

# Um algoritmo do Módulo Constante Acelerado para equalização espaço-temporal

Magno T. M. da Silva<sup>1</sup>, Max Gerken e Maria D. Miranda<sup>2</sup>

**Resumo**—A separação eficiente de sinais é um problema freqüente em sistemas de comunicação multiusuário. Técnicas autodidatas de equalização espaço-temporal vem sendo estudadas como uma possível solução. O algoritmo ACM (*Accelerated Constant Modulus*), proposto recentemente como uma técnica de equalização autodidata de sistemas monousuário, é baseado num algoritmo utilizado em controle adaptativo que utiliza a segunda derivada das estimativas dos coeficientes e minimiza a função custo de Godard. Neste trabalho é apresentada sua extensão para separação autodidata e equalização de múltiplos usuários na presença de interferência intersimbólica. Resultados de simulações evidenciam que, às custas de um moderado aumento da complexidade computacional, o algoritmo proposto apresenta uma maior velocidade de convergência que o conhecido algoritmo MU-CM (*Multi-user Constant Modulus*).

**Palavras-Chave**—Equalização espaço-temporal autodidata, múltiplo acesso, algoritmo do Módulo Constante Acelerado.

**Abstract**—The efficient separation of signals is a frequent problem in multi-user communications systems. Space-time blind equalization techniques have been studied as a possible solution. The Accelerated Constant Modulus algorithm was recently proposed as a blind equalization technique for combating inter-symbol interference in single-user systems. It is based on a tuner used in adaptive control that adjusts the second derivative of the coefficient estimates and minimizes the cost function introduced by Godard. In this paper, we extend this algorithm for blind separation and equalization of multiple users in the presence of inter-symbol interference. Through simulations, we show that the proposed algorithm outperforms the well-known MU-CM (*Multi-user Constant Modulus*) algorithm at the cost of introducing a moderate increase in computational complexity.

**Keywords**—Space-time blind equalization, multi-access, Accelerated Constant Modulus algorithm.

## I. INTRODUÇÃO

Em um ambiente com múltiplos usuários, o problema da recuperação simultânea de todos os sinais tem se tornado de grande importância em sistemas CDMA (*Code Division Multiple Access*) e SDMA (*Space Division Multiple Access*) [8], [10]. Nestes casos, é conveniente uma atualização autodidata para melhorar a eficiência da transmissão de dados pois não são necessários sinais de treinamento para atualizar equalizadores adaptativos ou redes de antenas [11].

No caso monousuário, dois algoritmos para equalização autodidata que se destacam na literatura são o algoritmo CM (*Constant Modulus*) [3] e o algoritmo SW (*Shalvi-Weinstein*)

[2]. Entretanto, eles possuem um compromisso desfavorável entre velocidade de convergência e complexidade computacional. Em [1] foi apresentado o algoritmo ACM (*Accelerated Constant Modulus*) para combater interferência intersimbólica de forma autodidata em sistemas monousuário. Este algoritmo é baseado na discretização de um algoritmo de tempo contínuo utilizado em controle adaptativo e na minimização da função custo de Godard [3]. Através de resultados de simulações, verificou-se que ele possui um compromisso mais favorável entre velocidade de convergência e complexidade computacional que os algoritmos CM e SW, apresentando um desempenho melhor que o do algoritmo CM e um comportamento próximo ao do algoritmo SW em muitas situações.

No contexto da equalização espaço-temporal, foi proposto em [6], [7] e [9] o algoritmo MU-CM (*Multi-user Constant Modulus*), uma extensão do algoritmo de Godard, com capacidade de separar os sinais de diferentes usuários. Esta capacidade é devida a utilização de um termo que penaliza as correlações cruzadas das saídas dos equalizadores. Uma extensão do algoritmo Super-Exponencial para sistemas com múltiplos usuários também foi considerada em [4]. Como no caso monousuário, estes algoritmos apresentam um compromisso desfavorável entre complexidade computacional e velocidade de convergência. O algoritmo MU-CM, embora tenha uma complexidade computacional baixa, apresenta uma convergência lenta podendo inclusive ficar “encalhado” em mínimos locais. Por outro lado, a complexidade computacional do algoritmo Super-Exponencial inviabiliza sua utilização na prática.

Estes resultados motivaram a extensão do algoritmo ACM para o caso multiusuário. Como o algoritmo ACM apresenta um desempenho melhor que o algoritmo CM às custas de um moderado aumento da complexidade computacional, espera-se que para equalização espaço-temporal, o algoritmo MU-ACM (*Multi-user ACM*) apresente também um comportamento mais favorável que o algoritmo MU-CM.

O artigo é organizado da seguinte forma. Após apresentar um modelo do sistema multiusuário considerado, o algoritmo MU-CM é revisto na Seção III. Na Seção IV é feita uma revisão do algoritmo ACM para o caso monousuário e a extensão para sua utilização no contexto espaço-temporal é considerada na Seção V. Resultados de simulações comparando os dois algoritmos e conclusões são mostrados respectivamente nas seções VI e VII.

