o objetivo de explorar melhor a filosofia proporcional originária do PNLMS. Apesar do desempenho satisfatório para plantas de esparsidade média, o IPNLMS não atinge a mesma velocidade inicial de convergência do PNLMS para plantas altamente esparsas (Deng & Doroslovsky, 2006). Além disso, seu desempenho depende de uma constante própria deste algoritmo denominada parâmetro de proporcionalidade. Resultados experimentais demonstram que 0,0, -0,5 e -0,75 são boas escolhas para tal parâmetro (Khong & Naylor, 2006).

Fatores como temperatura e pressão podem alterar a esparsidade de uma planta (Loganathan, Khong & Naylor, 2008). Um valor de parâmetro de proporcionalidade constante pode comprometer o desempenho do IPNLMS ao lidar com uma planta de esparsidade variável. Além disso, o valor deste parâmetro deve ser escolhido de acordo com o grau de esparsidade da planta. Dessa forma, um valor inadequado para tal parâmetro pode degradar o desempenho do IPNLMS.

Neste trabalho propõe-se uma metodologia para otimizar a escolha do parâmetro de proporcionalidade do algoritmo IPNLMS, usando um procedimento baseado em busca tabu para determinar o valor ótimo deste parâmetro para cada iteração do processo de adaptação. A função objetivo a ser minimizada é baseada no erro de estimação de saída *a posteriori* (Sayed, 2003). O algoritmo IPNLMS proposto é aplicado em um problema de identificação de plantas esparsas. Resultados de simulações mostram que a metodologia proposta tem maior velocidade de convergência e resposta rápida a perturbações na planta em relação ao algoritmo IPNLMS original.

2 Revisitando o Algoritmo IPNLMS

Esta seção descreve em detalhes o algoritmo adaptativo IPNLMS. Primeiramente, a estrutura geral dos algoritmos adaptativos proporcionais é apresentada. Logo após, é mostrado o algoritmo PNLMS padrão seguido de um procedimento para obter o IPNLMS a partir do PNLMS padrão.

2.1 Estrutura Geral dos Algoritmos Adaptativos Proporcionais

A regra de adaptação dos algoritmos adaptativos proporcionais é dada por (Souza, Seara & Morgan, 2012)

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mu \mathbf{G}(n) e(n) \mathbf{x}(n)}{\mathbf{x}^T \mathbf{G}(n) \mathbf{x}(n) + \varepsilon} \quad (1)$$

onde $\mathbf{x}(n) = [x(n) \ x(n-1) \ \dots \ x(n-N+1)]^T$ e $\mathbf{w}(n) = [w_1(n) \ w_2(n) \ \dots \ w_N(n)]^T$ são os vetores de entrada e de pesos do filtro adaptativo, respectivamente, $0 < \mu < 2$ é o parâmetro de passo e $\varepsilon > 0$ é um parâmetro de regularização responsável por estabilizar a solução. O sobrescrito T denota transposta do vetor ou matriz em questão e N é a quantidade de pesos do filtro. A matriz

$$\mathbf{G}(n) = \text{diag}[g_1(n) \ g_2(n) \ \dots \ g_N(n)] \quad (2)$$

de ordem $N \times N$ é responsável pela distribuição dos ganhos individuais dos pesos do filtro adaptativo. A regra de distribuição destes ganhos varia de acordo com o algoritmo considerado. O sinal de erro é calculado por

$$e(n) = d(n) - y(n) + z(n) \quad (3)$$

onde $d(n)$ é o sinal desejado, $y(n) = \mathbf{x}^T(n) \mathbf{w}(n)$ é a saída do filtro adaptativo e $z(n)$ é um ruído de medição de variância σ_z^2 .

2.2 Algoritmo PNLMS Padrão

O ganho individual que controla o ajuste do i -ésimo peso do algoritmo PNLMS padrão é obtido a partir de

$$g_i(n) = \frac{\phi_i(n)}{\sum_{j=1}^N \phi_j(n)} \quad (4)$$

onde

$$\phi_i(n) = \max[f(n), |w_i(n)|] \quad (5)$$

e

$$f(n) = \rho \max[\xi, \|\mathbf{w}(n)\|_\infty]. \quad (6)$$

A função de proporcionalidade, $\phi_i(n)$, de (5) determina o valor do i -ésimo ganho, $g_i(n)$, conforme a magnitude do respectivo i -ésimo peso, $w_i(n)$. O fator de ativação, $f(n)$, de (6) influencia diretamente na adaptação dos pesos considerados inativos (Souza, Tobias, Seara & Morgan, 2010). Tal fator depende da norma infinita do vetor dos pesos do filtro, $\|\mathbf{w}(n)\|_\infty$, e dos parâmetros de inicialização, ξ , e de proporcionalidade, ρ (Duttweiler, 2000).

