

ANÁLISE E PROJETO DE ALGORITMOS NUMÉRICOS PARA PROCESSAMENTO
DE SINAIS UTILIZANDO A TEORIA DE CONTROLE

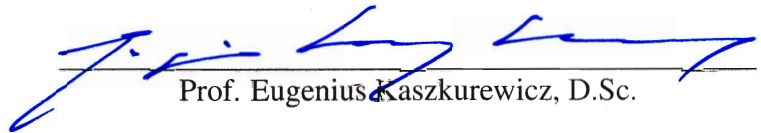
Oumar Diene

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS
EM ENGENHARIA ELÉTRICA

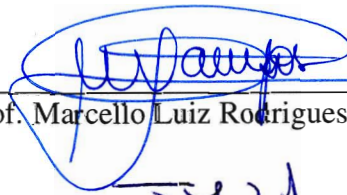
Aprovada por:



Prof. Amit Bhasa, Ph.D.



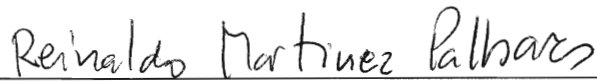
Prof. Eugenius Kaszkurewicz, D.Sc.



Prof. Marcello Luiz Rodrigues de Campos, Ph.D.



Prof. João Marcos Travassos Romano, Dr.



Prof. Reinaldo Martinez Palhares, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

JUNHO DE 2008

DIENE, OUMAR

Análise e Projeto de Algoritmos Numéricos
para Processamento de Sinais utilizando a Teo-
ria de Controle [Rio de Janeiro] 2008

XXVI, 207p. 29,7 cm (COPPE/UFRJ,
D.Sc., Engenharia Elétrica, 2008)

Tese - Universidade Federal do Rio de
Janeiro, COPPE

1. Filtragem adaptativa
2. Filtragem adaptativa cega
3. Treinamento de perceptrons
4. Estabilidade numérica de métodos iterativos
5. Função de Liapunov com controle
6. Sistemas dinâmicos realimentados

I. COPPE/UFRJ II. Título (série)

*Des chercheurs qui cherchent on en trouve,
Des chercheurs qui trouvent on en cherche.*

Aos meus pais *Ndéye Touty Guèye* e *Médoune Diene* pela educação, pela exortação e pela consolação ímpar nos momentos de minhas impaciências e atribulações, sendo para mim fonte de inspiração e referência na minha vida profissional e pessoal.

À minha querida *Andresa Nunes do Nascimento* pela companhia em todos os momentos alegres e tristes durante os quatro últimos anos, pelo incentivo nos momentos mais difíceis.

Ela tem sido minha inspiração e meu ombro amigo desde que Deus cruzou nossos caminhos, certamente tornou minha vida mais alegre.

Á minha avó *Nafi Bitéye*, que Deus lhe dê mais saúde e prolongue muito mais sua vida.

Às minhas irmãs e aos meus irmãos *Mariétou*, *Aminata*, *Anna*, *Babacar*, *Amadou*, *Abdoul Aziz*, *Younouss* e *Binetou*, que Deus os proteja.

Aos meus queridos sobrinhos e sobrinhas *Papa Nguirane Ndiaye*, *Amadou Médoune Lack*, *Khadidiatou Ndiaye*, *Fatou Nafi Lack*, *Médoune Diene*, *Mamadou Diene*, *Akhmed José Lack*, *Moussa Diene*, *Touty Mariam Aminata Nascimento Diene*, *Landing Diene*, *Bineta Mbissine Diene* e *Tamsir Badou Benjamin Aziz Nascimento Diene*, que este trabalho possa lhes servir de fonte de inspiração.

Às minhas tias *Amy Ba* e *Awa Diene*, pela dedicação e disposição sempre demonstrada.

Aos meus avôs *Moussa Diene*, *Amadou Guèye* e *Ibrahima Tamsir Bitéye*, à minha avó *Fatou Guèye*, que Deus os tenha no seu reino.