## II. UM MODELO DO SISTEMA MULTIUSUÁRIO

No sistema de equalização espaço-temporal utilizado considera-se a transmissão de sinais i.i.d. de  $N$  usuários

Magno T. M. da Silva está com o Departamento Engenharia de Telecomunicações e Controle da Escola Politécnica da USP, São Paulo, SP, Brasil. Maria D. Miranda está com a Universidade Presbiteriana Mackenzie, São Paulo, SP, Brasil. E-mails: <sup>1</sup>magno@lcs.poli.usp.br, <sup>2</sup>mdm@mackenzie.com.br. Este trabalho foi financiado pela <sup>1</sup>FAPESP - (00/12350-6) e <sup>2</sup>Mackpesquisa.

síncronos através de canais de comunicação lineares. No receptor há uma rede de  $L$  ( $L \geq N$ ) antenas que recebe misturas ruidosas dos sinais transmitidos possuindo interferências do tipo co-canal (CCI - *Co-Channel Interference*) e intersimbólica (ISI - *Inter-symbol Interference*). Supondo conhecido o número de usuários, o equalizador é formado por  $N$  filtros adaptativos do tipo FIR (*Finite Impulse Response*) com  $M$  coeficientes cada. As saídas  $y_i$ ,  $i = 1, \dots, N$ , são usadas para atualizar estes coeficientes por meio de um algoritmo autodidata. Na Figura 1 é mostrado um modelo deste sistema considerando  $N = 2$  usuários e  $L = 3$  antenas. Nesta figura, os sinais  $a_1(n)$  e  $a_2(n)$  são transmitidos por dois usuários distintos. No receptor há três antenas que recebem as misturas  $u_1(n)$ ,  $u_2(n)$  e  $u_3(n)$ , sendo  $\eta_1(n)$ ,  $\eta_2(n)$  e  $\eta_3(n)$  sinais de ruído considerado branco, gaussiano e de média zero. O equalizador é formado por dois filtros  $\mathbf{w}_1$  e  $\mathbf{w}_2$  do tipo FIR cujas saídas são denotadas respectivamente por  $y_1(n)$  e  $y_2(n)$ .

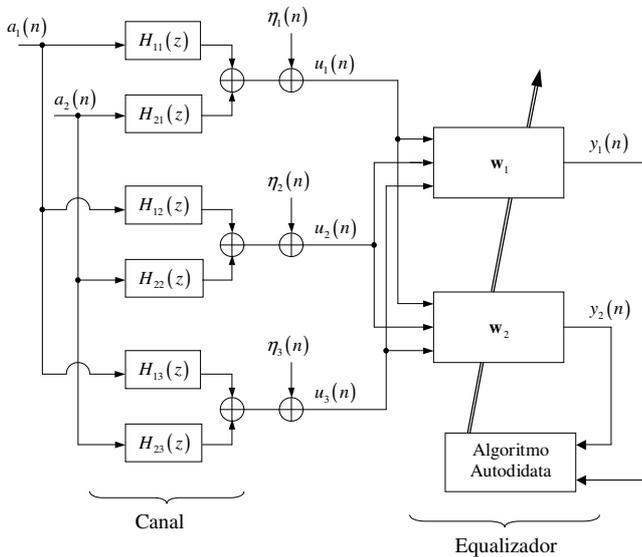


Fig. 1. Modelo de um sistema de equalização espaço-temporal, considerando  $N = 2$  usuários e  $L = 3$  antenas.

Considerando diversidade temporal e espacial, o vetor de entrada de cada filtro do equalizador pode ser representado como

$$\mathbf{u}^T(n) = [\mathbf{u}_1^T(n) \quad \mathbf{u}_2^T(n) \quad \dots \quad \mathbf{u}_L^T(n)], \quad (1)$$

sendo

$$\mathbf{u}_p(n) = [u_p(n) \quad u_p(n-1) \quad \dots \quad u_p(n-K_t+1)]^T, \quad (2)$$

com  $p = 1, 2, \dots, L$  e  $K_t$  um inteiro associado à diversidade temporal.

O mapeamento entre a entrada e a saída do canal na ausência de ruído pode ser representado como

$$\mathbf{u}(n) = \underbrace{\begin{bmatrix} \mathcal{H}_{11}^T & \mathcal{H}_{21}^T & \dots & \mathcal{H}_{N1}^T \\ \mathcal{H}_{12}^T & \mathcal{H}_{22}^T & \dots & \mathcal{H}_{N2}^T \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{H}_{1L}^T & \mathcal{H}_{2L}^T & \dots & \mathcal{H}_{NL}^T \end{bmatrix}}_{\mathcal{H}^T} \underbrace{\begin{bmatrix} \mathbf{a}_1(n) \\ \mathbf{a}_2(n) \\ \vdots \\ \mathbf{a}_N(n) \end{bmatrix}}_{\mathbf{a}(n)}, \quad (3)$$

sendo  $\mathbf{a}_p(n) = [a_p(n) \quad a_p(n-1) \quad \dots \quad a_p(n-K_t-K_c+2)]^T$ ,  $K_c - 1$  a ordem de cada canal e  $\mathcal{H}_{ij}$  a matriz de convolução do canal  $H_{ij}(z)$  que relaciona o usuário  $i$  à antena  $j$ .