2.3 Algoritmo IPNLMS

O algoritmo IPNLMS pode ser obtido a partir do PNLMS padrão (Souza, 2012). Para tal, em (6), substitui-se $\|\mathbf{w}(n)\|_\infty$ por $\|\mathbf{w}(n)\|_1$ (norma-1 do vetor de pesos do filtro) e adota-se $\rho = 1/N$. Assim, tem-se um novo fator de ativação dado por

$$f(n) = \begin{cases} \frac{1}{N} \max[\xi, \|\mathbf{w}(n)\|_1], & n = 0 \\ \frac{\|\mathbf{w}(n)\|_1}{N}, & n \geq 1 \end{cases} \quad (7)$$

Desta forma, para $n \geq 1$, a equação (5) pode ser reescrita como sendo

$$\phi_i(n) = \max \left[\frac{\|\mathbf{w}(n)\|_1}{N}, |w_i(n)| \right]. \quad (8)$$

Agora, para a função de proporcionalidade dada por (8), uma média ponderada é usada no lugar do operador $\max[\cdot]$. Logo, a nova função de proporcionalidade é (Benesty & Gay, 2002)

$$\phi_i(n) = (1 - \alpha) \frac{\|\mathbf{w}(n)\|_1}{N} + (1 + \alpha) |w_i(n)| \quad (9)$$

onde $-1 \leq \alpha < 1$ é um fator de ponderação. Note que a primeira parcela de (9) é comum a todos os pesos do filtro, enquanto que a segunda é proporcional à magnitude do i -ésimo peso.

Agora, considerando (9), desenvolve-se o denominador de (4) para obter

$$\begin{aligned} \sum_{i=1}^N \phi_i(n) &= \sum_{i=1}^N \left[(1 - \alpha) \frac{\|\mathbf{w}(n)\|_1}{N} + (1 + \alpha) |w_i(n)| \right] \\ &= \left[\frac{1 - \alpha}{N} \sum_{i=1}^N \|\mathbf{w}(n)\|_1 \right] + \left[(1 + \alpha) \sum_{i=1}^N |w_i(n)| \right] \\ &= (1 - \alpha) \|\mathbf{w}(n)\|_1 + (1 + \alpha) \|\mathbf{w}(n)\|_1 \\ &= 2 \|\mathbf{w}(n)\|_1. \end{aligned} \quad (10)$$

Então, substituindo (9) e (10) em (4) obtém-se o ganho individual do algoritmo IPNLMS como sendo

$$g_i(n) = (1 - \alpha) \frac{1}{2N} + (1 + \alpha) \frac{|w_i(n)|}{2 \|\mathbf{w}(n)\|_1 + \varsigma} \quad (11)$$

onde $\varsigma > 0$ é um parâmetro de regularização usado para evitar divisão por zero. O fator α é denominado parâmetro de proporcionalidade.

A regra de atualização dos pesos do algoritmo IPNLMS é dada por (1), com os elementos de $\mathbf{G}(n)$ sendo calculados a partir de (11). O algoritmo IPNLMS é então dado pelas equações (1), (2), (3) e (11). A seguir, a metodologia proposta para otimizar a escolha do parâmetro de proporcionalidade do algoritmo IPNLMS é apresentada.

3 Metodologia Proposta

Neste trabalho é proposto um algoritmo IPNLMS que utiliza a busca tabu para determinar o valor ótimo do parâmetro de proporcionalidade em cada iteração do processo de adaptação. A função objetivo a ser minimizada pela busca tabu é baseada no erro de estimação *a posteriori* (Sayed, 2003)

$$e_p(n) = d(n) - \mathbf{x}^T(n) \mathbf{w}(n+1) + z(n). \quad (12)$$

Note que (12) é obtida substituindo, em (3), $\mathbf{w}(n)$ por $\mathbf{w}(n+1)$. Este último é obtido a partir de (1), (2) e (3). Então, considerando (12), a função objetivo adotada é dada por