À todos meus familiares e amigos

Agradecimentos

Gostaria de deixar registrado os meus sinceros agradecimentos à equipe que esteve me ajudando nesta trajetória.

Agradeço a Deus por ele ter iluminado meu caminho e ter me dado a força para prosseguir no caminho apesar das inúmeras dificuldades encontradas.

À toda família Nascimento: Patricia, Dona Melinha e Seu Evaristo, por terem me acolhido tão bem.

Gostaria de manifestar a minha gratidão ao meu orientador Amit Bhaya, pela confiança no meu potencial e pela orientação acadêmica oferecida durante todo este percurso.

Aos amigos do NACAD e da área de Controle Marcos Moreira, Lilian Kawakami, Júnia Almeida, Leonardo Valente, Valeriana Roncero, Myriam Costa, Angelo Silva, Alessandro Jacoud, Tiago Roux, Eduardo Nunes, Josiel Alves, pela convivência agradável e pelas altas reflexões geradas na hora do almoço.

Aos meus amigos de infância Sékhou Séne, Serigne Samb, Alpha Oumar Touré, Mouhammadou Sarr, Gora Fall e Seydou Oumar Touré, pelo companheirismo de sempre, que este trabalho lhes sirva de exemplo.

Ao amigo Sidney Cândido da Silva e aos compatriotas senegaleses do Rio de Janeiro e do Brasil todo, por ajudarem a diminuir a saudade do minha terra querida Senegal.

Por fim, gostaria de manifestar minha gratidão ao professor Marcello Luiz Rodrigues de Campos, pela valiosa ajuda no entendimento e na simulação de algoritmos de processamento de sinais.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

ANÁLISE E PROJETO DE ALGORITMOS NUMÉRICOS PARA PROCESSAMENTO DE SINAIS UTILIZANDO A TEORIA DE CONTROLE

Oumar Diene

Junho/2008

Orientador: Amit Bhaya

Programa: Engenharia Elétrica

Análise numérica é uma área na qual a contribuição da teoria de controle é, até o presente momento, menos significativa. Nesta perspectiva é que se colocam as contribuições desta tese: a utilização da teoria e o ferramental de controle para analisar e desenvolver novos algoritmos numéricos para a resolução dos problemas de filtragem adaptativa, filtragem adaptativa com restrições lineares, filtragem adaptativa cega, filtragem adaptativa cega com restrições lineares, treinamento de perceptrons e para a análise da robustez dos algoritmos propostos.

Os métodos numéricos são representados como sistemas dinâmicos do tipo planta e controlador na configuração padrão de realimentação negativa. As demonstrações de convergência dos métodos e da otimalidade dos parâmetros são obtidas a partir de funções de Liapunov com controle. Os efeitos das perturbações devidos aos erros de precisão finita são formulados como perturbações multiplicativas na entrada da planta. Medidas da robustez dos algoritmos numéricos estudados são obtidas através do teorema do pequeno ganho e de funções de Liapunov com controle.

Os novos métodos propostos nesta tese não utilizam nenhum parâmetro empírico e seu desempenho é comparável ou superior ao dos melhores métodos existentes na literatura, conforme mostram as simulações numéricas que reforçam os resultados teóricos.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

ANALYSIS AND DESIGN OF NUMERICAL ALGORITHMS FOR SIGNAL
PROCESSING USING CONTROL THEORY

Oumar Diene

June/2008

Advisor: Amit Bhaya

Department: Electrical Engineering

Applications of control theory in numerical analysis have been relatively few so far. This thesis makes a contribution by using control theory and tools for the analysis and design of new numerical algorithms for solving adaptive filtering, linearly constrained adaptive filtering, blind adaptive filtering, blind linearly constrained adaptive filtering and perceptron training problems, as well as in the robustness analysis of the proposed algorithms.