Os sinais de saída dos filtros do equalizador podem ser representados como

$$y_i(n) = \mathbf{w}_i^T(n-1)\mathbf{u}(n), \quad i = 1, \dots, N, \quad (4)$$

ou ainda

$$y_i(n) = \mathbf{s}_i^T(n-1)\mathbf{a}(n), \quad (5)$$

sendo que  $\mathbf{s}_i(n) = \mathcal{H}\mathbf{w}_i(n)$  representa a resposta combinada do usuário  $i$  com  $i = 1, 2, \dots, N$ . A medida de desempenho baseada na interferência intersimbólica usada no caso monousuário pode ser estendida para sistemas multiusuário [5] como se segue

$$\text{IR}(\mathbf{s}_i) = \frac{\sum_k |s_k|^2}{|s_{k_o}|^2} - 1, \quad (6)$$

sendo  $s_{k_o}$  o componente de  $\mathbf{s}_i$  que possui o máximo valor absoluto. Esta medida de desempenho indica a quantidade de interferência residual (IR) do usuário  $i$  depois do equalizador. A equalização perfeita é obtida se a condição ótima de *zero-forcing* for satisfeita simultaneamente para as  $N$  saídas do equalizador.

### III. O ALGORITMO MU-CM

Uma extensão direta da função custo de Godard para o caso multiusuário é dada por [7]

$$E \left\{ \sum_{i=1}^N (|y_i(n)|^2 - R_2^a)^2 \right\}.$$

As saídas de um equalizador otimizado por esta função podem convergir para apenas um dos diferentes sinais transmitidos e geralmente converge para o de maior potência, ficando difícil prever qual dos sinais será recuperado [7]. Para evitar que a mesma fonte seja extraída em diferentes receptores, é usual a introdução de um termo que penaliza as correlações cruzadas entre as saídas dos filtros espaciais. Desta maneira, a extensão da função custo de Godard assume a forma [6], [7], [9]

$$J_G(n) = E \left\{ \sum_{i=1}^N (|y_i(n)|^2 - R_2^a)^2 \right\} + \frac{\xi}{2} \sum_{i=1}^N \sum_{j=1, j \neq i}^N |E \{ y_i(n) y_j^*(n) \}|^2, \quad (7)$$

sendo  $\xi$  o peso do termo de correlação cruzada. No segundo termo do lado direito desta equação também pode ser considerado o espalhamento de atraso (*delay spread*) do canal [7].

O vetor gradiente da Equação (7) em relação ao usuário  $i$  é dado por

$$\nabla_{\mathbf{w}_i} J_G(n) = (\beta - 1) \left( 2E \{ e_i(n) \mathbf{u}^*(n) \} + \xi \sum_{j=1, i \neq j}^N E \{ y_j(n) \mathbf{u}^*(n) \} E \{ y_i(n) y_j^*(n) \} \right), \quad (8)$$

sendo

$$e_i(n) = (|y_i(n)|^2 - R_2^a)y_i(n) \quad (9)$$

e  $\beta = 2$  ( $= 3$ ) para o caso complexo (real).

Um algoritmo do tipo *Steepest Descent* pode ser obtido com a equação de atualização dos coeficientes dada por

$$\mathbf{w}_i(n) = \mathbf{w}_i(n-1) - \mu \nabla_{\mathbf{w}_i} J_G(n), \quad i = 1, 2, \dots, N, \quad (10)$$

sendo  $\mu$  o passo de adaptação. Para se obter o algoritmo MU-CM os gradientes da expressão anterior devem ser substituídos por estimativas que podem ser feitas utilizando janelas retangulares ou exponenciais [7]. Neste trabalho, o primeiro termo do lado direito da Equação (8) foi estimado instantaneamente enquanto as demais esperanças foram estimadas usando janelas exponenciais com um fator de esquecimento  $0 < f_e < 1$ .

#### IV. O ALGORITMO ACM

Em [1] foi apresentado um algoritmo baseado na discretização de um algoritmo de tempo contínuo utilizado em controle adaptativo [12], [13] descrito pelas seguintes equações:

$$\dot{\mathbf{w}}(t) = \mathbf{q}(t) \quad (11)$$

$$\dot{\mathbf{q}}(t) = -\mathbf{M}_1 \mathbf{u}^*(t) e(t) + \\ -2\mathbf{M}_1 (\mathbf{M}_2 + \mathbf{u}^*(t) \mathbf{u}^T(t) \mathbf{M}_1 \mathbf{M}_3) \mathbf{q}(t) \quad (12)$$

$$e(t) = \mathbf{u}^T(t) \mathbf{w}(t) - d(t), \quad (13)$$

sendo  $\mathbf{u}(t)$  o sinal de entrada (vetor regressor),  $\mathbf{w}(t)$  o vetor de coeficientes,  $d(t)$  o sinal desejado,  $e(t)$  o erro de predição e  $\mathbf{q}(t)$  a derivada do vetor de coeficientes. As matrizes  $\mathbf{M}_1$ ,  $\mathbf{M}_2$  e  $\mathbf{M}_3$  são simétricas e positivas definidas. Conforme [12] e [13], este sistema é estável segundo o critério de Lyapunov quando estas matrizes satisfizerem as condições

$$4\mathbf{M}_1 \mathbf{M}_3 \mathbf{M}_1 \mathbf{M}_2 \geq \mathbf{I} \quad (14)$$

$$\mathbf{M}_2 \mathbf{M}_1 \mathbf{M}_3 + \mathbf{M}_1 \mathbf{M}_3 \mathbf{M}_2 \geq \mathbf{M}_1^{-1}/2. \quad (15)$$

Este algoritmo foi chamado de acelerador pois é obtido a partir da segunda derivada (“aceleração”) dos coeficientes.