$$\psi(n) = 10 \log_{10} \{ [e_p(n)]^2 \} \quad (13)$$

onde $\psi(n)$ de (13) é o erro quadrático *a posteriori* em dB. Assim, o problema de otimização a ser solucionado pelo método da busca tabu, em cada iteração do algoritmo IPNLMS proposto, é

$$\alpha_{\text{opt}}(n) = \arg \min_{-1 \leq \alpha(n) < 1} \psi(n) \quad (14)$$

sendo $\alpha_{\text{opt}}(n)$ o parâmetro de proporcionalidade ótimo na n -ésima iteração do processo de adaptação do algoritmo IPNLMS proposto. Note, de (1) e (11), que α influencia diretamente no valor de $\mathbf{w}(n+1)$ e este, por sua vez, impacta no valor de (13).

Esta seção descreve a metodologia proposta para determinar $\alpha_{\text{opt}}(n)$ em cada iteração do processo de adaptação do algoritmo IPNLMS. Primeiramente, é feito um estudo sobre o efeito do parâmetro α no desempenho do IPNLMS. Logo após, o método da busca tabu é apresentado em detalhes. Por fim, o algoritmo IPNLMS com busca de $\alpha_{\text{opt}}(n)$ é obtido.

3.1 Impacto do Parâmetro de Proporcionalidade no Desempenho do Algoritmo IPNLMS

Nesta seção é estudado o efeito de α no desempenho do algoritmo adaptativo IPNLMS, considerando um problema de identificação de sistemas. Para tal, são realizadas simulações de Monte Carlo (média de 100 realizações independentes). Os valores das variáveis avaliadas são obtidas por (Fishman, 1996), (Manolakis, Ingle & Kogon, 2005) e (Rubstein, 2008)

$$\begin{aligned} E\{V(n)\} &\cong \frac{V_1(n) + V_2(n) + \dots + V_R(n)}{R} \\ &= \frac{1}{R} \sum_{r=1}^R V_r(n) \end{aligned} \quad (15)$$

sendo $E\{\cdot\}$ o operador valor esperado, $V_r(n)$ é o valor da variável avaliada V na n -ésima iteração da r -ésima realização e R é o número de realizações. O cenário de todas as simulações consiste de uma planta esparsa com $N = 100$ pesos (Souza, Tobias, Seara & Morgan, 2010). Nesta planta, os pesos ativos (ou não-nulos) estão localizados nas posições $\{1, 30, 35, 85\}$, e seus respectivos valores são $\{0, 1, 1, 0, -0, 5, 0, 1\}$. A medida de esparsidade desta planta é $S(\mathbf{p}) = 0,9435$, conforme a definição (Huang, Benesty & Chen, 2006)

$$S(\mathbf{p}) = \frac{N}{\sqrt{N} - N} \left(1 - \frac{\|\mathbf{p}\|_1}{\sqrt{N} \|\mathbf{p}\|_2} \right) \quad (16)$$

onde $\mathbf{p} = [p_1 \ p_2 \ \dots \ p_N]^T$ é o vetor da resposta impulsiva da planta. Como entrada foi utilizado um sinal correlacionado auto regressivo de ordem 2, AR(2), de variância unitária dado por

$$x(n) = 0,4x(n-1) - 0,4x(n-2) + v(n) \quad (17)$$

sendo $v(n)$ um ruído branco gaussiano de variância $\sigma_v^2 = 0,77$. A dispersão da matriz de autocorrelação da entrada é $\chi = 10$. O ruído de medição, $z(n)$, é branco gaussiano com variância $\sigma_z^2 = 10^{-3}$ (SNR = 30 dB). A figura de mérito utilizada para avaliar o desempenho do algoritmo IPNLMS é o desalinhamento normalizado em dB dado por

$$\eta(n) = 10 \log_{10} \left(\frac{\|\mathbf{p} - \mathbf{w}(n)\|_2^2}{\|\mathbf{p}\|_2^2} \right). \quad (18)$$

Com o intuito de ilustrar o impacto de α no algoritmo IPNLMS, a Figura 2 mostra o desempenho do IPNLMS para a identificação da planta esparsa, $\mathbf{p}$,

considerando diversos valores para α que vão de -1 até $0,99$. O parâmetro de passo adotado é $\mu = 0,5$.