The numerical methods are represented as dynamical systems composed of a plant and a controller in standard unitary feedback configuration. The convergence and the optimality of the parameters of the methods are proved using control Liapunov functions. The effects of the numerical errors due to finite precision are formulated as multiplicative perturbations at the input of the plant. Measures of the robustness of the numerical methods are obtained by using the small gain theorem and control Liapunov functions.

The new methods proposed in this thesis do not utilize any empirical parameters and their performance is comparable or superior to the best existing methods in the literature, as shown by numerical simulations that back up the theoretical results.

Résumé de la Thèse soutenue à COPPE/UFRJ comme partie des pré-requis nécessaire pour l'obtention du titre de Docteur en Sciences (D.Sc.)

ANALYSE ET SYNTHÈSE DE MÉTHODES NUMÉRIQUES POUR LE TRAITEMENT DES SIGNAUX UTILISANT LA THÉORIE DE CONTROLE

Oumar Diene

Juin/2008

Directeur: Amit Bhaya

Departement: Génie Electrique

La contribution de la théorie de contrôle dans l'analyse numérique est jusqu'à présent peu significative. C'est dans cette perspective que se situent les contributions de cette thèse: l'utilisation de la théorie et de l'outillage de contrôle pour analyser et développer de nouveaux algorithmes numériques pour la résolution des problèmes de filtrage adaptatif, filtrage adaptatif avec des restrictions linéaires, filtrage adaptatif aveugle, filtrage adaptatif aveugle avec des restrictions linéaires, l'entraînement des perceptrons et pour l'analyse de la stabilité numérique des méthodes proposées.

Les méthodes numériques sont représentées comme des systèmes dynamiques du genre plante et contrôleur dans la configuration standard de réalimentation unitaire. Les preuves de convergence des méthodes et les preuves de l'optimalité des paramètres sont obtenues à partir de fonctions de contrôle de Lyapunov. Les effets des perturbations dues aux erreurs de la précision finite sont formulés comme des perturbations multiplicatives à l'entrée de la plante. Des mesures de la robustesse des algorithmes numériques étudiés sont dérivées à partir du théorème du petit gain et de fonctions de contrôle de Lyapunov.

Les nouvelles méthodes proposées dans cette thèse n'utilisent aucun paramètre empirique et leur performance est comparable ou supérieur à celle des meilleures méthodes qui existent dans la littérature, comme le montrent les simulations numériques qui viennent renforcer les résultats théoriques.

Sumário

Resumo	vi
Abstract	vii
Résumé	viii
Lista de Algoritmos	xii
Lista de Figuras	xiii
Lista de Tabelas	xx
Lista de Símbolos	xxii
Lista de Abreviaturas	xxiv
1 Introdução	1
1.1 Revisão bibliográfica comentada	1
1.1.1 Métodos iterativos para sistemas lineares	1
1.1.2 Interpretação de métodos iterativos como sistemas dinâmicos com controle	2
1.1.3 Métodos iterativos em processamento de sinais: filtragem adaptativa	3
1.1.4 Métodos iterativos em processamento de sinais: filtragem adaptativa cega	3
1.1.5 Métodos iterativos para desigualdades lineares: projeto de perceptrons	4
1.1.6 Robustez de algoritmos numéricos	4
1.2 Organização da tese	5
2 Análise e projeto de algoritmos de filtragem adaptativa	7
2.1 Introdução	7
2.2 Formulação do problema	8
2.2.1 Filtragem adaptativa	8
2.2.2 Filtragem adaptativa com restrições lineares	10
2.3 Preliminares: formulação de algoritmos iterativos como sistemas de controle	10
2.4 Algoritmos de filtragem adaptativa	13
2.4.1 Método LMS (<i>Least Mean Squares</i>)	13
2.4.2 Método RLS (<i>Recursive Least Squares</i>)	15