Uma versão de tempo discreto e autodidata deste algoritmo foi obtida através dos seguintes passos [1]:

- Aplicação do método de Euler reverso para discretizar as equações (11), (12) e (13);
- Introdução de um erro *a priori* definido como

$$e_a(n) = \varphi(\mathbf{u}^T(n) \mathbf{w}(n-1)), \quad (16)$$

sendo  $\varphi(\cdot)$  uma função não-linear baseada na função custo de Godard [3], ou seja

$$\varphi(y) = (|y|^2 - R_2^a)y; \quad (17)$$

- Utilização de uma aproximação linear de  $\varphi(\cdot)$  descrita por

$$\varphi(\mathbf{u}^T \mathbf{w}) \approx \varphi(\mathbf{u}^T \mathbf{w}_v) + \left[ \frac{\partial \varphi(\mathbf{u}^T \mathbf{w})}{\partial \mathbf{w}} \Big|_{\mathbf{w}=\mathbf{w}_v} \right]^T (\mathbf{w} - \mathbf{w}_v) \\ \approx \varphi(\mathbf{u}^T \mathbf{w}_v) + g(\mathbf{u}^T \mathbf{w}_v) \mathbf{u}^T (\mathbf{w} - \mathbf{w}_v), \quad (18)$$

sendo

$$g(y) = \beta |y|^2 - R_2^a; \quad (19)$$

- Consideração de  $\mathbf{M}_i = m_i \mathbf{I}$ ,  $i = 1, 2, 3$ , sendo  $m_i$  constantes positivas.

A partir destes passos chegou-se a uma versão do algoritmo que possui  $6M+6$  multiplicações, 1 divisão e  $5M+4$  adições. Observe que esta complexidade computacional fica entre as dos algoritmos CM e SW, esta última sendo proporcional a  $M^2$ . As operações do algoritmo ACM são apresentadas na Tabela I.

TABELA I  
ALGORITMO ACM.

<p>Inicialização do algoritmo:  <math>R_2^a = E[ a(n) ^4] / E[ a(n) ^2]</math>  <math>\mathbf{w}(0) = [0 \ \dots \ 0 \ 1 \ 0 \ \dots \ 0]^T</math>, <math>\mathbf{q}(0) = \mathbf{0}</math>  <math>\alpha = 1 + 2\mu m_1 m_2</math>  <math>\beta = 2</math> (<math>= 3</math>) para o caso complexo (real)          Para cada instante de tempo, <math>n = 1, 2, \dots</math>, calcule:  <math>y(n) = \mathbf{u}^T(n) \mathbf{w}(n-1)</math>  <math>e_a(n) = ( y(n) ^2 - R_2^a)y(n)</math>  <math>b(n) = 2\mu m_1^2 m_3 + \mu^2 m_1 (\beta  y(n) ^2 - R_2^a)</math>  <math>c(n) = \frac{b(n) \mathbf{u}^T(n) \mathbf{q}(n-1) + \mu m_1 \alpha e_a(n)}{\alpha + b(n) \ \mathbf{u}(n)\ ^2}</math>  <math>\mathbf{q}(n) = \frac{1}{\alpha} [\mathbf{q}(n-1) - c(n) \mathbf{u}^*(n)]</math>  <math>\mathbf{w}(n) = \mathbf{w}(n-1) + \mu \mathbf{q}(n)</math></p>
---

#### V. O ALGORITMO MU-ACM

Seguindo os passos de obtenção do algoritmo ACM apresentados na seção anterior, sua extensão direta para o caso multiusuário pode ser feita considerando-se a função custo da Equação (7) e uma versão espaço-temporal do algoritmo “acelerador” de tempo contínuo. Entretanto, este procedimento exige uma grande quantidade de manipulações algébricas para se obter uma versão simplificada do algoritmo no contexto espaço-temporal.

Por outro lado, observando que a atualização dos coeficientes do algoritmo ACM é dada por

$$\mathbf{w}(n) = \mathbf{w}(n-1) - \mu (-\mathbf{q}(n)), \quad (20)$$

nota-se que ele pode ser interpretado como um algoritmo do gradiente estocástico. Observando a Tabela I, o termo que atualiza o vetor  $\mathbf{q}(n)$  é dado por  $c(n) \mathbf{u}^*(n)$ , ou seja

$$\frac{b(n) \mathbf{u}^T(n) \mathbf{q}(n-1) \mathbf{u}^*(n) + \overbrace{\mu m_1 \alpha e_a(n) \mathbf{u}^*(n)}^{(I)}}{\alpha + b(n) \|\mathbf{u}(n)\|^2}.$$

O termo (I) =  $e_a(n) \mathbf{u}^*(n) = \widehat{\nabla}$  nesta atualização é uma estimativa instantânea do vetor gradiente da função custo de Godard [3] para o caso monousuário. Considerando uma função custo qualquer, pode-se substituir a estimativa instantânea do gradiente de Godard pela estimativa do gradiente

desta função custo. Com esta aproximação e a função custo de Gordard para o caso multiusuário dada pela Equação (7), chega-se a uma versão multiusuário do algoritmo ACM, cujas equações são mostradas na Tabela II. Note que o vetor gradiente da Equação (8) pode ser estimado da mesma forma que no algoritmo MU-CM.