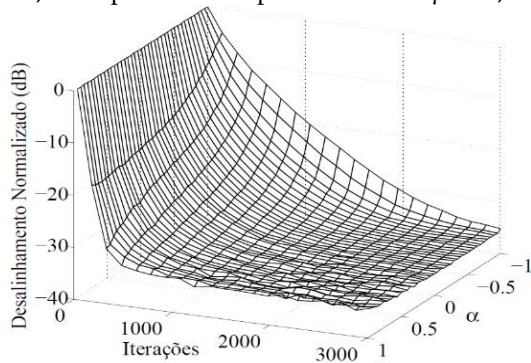


Figura 2. Desempenho do algoritmo IPNLMS para a identificação de uma planta de esparsidade $S(\mathbf{p}) = 0,9435$, considerando diversos valores para α que vão de -1 a $0,99$ e passo $\mu = 0,5$.

Note da Figura 2 que o valor escolhido para o parâmetro de proporcionalidade impacta diretamente na velocidade de convergência durante todo o processo de adaptação do algoritmo IPNLMS.

Neste trabalho propõe-se um procedimento para determinar o valor ótimo de α para cada iteração do processo de adaptação do algoritmo IPNLMS. Tal procedimento é baseado no método da busca tabu, o qual é apresentado a seguir em detalhes.

3.2 Método da Busca Tabu

A busca tabu é uma meta-heurística que aplica uma técnica de busca local com movimentos simples e mecanismos de memória que evitam a estagnação em ótimos locais (Glover, 1989). Tendo suas origens no final da década de 70, na resolução de problemas da área de engenharia da computação (Glover, 1990), a busca tabu tem ampliado sua área de aplicação nas últimas décadas, sendo também empregada em áreas que vão da microeletrônica (Zuo, Murray & Smith, 2014) aos sistemas elétricos de potência (Junior, Cossi, Contreras & Mantovani, 2013).

O ponto de partida de uma busca tabu é uma solução inicial escolhida aleatoriamente. Então, a partir desta solução inicial e dos movimentos adotados, cria-se um conjunto aleatório de soluções vizinhas. Estas soluções vizinhas são então comparadas com a solução inicial para determinar a melhor solução do ciclo atual da busca. Então, inicia-se um novo ciclo e cria-se um novo conjunto aleatório de soluções vizinhas a partir da melhor solução do ciclo anterior. As novas soluções vizinhas são então comparadas com a melhor solução do ciclo anterior para determinar a melhor do ciclo atual. Este processo se repete até que o critério de convergência seja satisfeito.

Para prevenir repetições, a busca tabu possui uma ferramenta de memória de curto prazo chamada de lista tabu. Nesta lista são armazenados movimentos realizados recentemente, tornando-os proibidos de serem executados durante um dado intervalo de ciclos denominado período tabu. Outra ferramenta chave é o critério de aspiração, o qual anula a condição tabu de determinado movimento caso este leve a uma solução que traga ganhos significativos para os objetivos da busca. Conforme as características do problema, a busca tabu também pode utilizar outras

ferramentas para aprimorar a busca, tais como mecanismos de memória de longo prazo, lista de candidatos, dentre outros (Glover & Laguna, 1997).

Para a metodologia proposta, considere o conjunto V^k das soluções vizinhas e a melhor solução $\alpha^k(n)$ do ciclo atual k da busca tabu. O conjunto V^k pode ser representado por um vetor de M números decimais, cada um contendo um possível valor de $\alpha^k(n)$. Cada solução vizinha $v_i^k \in V^k$ ($i = 1, 2, \dots, M$) é um número decimal com D casas decimais. Um exemplo de V^k , formado por M possíveis $\alpha^k(n)$ com duas casas decimais cada ($D = 2$), é mostrado na Figura 3.

v_1^k	v_2^k	v_3^k	...	v_{M-1}^k	v_M^k
0,45	-0,36	0,98	...	0,00	-0,22

Figura 3. Exemplo de um conjunto V^k de soluções vizinhas.

O movimento utilizado para criar as novas soluções vizinhas consiste de uma soma. Esta soma tem como parcelas a melhor solução do ciclo anterior, $\alpha^{k-1}(n)$, e um número escolhido aleatoriamente de um conjunto finito H de números decimais, com D casas cada, contidos em um intervalo $[L, U]$. Como exemplo, escolhe-se $L = -0,2$, $U = 0,2$ e $D = 1$. Então, $H = \{-0,2, -0,1, 0, 0,1, 0,2\}$. Considerando $\alpha^{k-1}(n) = 0,7$, duas possíveis novas soluções vizinhas são $v_j^k = 0,7 + 0,1 = 0,8$ e $v_l^k = 0,7 + (-0,2) = 0,5$, com $1 \leq j, l \leq M$.