2.4.3	Método SD-CLF (<i>Steepest Descent</i>)	17
2.4.4	Método CG-CLF (<i>Conjugate Gradient</i>)	21
2.5	Algoritmos de filtragem adaptativa com restrições lineares	27
2.5.1	Método CLMS (<i>Constrained Least Mean Squares</i>)	27
2.5.2	Método CRLS (<i>Constrained Recursive Least Squares</i>)	29
2.5.3	Método com estrutura GSC	32
2.5.4	Método CSD-CLF (<i>Constrained Steepest Descent</i>)	33
2.5.5	Método CCG-CLF (<i>Constrained Conjugate Gradient</i>)	36
2.6	Simulações Numéricas	39
2.6.1	Equalização Adaptativa	40
2.6.2	Identificação de Sistema	42
2.6.3	Predição Linear	45
2.6.4	Sistema de comunicação móvel DS-CDMA	47
2.7	Contribuições e Publicações	48
3	Análise e projeto de algoritmos de filtragem adaptativa cega	51
3.1	Introdução	51
3.2	Formulação do Problema	52
3.2.1	Filtragem adaptativa cega	52
3.2.2	Filtragem adaptativa cega com restrições lineares	53
3.3	Algoritmos de filtragem adaptativa cega	54
3.3.1	Método SGD-CMA (<i>Stochastic Gradient Constant Modulus</i>)	54
3.3.2	Método RLS-CMA (<i>Recursive Least Squares Constant Modulus</i>)	55
3.3.3	Método CGCLF-CMA (<i>Conjugate Gradient Constant Modulus</i>)	59
3.3.4	Método NLSD-CMA (<i>Non-Linear Steepest Descent Constant Modulus</i>)	61
3.3.5	Método SDCLF-CMA (<i>Steepest Descent Constant Modulus</i>)	63
3.4	Algoritmos de filtragem adaptativa cega com restrições lineares	64
3.4.1	Método CSGD-CMA (<i>Constrained Stochastic Gradient Constant Modulus</i>)	64
3.4.2	Método CRLS-CMA (<i>Constrained Recursive Least Squares Constant Modulus</i>)	67
3.4.3	Método com estrutura GSC	68
3.4.4	Método CCGCLF-CMA (<i>Constrained Conjugate Gradient Constant Modulus</i>)	69
3.4.5	Método CSDCLF-CMA (<i>Constrained Steepest Descent Constant Modulus</i>)	71
3.5	Síntese dos controladores	73
3.6	Simulações numéricas	74
3.6.1	Sistema de antenas adaptativas	74
3.6.2	Sistema de comunicação móvel DS-CDMA	80
3.7	Contribuições e Publicações	82
4	Análise e projeto de algoritmos de treinamento de perceptron	84
4.1	Introdução	84
4.2	Formulação do problema	85
4.3	Algoritmos de treinamento de perceptrons	86
4.3.1	Método Ho-Kashyap	86

4.3.2	Método Ho-Kashyap adaptativo	88
4.3.3	Método SDP-CLF (<i>Steepest Descent Perceptron</i>)	91
4.3.4	Método CGP-CLF (<i>Conjugate Gradient Perceptron</i>)	94
4.4	Simulações Numéricas: Reconhecimento de padrão	98
4.5	Contribuições e Publicações	103
5	Estudo da robustez de métodos iterativos lineares	105
5.1	Introdução	105
5.2	Formulação do problema	106
5.3	Métodos iterativos estacionários	110
5.3.1	Método de Jacobi	111
5.3.2	Método de Gauss-Seidel	116
5.4	Métodos iterativos dinâmicos	118
5.4.1	Método SD-CLF para sistemas lineares	119
5.4.2	Método CG-CLF para sistemas lineares	120
5.4.3	Método SD-CLF para filtros adaptativos	122
5.4.4	Método CG-CLF para filtros adaptativos	123
5.4.5	Método CSD-CLF para filtros adaptativos com restrições lineares	124
5.4.6	Método CCG-CLF para filtros adaptativos com restrições lineares	125
5.4.7	Método SDCLF-CMA para filtros adaptativos cegos	125
5.4.8	Método CGCLF-CMA para filtros adaptativos cegos	126
5.4.9	Método CSDCLF-CMA para filtros adaptativos cegos com restrições lineares	127
5.4.10	Método CCGCLF-CMA para filtros adaptativos cegos com restrições lineares	128
5.4.11	Método SDP-CLF para o treinamento de perceptrons	129
5.4.12	Método CGP-CLF para o treinamento de perceptrons	129
5.5	Simulações Numéricas	130
5.5.1	Sistemas lineares	131
5.5.2	Filtros adaptativos	133
5.5.3	Filtros adaptativos com restrições lineares	135
5.5.4	Filtros adaptativos cegos	135
5.5.5	Filtros adaptativos cegos com restrições lineares	138
5.5.6	Treinamento de perceptrons	139
5.6	Contribuições e Publicações	140
6	Conclusões, Contribuições e Trabalhos futuros	142
6.1	Conclusões	142
6.2	Contribuições	144
6.3	Trabalhos futuros	147
	Apêndices	149
	A Artigos Publicados	149
	Referências Bibliográficas	199