Uma das dificuldades deste tipo de algoritmo é o ajuste dos parâmetros  $\mu$ ,  $m_1$ ,  $m_2$  e  $m_3$  a fim de se obter uma alta velocidade de convergência dentro dos limites de estabilidade. De [12] e [13], sabe-se que a velocidade de convergência do algoritmo "acelerador" para filtragem adaptativa supervisionada é máxima quando os parâmetros estão no limite de estabilidade ( $4m_1^2m_2m_3 = 1$ ). Devido a este fato, introduziu-se o parâmetro  $\gamma$  tal que  $m_1m_2 = (2\gamma)^{-1}$  e  $m_1m_3 = \gamma/2$  garantindo  $4m_1^2m_2m_3 = 1$ . Conseqüentemente, é necessário escolher três parâmetros positivos  $\mu$ ,  $\gamma$  e  $m_1$ . Para os algoritmos ACM e MU-ACM, como ainda não foi feita uma análise de estabilidade, adotou-se a mesma sistemática na escolha dos parâmetros. Resultados de simulações sugerem que escolhendo  $m_1$  próximo de 0,16 e  $\gamma$  próximo de 100, deve-se variar o parâmetro  $\mu$  no intervalo ]0, 1[ para obter o melhor desempenho do algoritmo. Com esta sistemática os algoritmos ACM e MU-ACM sempre apresentam um comportamento adequado.

TABELA II  
ALGORITMO MU-ACM.

<p>Inicialização do algoritmo:  <math>R_2 = E [ a(n) ^4] / E [ a(n) ^2]^2</math>, <math>\alpha = 1 + 2\mu m_1 m_2</math>  <math>\mathbf{w}_i(0) = [0 \ \dots \ 0 \ 1 \ 0 \ \dots \ 0]^T</math>, <math>i = 1, 2, \dots, N</math>  <math>\mathbf{q}_i(0) = \mathbf{0}</math>, <math>i = 1, 2, \dots, N</math>,  <math>\beta = 2 (= 3)</math> para o caso complexo (real), <math>0 &lt; f_e &lt; 1</math>                      Para cada instante de tempo, <math>n = 1, 2, \dots</math>, calcule:                      Para <math>i = 1, 2, \dots, N</math>, calcule:  <math>y_i(n) = \mathbf{w}_i^T(n-1)\mathbf{u}(n)</math>  <math>e_i(n) = ( y_i(n) ^2 - R_2^a)y_i(n)</math>  <math>b_i(n) = 2\mu m_1^2 m_3 + \mu^2 m_1 (\beta  y_i(n) ^2 - R_2^a)</math>                      Para <math>j = 1, 2, \dots, N</math>, <math>j \neq i</math>, calcule:  <math>E \{y_i y_j^*\}_n = f_e E \{y_i y_j^*\}_{n-1} + (1 - f_e) y_i(n) y_j^*(n)</math>.  <math>E \{y_j \mathbf{u}^*\}_n = f_e E \{y_j \mathbf{u}^*\}_{n-1} + (1 - f_e) y_j(n) \mathbf{u}^*(n)</math>.                      Para <math>i = 1, 2, \dots, N</math>, calcule:  <math>\hat{\nabla}_{\mathbf{w}_i} J_G(n) = 2e_i(n)\mathbf{u}^*(n) +</math>  <math>+ \xi \sum_{j=1, j \neq i}^N E \{y_j \mathbf{u}^*\}_n E \{y_i y_j^*\}_n</math>  <math>\mathbf{c}_i(n) = \frac{b_i(n)\mathbf{u}^T(n)\mathbf{q}_i(n-1)\mathbf{u}^*(n) + \mu m_1 \alpha \hat{\nabla}_{\mathbf{w}_i} J_G(n)}{\alpha + b_i(n)\ \mathbf{u}(n)\ ^2}</math>  <math>\mathbf{q}_i(n) = \frac{1}{\alpha} (\mathbf{q}_i(n-1) - \mathbf{c}_i(n))</math>  <math>\mathbf{w}_i(n) = \mathbf{w}_i(n-1) + \mu \mathbf{q}_i(n)</math> }                 </p>
---

## VI. RESULTADOS EXPERIMENTAIS

Nas simulações foi considerado um sistema com  $N = 2$  usuários e  $L = 3$  antenas. Os modelos dos canais  $H_{ij}(z) = h_0 + h_1 z^{-1} + h_2 z^{-2}$ ,  $i = 1, \dots, N$ ,  $j = 1, \dots, L$ , utilizados estão mostrados na Tabela III. O Canal 1 foi escolhido aleatoriamente, sendo que apenas o canal  $H_{13}(z)$  é de fase mínima. O Canal 2 foi escolhido para que cada canal  $H_{ij}$  tivesse zeros dentro da circunferência unitária, exceto o canal  $H_{22}(z)$  que é variante no tempo. Os coeficientes  $h_0(n)$ ,  $h_1(n)$  e  $h_2(n)$  deste canal são obtidos da filtragem de um ruído branco gaussiano por um filtro de Butterworth de 2ª ordem que é projetado para, operando a uma taxa de 2400 amostras/segundo, simular uma taxa de desvanecimento de 0,1 Hz [14].

TABELA III  
MODELOS DOS CANAIS DE COMUNICAÇÃO USADOS NAS SIMULAÇÕES.