A estrutura adotada para a lista tabu é a de uma matriz cujo número de linhas é igual ao total de possíveis $-1 \leq \alpha(n) < 1$, conforme H , D , L e U adotados. Já o número de colunas desta matriz é igual ao total de elementos de H . Os movimentos “tabu” são registrados armazenando-se o valor “ k + período tabu” no elemento correspondente à i -ésima linha [i -ésimo possível $\alpha(n)$] e à j -ésima coluna [j -ésimo possível incremento ou decremento] da lista tabu.

Em relação ao critério de aspiração adotado, considere uma dada solução vizinha v_i^k criada a partir de um movimento “tabu”. Assim, se

$$\psi[v_i^k] < \psi[v_j^k], \text{ com } 1 \leq i, j \leq M \text{ e } i \neq j \quad (19)$$

e

$$\psi[v_i^k] < \psi[\alpha^{k-1}(n)] \quad (20)$$

então v_i^k tem sua condição “tabu” anulada e passa a ser a melhor solução do ciclo atual k , ou seja, $\alpha^k(n) = v_i^k$.

Para o critério de convergência, considera-se que a busca tabu convergiu se uma mesma solução permanecer como a melhor solução atual durante três ciclos consecutivos. Em outras palavras, caso tenha-se no ciclo atual k

$$\alpha^k(n) = \alpha^{k-1}(n) = \alpha^{k-2}(n) \quad (21)$$

diz-se que o critério de convergência é satisfeito, ou seja, $\alpha_{\text{opt}}(n) = \alpha^k(n)$. A seguir, o algoritmo IPNLMS com parâmetro α otimizado é obtido.

3.3 Algoritmo IPNLMS Proposto

O algoritmo IPNLMS proposto é descrito a seguir em detalhes:

1. Inicialize o vetor de pesos do filtro ($n = 0$)

$$\mathbf{w}(0) = \mathbf{0}.$$

2. Obtenha os dados de entrada e saída
 $d(n) = \mathbf{x}^T(n)\mathbf{p}(n)$, $y(n) = \mathbf{x}^T(n)\mathbf{w}(n)$ e $z(n)$.
3. Calcule o sinal de erro
 $e(n) = d(n) - y(n) + z(n)$.
4. Atualize dos pesos do filtro adaptativo.
Nesta etapa, obtenha
$$\alpha_{\text{opt}}(n) = \arg \min_{-1 \leq \alpha(n) < 1} \psi(n)$$
onde
$$\psi(n) = 10 \log_{10} \{ [d(n) - \mathbf{x}^T(n)\mathbf{w}(n+1) + z(n)]^2 \}.$$
Para tal, utilize o seguinte procedimento baseado no método da busca tabu:
 - i) Faça $k = 0$, escolha uma solução inicial aleatória, $-1 \leq \alpha^0(n) < 1$ e, a partir de (13), calcule $\psi[\alpha^0(n)]$;
 - ii) Faça $k = k + 1$, gere V^k a partir de $\alpha^{k-1}(n)$ e verifique se $v_i^k \in V^k$ ($i = 1, 2, \dots, M$) é “tabu”;
 - iii) A partir de (13), calcule $\psi[v_i^k]$ ($i = 1, 2, \dots, M$);
 - iv) Se existe v_i^k tal que
$$\psi[v_i^k] < \psi[v_j^k], \text{ com } 1 \leq i, j \leq M \text{ e } i \neq j$$
e
$$\psi[v_i^k] < \psi[\alpha^{k-1}(n)]$$
faça $\alpha^k(n) = v_i^k$. Caso contrário, faça $\alpha^k(n) = \alpha^{k-1}(n)$;
 - v) Atualize a lista tabu;
 - vi) Se o critério de convergência de (21) for satisfeito, vá para vii. Caso contrário, retorne a ii;
 - vii) Retorne $\alpha_{\text{opt}}(n) = \alpha^k(n)$. Em seguida, calcule $\mathbf{w}(n+1)$ usando
$$g_l(n) = [1 - \alpha_{\text{opt}}(n)] \frac{1}{2N} + [1 + \alpha_{\text{opt}}(n)] \frac{|w_l(n)|}{2\|\mathbf{w}(n)\|_1 + \varepsilon}$$
5. Vá para a iteração seguinte ($n = n + 1$) e repita os passos 2, 3 e 4 até o fim do processo de adaptação.