Lista de Algoritmos

2.1	Método CG para sistemas lineares	11
2.2	Método LMS para filtros adaptativos	14
2.3	Método RLS para filtros adaptativos	17
2.4	Método SD-CLF para filtros adaptativos	21
2.5	Método CG-CLF para filtros adaptativos	27
2.6	Método CLMS para filtros adaptativos com restrições lineares	29
2.7	Método CRLS para filtros adaptativos com restrições lineares	33
2.8	Método CSD-CLF para filtros adaptativos com restrições lineares	37
2.9	Método CCG-CLF para filtros adaptativos com restrições lineares	39
3.1	Método SGD-CMA para filtros adaptativos cegos [30]	55
3.2	Método RLS-CMA para filtros adaptativos cegos [70]	59
3.3	Método CGCLF-CMA para filtros adaptativos cegos	61
3.4	Método NLSD-CMA para filtros adaptativos cegos	62
3.5	Método SDCLF-CMA para filtros adaptativos cegos	64
3.6	Método CSGD-CMA para filtros adaptativos cegos com restrições lineares .	66
3.7	Método CRLS-CMA para filtros adaptativos cegos com restrições lineares .	69
3.8	Método CCGCLF-CMA para filtros adaptativos cegos com restrições lineares	71
3.9	Método CSDCLF-CMA para filtros adaptativos cegos com restrições lineares	73
4.1	Método Ho-Kashyap para treinamento de perceptrons em modo batelada . .	88
4.2	Método Ho-Kashyap adaptativo para treinamento de perceptrons em modo batelada	89
4.3	Método SDP-CLF para treinamento de perceptrons em modo batelada . . .	93
4.4	Método CGP-CLF para treinamento de Perceptron em modo batelada . . .	98

Lista de Figuras

2.1	Filtro transversal adaptativo de resposta finita ao impulso (FIR).	8
2.2	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (2.9) por métodos iterativos. O problema consiste em projetar um controlador tal que $\mathbf{r}_k \rightarrow \mathbf{0}$ e $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}$, quando $k \rightarrow \infty$	12
2.3	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema da filtragem adaptativa (2.7) pelo sistema dinâmico (2.18) correspondente ao método LMS. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $y_k \rightarrow d_k$, quando $k \rightarrow \infty$	14
2.4	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (2.7) pelo sistema dinâmico (2.28) correspondente ao método RLS. O vetor de ganhos proporcionais \mathbf{v}_{k+1} é dado por (2.23).	17
2.5	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa (2.7) pelo sistema dinâmico (2.31) correspondente ao método SD-CLF. O problema consiste em determinar o parâmetro de controle α_k tal que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{R}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$	19
2.6	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa (2.7) pelo sistema dinâmico (2.50) correspondente ao método CG-CLF, sendo $\mathbf{f}_1(\mathbf{p}_k) = \beta_k \mathbf{p}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} e_{k+1}^*$. O problema consiste em determinar os parâmetros de controle α_k e β_k tais que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{R}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$	24
2.7	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (2.73) correspondente ao método CLMS. O controle é uma função afim do erro, $\mathbf{f}_1(e_k^*) = \mu \mathbf{P} \mathbf{x}_k e_k^* + \bar{\mathbf{f}}$. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $y_k \rightarrow d_k$, quando $k \rightarrow \infty$	30
2.8	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (2.85) correspondente ao método CRLS. O controle é uma função afim do erro, $\mathbf{f}_1(e_{k+1}) = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1}^* + \bar{\mathbf{f}}_{k+1}$ sendo \mathbf{v}_{k+1} dado por (2.23).	32