$H_{ij}$ do Canal 1			
$ij$	$h_0$	$h_1$	$h_2$
11	+0,43 + j0,36	+0,46 + j0,53	-0,30 - j0,33
21	+0,21 + j0,15	-0,26 - j0,91	+0,16 - j0,08
12	-0,37 + j0,20	-0,87 - j0,12	+0,15 - j0,20
22	+0,55 - j0,25	-0,61 + j0,06	-0,09 - j0,50
13	+0,61 + j0,66	+0,12 - j0,31	-0,24 - j0,17
23	+0,43 + j0,29	-0,52 - j0,58	+0,31 + j0,15
$H_{ij}$ do Canal 2			
$ij$	$h_0$	$h_1$	$h_2$
11	-0,50	+0,48	-0,03
21	+0,15	-0,03	+0,10
12	-0,26	-0,44	+0,19
22	$h_0(n)$	$h_1(n)$	$h_2(n)$
13	+1,00	-1,00	+0,41
23	+1,00	-1,60	+0,68

Os comportamentos dos algoritmos MU-ACM e MU-CM durante a convergência são mostrados na Figura 2 através de curvas de interferência residual (IR) para cada um dos usuários. Estas curvas correspondem a uma média de 50 realizações considerando modulação 4-QAM e o Canal 1 sem ruído. Neste caso o equalizador é formado por dois vetores com  $M = 15$  coeficientes cada, inicializados com apenas dois elementos não nulos (iguais a um) nas posições 5 e 7 respectivamente. Pode-se observar nesta figura que o algoritmo MU-ACM apresenta uma velocidade de convergência maior que a do algoritmo MU-CM para as duas saídas do equalizador. Os algoritmos foram ajustados para que atingissem o mesmo patamar de interferência como pode ser visto na Figura 2-b. No caso da Saída 1 (Figura 2-a) o algoritmo MU-CM converge para um mínimo local enquanto o MU-ACM converge para o mesmo patamar de interferência da Saída 2. Na Figura 3 são mostradas as constelações de saída dos equalizadores para uma realização considerando as  $10^3$  últimas iterações dos algoritmos. Passando os sinais de saída por decisores e comparando com os sinais transmitidos, verifica-se que, a menos de uma rotação de fase, os sinais transmitidos são recuperados sendo que a recuperação obtida pelo algoritmo MU-CM é mais lenta que a do MU-ACM. Para a Saída 1 foi observado que neste caso o algoritmo MU-CM falha na presença de ruído. Na

Tabela IV são mostrados os números de operações por iteração dos dois algoritmos aplicados a um sistema com  $N = 2$  usuários. Como pode ser observado nesta tabela, o algoritmo MU-ACM apresenta um moderado aumento da complexidade computacional em relação ao algoritmo MU-CM mantendo-a proporcional a  $M$  operações.

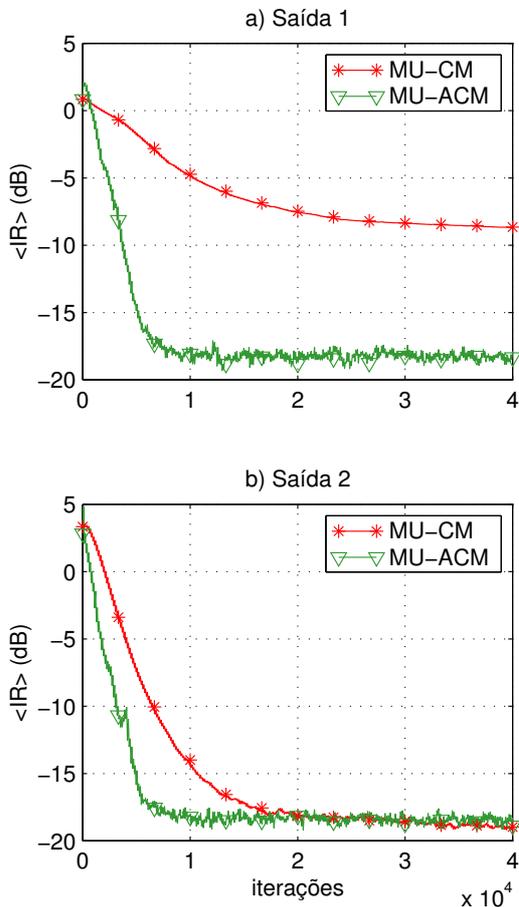


Fig. 2. Curvas de interferência residual (IR) dos algoritmos MU-CM ( $\mu = 5 \times 10^{-4}$ ,  $\xi = 3$ ,  $f_e = 0,55$ ), MU-ACM ( $\mu = 0,25$ ,  $m_1 = 0,16$ ,  $\gamma = 106$ ,  $\xi = 3$ ,  $f_e = 0,55$ ), para 4-QAM,  $N = 2$ ,  $L = 3$ ,  $M = 15$ , Canal 1 sem ruído e média de 50 realizações.

TABELA IV

NÚMERO DE OPERAÇÕES POR ITERAÇÃO DOS ALGORITMOS PARA SINAIS REAIS E  $N = 2$  USUÁRIOS.

Op.	MU-CM	MU-ACM
$\times$	$10M + 12$	$21M + 19$
$+$	$8M + 1$	$15M + 4$
$\div$	-	2

Na Figura 4 são mostrados, durante um certo intervalo de tempo, os erros comparando os sinais transmitidos com os sinais das saídas dos decisores quando da utilização do canal 2 sem ruído e modulação 2-PAM. Na Figura 4-e é mostrado um gráfico das magnitudes dos zeros do canal  $H_{22}(z)$  ao longo das iterações. As situações de maior erro correspondem àquelas em que ocorre uma mudança muito brusca nos valores das