A seguir, o desempenho do algoritmo IPNLMS proposto é avaliado.

4 Simulação e Resultados

Para avaliar o algoritmo IPNLMS proposto, referido deste ponto em diante como TS-IPNLMS (*tabu search* IPNLMS), foram realizadas simulações de Monte Carlo (média de 100 realizações independentes) para um problema de identificação de sistemas. O desempenho do algoritmo TS-IPNLMS é comparado com o do algoritmo IPNLMS. O cenário das simulações é o mesmo apresentado na seção 3. A figura de mérito adotada para mostrar o comportamento dos algoritmos considerados é o desalinhamento normalizado em dB dado por (18).

A seguir, dois exemplos comparativos são mostrados. O primeiro considera uma inversão, enquanto que o segundo leva em conta um deslocamento na resposta impulsiva da planta esparsa apresentada na seção 3. Para a execução da busca tabu, em todos os exemplos considera-se $D = 2$, $L = -2,0$, $U = 2,0$, $M = 25$ e um período tabu igual a três ciclos.

4.1 Exemplo 1

Este exemplo tem o objetivo de comparar o desempenho dos algoritmos IPNLMS e TS-IPNLMS

em termos de velocidade de convergência e resposta a uma perturbação ocorrida no instante $n = 2000$, quando o vetor de pesos da resposta impulsiva da planta muda de $\mathbf{p}$ para $-\mathbf{p}$.

A Figura 4 mostra o desalinhamento normalizado em dB dos algoritmos considerados e o comportamento do α ótimo do TS-IPNLMS. O parâmetro de passo adotado é $\mu = 0,5$. Para o algoritmo IPNLMS considera-se $\alpha = 0,0$. Note das curvas da Figura 4 que o TS-IPNLMS alcança a mais rápida velocidade de convergência, mesmo após ocorrer a inversão na resposta impulsiva da planta. Note também que o α ótimo inicia em $-0,006$, muda para $0,975$ logo na segunda iteração e, em menos de 20 iterações, mantém-se estável em torno de $0,976$ até o final da simulação, mesmo após a inversão.

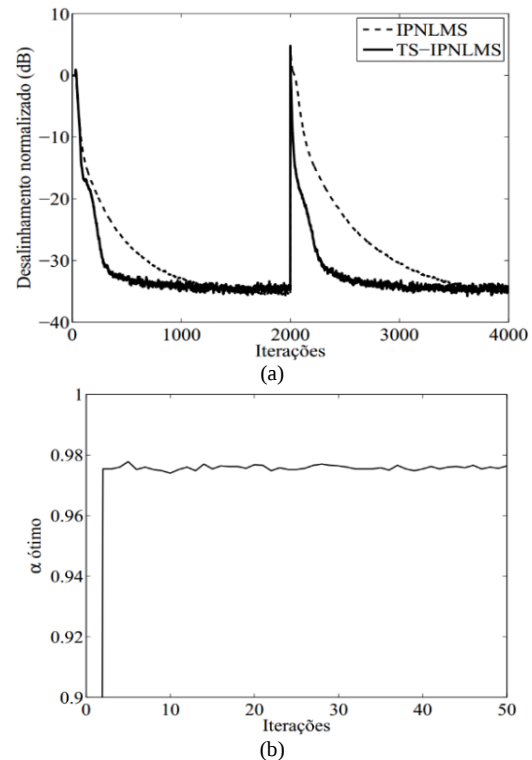


Figura 4. Comportamento dos algoritmos IPNLMS e TS-IPNLMS considerando uma inversão na resposta da planta em $n = 2000$. (a) Desalinhamento normalizado em dB dos algoritmos IPNLMS (com $\alpha = 0,0$) e TS-IPNLMS. (b) Parâmetro α ótimo do TS-IPNLMS.