2.9	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (2.102) correspondente ao método CSD-CLF, sendo $\mathbf{f}_1(\mathbf{g}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k$. O problema consiste em determinar o parâmetro de controle α_k tal que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\bar{\mathbf{R}}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$	36
2.10	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (2.110) correspondente ao método CCG-CLF, sendo $\mathbf{f}_1(\mathbf{p}_k) = \beta_k \mathbf{p}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{e}}_{k+1}^*$ e $\mathbf{f}_2(\mathbf{p}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k$. O problema consiste em determinar os parâmetros de controle α_k e β_k tais que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\bar{\mathbf{R}}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$	38
2.11	Diagrama de blocos do equalizador adaptativo.	40
2.12	Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos equalizadores adaptativos com $W = 2.9$. Os valores de regime estão apresentados na tabela 2.1.	41
2.13	Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos equalizadores adaptativos com $W = 3.5$. Os valores de regime estão apresentados na tabela 2.1.	41
2.14	Identificação de uma planta dinâmica desconhecida por filtro adaptativo com <i>tap</i> atrasador. Os dados de entrada u e os pesos do <i>tap</i> w são reais [15]. . .	43
2.15	Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos na identificação de um FIR de ordem 10. Os valores de regime estão apresentados na tabela 2.2.	44
2.16	Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos na identificação de um FIR de ordem 20. Os valores de regime estão apresentados na tabela 2.2.	44
2.17	Diagrama de blocos de um filtro adaptativo na configuração de predição linear. . .	45
2.18	Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos na configuração de predição linear com $\kappa = 3$. Os valores de regime estão apresentados na tabela 2.3.	46
2.19	Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos na configuração de predição linear com $\kappa = 10$. Os valores de regime estão apresentados na tabela 2.3.	46
2.20	Modelo simplificado de transmissão síncrona <i>downlink</i> de K usuários. . . .	48
2.21	Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 3 usuários.	49
2.22	Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 5 usuários.	49