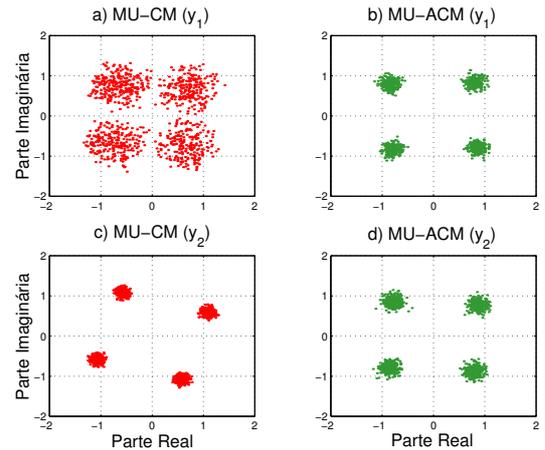


Fig. 3. Constelações nas saídas filtros do equalizador entre as iterações 39000 e 40000 considerando os algoritmos MU-CM ( $\mu = 5 \times 10^{-4}$ ,  $\xi = 3$ ,  $f_e = 0,55$ ), MU-ACM ( $\mu = 0,25$ ,  $m_1 = 0,16$ ,  $\gamma = 106$ ,  $\xi = 3$ ,  $f_e = 0,55$ ), para 4-QAM,  $N = 2$ ,  $L = 3$ ,  $M = 15$  e Canal 1 sem ruído.

magnitudes dos zeros e também quando há nulos espectrais no canal, o que acontece próximo das iterações 5000, 27000 e 37000. Próximo da iteração 5000, o algoritmo MU-ACM apresenta uma recuperação mais rápida que a do algoritmo MU-CM para as duas saídas. Considerando a Saída 2 (figuras 4-c e 4-d), entre as iterações 27000 e 37000 o algoritmo MU-ACM apresenta novamente um desempenho melhor que o do algoritmo MU-CM. As curvas de interferência residual para este caso são mostradas na Figura 5. Novamente o equalizador é formado por dois vetores com  $M = 15$  coeficientes cada inicializados com apenas dois elementos não nulos (iguais a um) nas posições 5 e 7 respectivamente. Embora este canal não seja muito realista pois na prática haveriam mais canais variantes, esta simulação é interessante para comparar o comportamento dos algoritmos em situações críticas: nulos espectrais e variações bruscas.

VII. CONCLUSÕES

O algoritmo autodidata do tipo “acelerador” introduzido em [1] foi estendido neste trabalho para equalização espaço-temporal. Ele apresenta um compromisso mais favorável entre velocidade de convergência e complexidade computacional que o algoritmo MU-CM. Por meio de simulações, o comportamento observado no caso monousuário se repetiu, ou seja o algoritmo proposto foi capaz de evitar mínimos locais e apresentou uma rápida recuperação para um canal variante no tempo. Uma análise de convergência e estabilidade deste algoritmo está sendo feita. Além disso, sua comparação com um algoritmo do tipo Super-Exponencial está sendo investigada.

AGRADECIMENTOS

Os autores agradecem Charles C. Cavalcante, aluno do DECOM-UNICAMP, por proveitosas discussões técnicas.

REFERÊNCIAS

[1] M. T. M. Silva, M. Gerken, M. D. Miranda, “An Accelerated Constant Modulus Algorithm,” *Proceedings of International Telecommunications Symposium - ITS'2002*, Sept. 2002, Natal, RN, Brazil.

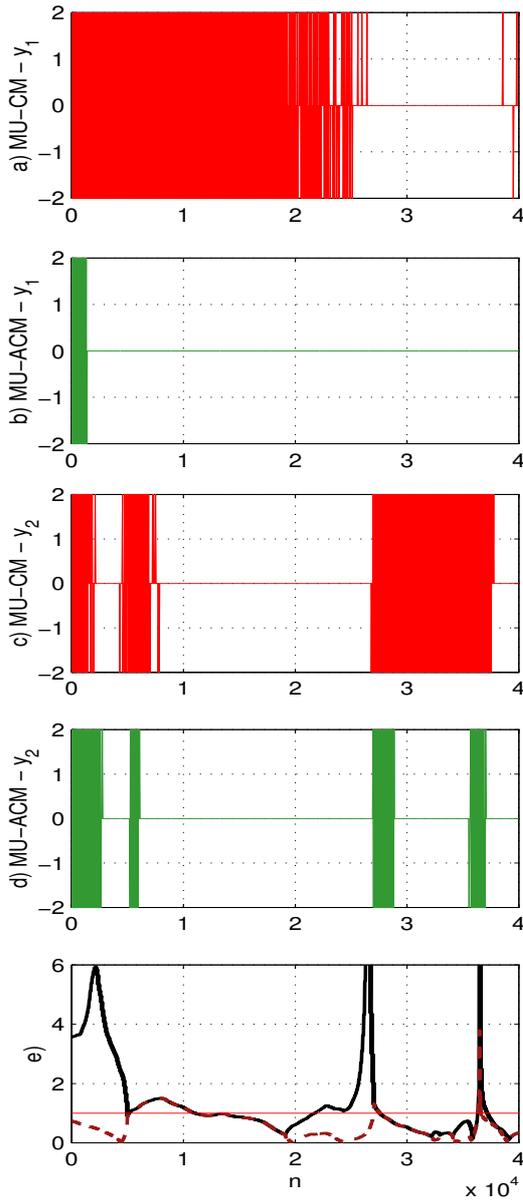


Fig. 4. a) e b) Erro na recuperação do sinal  $a_1(n)$ . c) e d) Erro na recuperação do sinal  $a_2(n)$ . e) Módulos dos zeros (curvas cheia e tracejada) do canal  $H_{22}(z)$  (canal variante no tempo). Curva com amplitude 1 representando a circunferência unitária. Algoritmos MU-CM ( $\mu = 5 \times 10^{-4}$ ,  $\xi = 3$ ,  $f_e = 0,99$ ), MU-ACM ( $\mu = 0,25$ ,  $m_1 = 0,16$ ,  $\gamma = 125$ ,  $\xi = 3$ ,  $f_e = 0,99$ ), para 2-PAM,  $N = 2$ ,  $L = 3$ ,  $M = 15$  e Canal 2 sem ruído.