4.2 Exemplo 2

Neste exemplo os algoritmos IPNLMS, e TS-IPNLMS são novamente comparados em termos de velocidade de convergência e resposta a uma perturbação na resposta impulsiva da planta. Desta vez, o vetor $\mathbf{p}$ é deslocado de 12 amostras para a direita no instante $n = 2000$. Assim, os pesos ativos de $\mathbf{p}$, cujos valores são iguais a $\{0,1, 1,0, -0,5, 0,1\}$, são movidos das posições $\{1, 30, 35, 85\}$ para as posições $\{13, 42, 47, 97\}$, respectivamente.

A Figura 5 mostra o desalinhamento normalizado em dB dos algoritmos considerados e o comportamento do α ótimo do TS-IPNLMS. O parâmetro de passo adotado é $\mu = 0,5$. Para o algoritmo IPNLMS considera-se $\alpha = 0,0$. Note das curvas da Figura 5 que o TS-IPNLMS alcança novamente a mais rápida velocidade de convergência mesmo após ocorrer o

deslocamento na resposta impulsiva da planta. Note também que o α ótimo inicia-se em 0,009, muda-se para 0,970 na iteração seguinte e, em menos de 20 iterações, estabiliza-se em torno de 0,971 até o final da simulação, mesmo após o deslocamento.

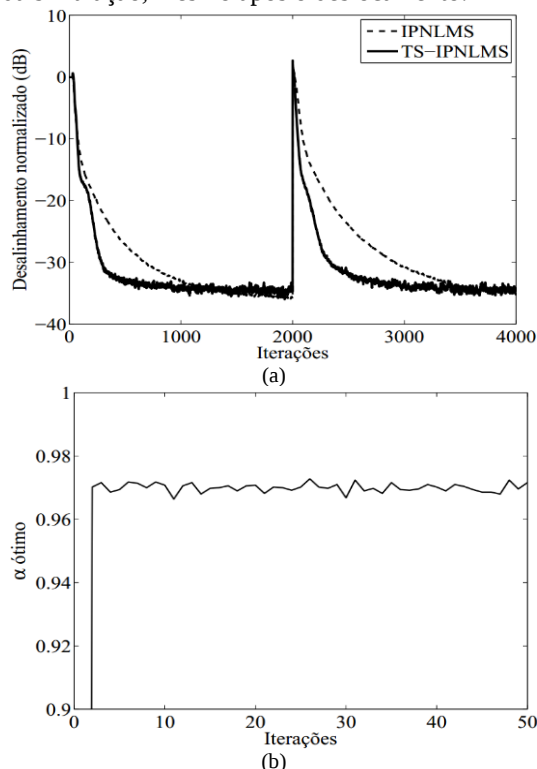


Figura 5. Comportamento dos algoritmos IPNLMS e TS-IPNLMS considerando um deslocamento na resposta da planta de 12 amostras para a direita em $n = 2000$. (a) Desalinhamento normalizado em dB dos algoritmos IPNLMS (com $\alpha = 0,0$) e TS-IPNLMS. (b) Parâmetro α ótimo do TS-IPNLMS.

5 Considerações Finais

Este trabalho propõe uma metodologia para otimizar a escolha do parâmetro de proporcionalidade do algoritmo adaptativo IPNLMS. Para tal, utiliza-se um método de otimização chamado busca tabu. Resultados de simulações demonstram que a abordagem proposta apresenta desempenho superior, em termos de velocidade de convergência e resposta a perturbações, em relação ao algoritmo IPNLMS original para identificação de plantas esparsas. Os resultados obtidos mostram também que o valor escolhido para o parâmetro de proporcionalidade afeta diretamente a velocidade de convergência durante todo o processo de adaptação do algoritmo IPNLMS.

Agradecimentos

Os autores agradecem a Fundação de Amparo à Pesquisa e Desenvolvimento Científico do Maranhão (FAPEMA), a Secretaria da Ciência, Tecnologia e Inovação (SECTI), o Governo do Estado do Maranhão e a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo apoio ao desenvolvimento do projeto de pesquisa.

Referências Bibliográficas

Abdohalli, A., Zhang, P., Xue, H. and Li, S. (2013) Enhanced Subspace-Least Mean Square for Fast and Accurate Power System Measurement. *IEEE Transactions on Power Delivery*, Vol 28, No 1, pp. 383-393.