[2] O. Shalvi, E. Weinstein, "Universal Methods for blind deconvolution," Chap. 4 in: *Blind Deconvolution*, S. Haykin, ed., Prentice Hall, New Jersey, 1994.  
 [3] D. N. Godard, "Self-recovering equalization and carrier tracking in two dimensional data communication system," *IEEE Transactions on Communications*, v. COM-28, n. 11, pp. 1867-1875, Nov. 1980.  
 [4] K. L. Yeung, S. F. Yau, "A cumulant-based super-exponential algorithm for blind deconvolution of multi-input multi-output systems," *Signal Processing*, v. 67, pp. 141-162, 1998.  
 [5] R. Lambert, C. Nikias, "Blind deconvolution of multipath mixtures," in: *Unsupervised Adaptive Filtering*, S. Haykin, ed., v. 1, John Wiley & Sons, New York, 2000.  
 [6] C. B. Papadias, A. Paulraj, "On the blind separability of multiple user signals in the presence of delay spread," *Proceedings of the Thirty-first Asilomar Conference on Signal Processing, Systems & Computers*, v. 2, pp. 1400-1404, Nov. 1997.

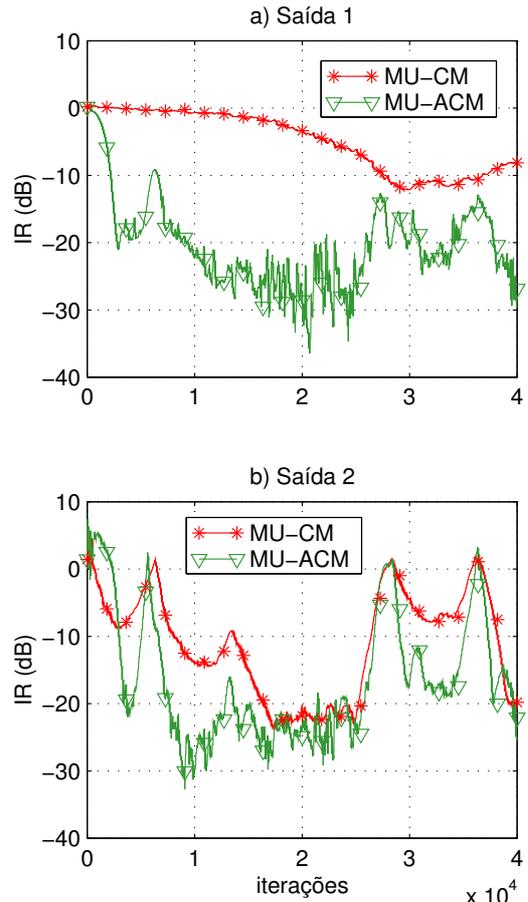


Fig. 5. Curvas de interferência residual (IR) dos algoritmos MU-CM ( $\mu = 5 \times 10^{-4}$ ,  $\xi = 3$ ,  $f_e = 0,99$ ), MU-ACM ( $\mu = 0,25$ ,  $m_1 = 0,16$ ,  $\gamma = 125$ ,  $\xi = 3$ ,  $f_e = 0,99$ ), para 2-PAM,  $N = 2$ ,  $L = 3$ ,  $M = 15$  e Canal 2 sem ruído.

[7] C. B. Papadias, A. J. Paulraj, "A constant modulus algorithm for multiuser signal separation in presence of delay spread using antenna arrays," *IEEE Signal Processing Letters*, v. 4, n. 6, pp. 178-181, June 1997.  
 [8] A. J. Paulraj, C. B. Papadias, "Space-time processing for wireless communications," *IEEE Signal Processing Magazine*, pp. 49-83, Nov. 1997.  
 [9] C. J. Escudero, "A blind signal separation method for multiuser communications," *IEEE Transactions on Signal Processing*, v. 45, pp. 1343-1348, May 1997.  
 [10] C. B. Papadias, A. Paulraj, "A space-time constant modulus algorithm for SDMA systems," *Proceedings of IEEE/VTS 46th Vehicular Technology Conf. (VTC-96)*, Atlanta, GA, Apr. 1996.  
 [11] F. R. P. Cavalcanti, A. L. Brandão, J. M. T. Romano, "The generalized constant modulus algorithm applied to multiuser space-time equalization," *Proceedings of Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 94-97, 1999.  
 [12] F. M. Pait, "A Tuner that Accelerates Parameters," *Systems & Control Letters*, v. 35, pp. 65-68, 1998.  
 [13] P.E. Jojoa Gomez, M. Gerken, F.M. Pait, "The Accelerating Adaptive Filtering Algorithm," *Proceedings of IFAC ALCOSP'2001*, pp. 331-335, Aug. 2001, Cernobio-Como, Italy.  
 [14] T. Shimamura, C. F. N. Cowan, "Equalisation of time variant multipath channels using amplitude banded techniques," *Proceedings of IEEE ICASSP'1997*, pp. 2497-2500.