- Benesty, J. and Gay, S. L. (2002). An improved PNLMS algorithm. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2002*, Vol 2, pp. 1881-1884.
- Chen, Y., Gu, Y. and Hero III, A. O. (2009). Sparse LMS for System Identification. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2010, Taipei, Taiwan, pp. 3125-3128.
- Deng, H., Doroslovacký, M. (2006). Proportionate Adaptive Algorithms for Network Echo Cancellation. *IEEE Transactions on Signal Processing*, Vol 54, No 5, pp. 1794-1803.
- Duttweiler, D. (2000). Proportionate normalized least-mean-squares adaptation in echo cancelers. *IEEE Transactions on Speech and Audio Processing*, Vol 8, No 5, pp.508-518.
- Farhang-Boroujeny, B. (2013). *Adaptive Filters: theory and Applications* 2nd ed. Wiley & Sons.
- Fishman, G. S. (1996). *Monte Carlo: concepts, algorithms and applications*. Springer-Verlag New York.
- Gay, S. L. (1998). Na efficient, fast converging adaptive filter for network echo cancellation. *XXXII Asilomar Conference on Signals, Systems and Computers*, Monterey, USA, Vol 1, pp. 394-398.
- Glover, F. (1989). Tabu Search Part I. *ORSA Journal on Computing*, Vol 1, No 3, pp. 190-206.
- Glover, F. (1990). Tabu Search Part II. *ORSA Journal on Computing*, Vol 2, No 1, pp. 4-32.
- Glover, F. and Laguna, M. (1997) *Tabu Search*. Springer, USA.
- Haykin, S. (2002) *Adaptive Filter Theory* 4th ed. Prentice Hall, New Jersey, EUA.
- Junior, B. R. P., Cossi, A. M., Contreras, J. and Mantovani, J. R. S. (2013). Multiobjective multistage distribution system planning using tabu search. *IET Generation, Transmission & Distribution*, Vol 8, Issue 1, pp. 35-45.
- Khong, W. H. and Naylor, P. A. (2006) Efficient Use of Sparse Adaptive Filters. *Asilomar Conference of Signals and Systems Computation*, October 2006, Pacific Grove, CA, pp. 1375-1379.
- Li, Y., Gu, Y. and Tang, K. (2006). Parallel NLMS Filters with Stochastic Active Taps and Step-Sizes for Sparse System Identification. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2006*, Toulouse, France, Vol 3, pp. 109-112.
- Loganathan, P., Khong, A. W. H. and Naylor, P. A. (2008). A Sparseness Controlled Proportionate Algorithm for Acoustic Echo Cancellation. *European Signal Processing Conference (EUSIPCO)* August 2008, Lausanne, Switzerland.
- Manolakis, D. G., Ingle, V. K. and Kogon, S. M. (2005). *Statistical and Adaptive Signal Processing*. Artech House.
- Nascimento, V. H. and Silva, M. T. M. (2014). Chapter 12 – Adaptive Filters. *Academic Press Library in Signal Processing*, Vol 1, pp. 619-671.
- Rubinstein, R. Y. (2008). *Simulation and the Monte Carlo Method* 2nd ed. Wiley.
- Sayed, H. A. (2003). *Fundamentals of Adaptive Filtering*. Wiley & Sons.
- Souza, F. C. (2012). *Algoritmos Adaptativos LMS Normalizados Proporcionais: Proposta de um Novo Algoritmo e sua Modelagem Estocástica*. Tese: Doutorado. Universidade Federal de Santa Catarina (UFSC).
- Souza, F. C., Tobias, O. J., Seara, R. and Morgan, D. R. (2010). A PNLMS Algorithm With Individual Activation Factors. *IEEE Transactions on Signal Processing*, Vol 58, No 4, pp. 2036-2047.
- Widrow, B. and Stearns (1985) *Adaptive Signal Processing*. Prentice Hall, New Jersey, EUA.
- Y. Huang, J. Benesty and J. Chen (2006) *Acoustic MIMO Signal Processing*. Springer, New York, EUA.
- Zhao, H. and Zhang, J. (2009). Aptively Combined FIR and Functional Link Artificial Neural Network Equalizer for Nonlinear Communication Channel. *IEEE Transactions on Neural Networks*, Vol 20, No 4, pp. 665-674.
- Zuo, X., Murray, C. C. and Smith, A. E. (2014). Solving an Extended Double Row Layout Problem Using Multiobjective Tabu Search and Linear Programing. *IEEE Transactions on Automation Science and Engineering*, Vol 11, No 4, pp. 1122-1132.