3.1	Filtro transversal adaptativo cego de resposta finita ao impulso (FIR).	52
3.2	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.4) pelo sistema dinâmico (3.10) correspondente ao método SGD-CMA, sendo $\mathbf{v}_k = \mathbf{x}_k y_k^*$ e $\mathbf{f}_1(y_k^*) = y_k ^2$. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $ y_k ^2 \rightarrow \rho$, quando $k \rightarrow \infty$	55
3.3	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (3.24) correspondente ao método RLS-CMA, sendo \mathbf{v}_{k+1} dada por (3.20).	58
3.4	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (3.26) correspondente ao método CGCLF-CMA, sendo $\mathbf{f}_1(\mathbf{p}_k) = \beta_k \mathbf{p}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{z}_{k+1} e_{k+1}^*$. O problema consiste em determinar os parâmetros de controle α_k e β_k tais que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{R}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$. . .	60
3.5	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (3.34) correspondente ao método SDCLF-CMA, sendo \mathbf{R}_k e \mathbf{b}_k dados por (3.14) e (3.15) respectivamente. O problema consiste em determinar o parâmetro de controle α_k tal que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{R}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$. .	63
3.6	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega com restrições lineares (3.37) pelo sistema dinâmico (3.41) correspondente ao método CSGD-CMA, sendo $\mathbf{f}_1(y_k^*) = y_k ^2$ e $\mathbf{f}_2(e_k) = \mu \mathbf{P} \mathbf{x}_k y_k^* e_k + \bar{\mathbf{f}}$. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $ y_k ^2 \rightarrow \rho$, quando $k \rightarrow \infty$	66
3.7	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.43) pelo sistema dinâmico (3.52) correspondente ao método CRLS-CMA, sendo $\mathbf{f}_1(e_{k+1}) = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1} + \bar{\mathbf{f}}_{k+1}$ e \mathbf{v}_{k+1} dado por (3.20).	69
3.8	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.5) pelo sistema dinâmico (3.59) correspondente ao método CCGCLF-CMA, sendo $\mathbf{f}_1(\mathbf{p}_k) = \beta_k \mathbf{p}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{z}}_{k+1} \bar{e}_{k+1}^*$ e $\mathbf{f}_2(\mathbf{p}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k$. O problema consiste em determinar os parâmetros de controle α_k e β_k tais que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\bar{\mathbf{R}}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$	71
3.9	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.5) pelo sistema dinâmico (3.66) correspondente ao método CSDCLF-CMA, sendo $\mathbf{f}_1(\mathbf{g}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k$. O problema consiste em determinar o parâmetro de controle α_k tal que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\bar{\mathbf{R}}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$	72
3.10	Conjunto de antenas adaptativas não-supervisionadas.	74
3.11	Quociente entre o sinal e a interferência mais o ruído dos métodos CGCLF-CMA, RLS-CMA, SGD-CMA e NLS-D-CMA	77
3.12	<i>Beampattern</i> normalizado dos métodos CGCLF-CMA, RLS-CMA, SGD-CMA e NLS-D-CMA obtido na iteração 5000.	78

3.13	<i>Beampattern</i> normalizado dos métodos CGCLF-CMA, RLS-CMA, SGD-CMA e NLSD-CMA obtido na iteração 10000.	78
3.14	Erro médio quadrático dos métodos CGCLF-CMA, RLS-CMA, SDCLF-CMA e NLSD-CMA.	79
3.15	Versão filtrada do erro médio quadrático dos métodos CGCLF-CMA, RLS-CMA, SDCLF-CMA e NLSD-CMA.	80
3.16	Curvas de aprendizado dos filtros adaptativos cegos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 3 usuários.	81
3.17	Curvas de aprendizado dos filtros adaptativos cegos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 5 usuários.	81
4.1	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.8) pelo sistema dinâmico (4.16) correspondente ao método Ho-Kashyap, sendo $\mathbf{f}_1(\mathbf{r}_k) = \frac{\mu_1}{2}(\mathbf{r}_k - \mathbf{r}_k)$. O problema consiste em levar o resíduo \mathbf{r}_k para $\mathbf{0}$ de tal forma que $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}_k > \mathbf{0}$, quando $k \rightarrow \infty$	87
4.2	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.8) pelo sistema dinâmico (4.19) correspondente ao método Ho-Kashyap adaptativo, sendo $\mathbf{f}_1(\mathbf{r}_k) = \frac{\mu_1\mu_2}{2} \left[\mathbf{r}_k - \mathbf{r}_k \left(1 - \frac{2}{\mu_1} \right) \right]$ e $\mathbf{f}_2(\mathbf{r}_k) = \frac{\mu_1}{2}(\mathbf{r}_k - \mathbf{r}_k)$. O problema consiste em levar o resíduo \mathbf{r}_k para $\mathbf{0}$ de tal forma que $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}_k > \mathbf{0}$, quando $k \rightarrow \infty$	89
4.3	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.6) pelo sistema dinâmico (4.30) correspondente ao método SDP-CLF, sendo $\mathbf{f}_1(\mathbf{r}_k) = \alpha_k(\mathbf{r}_k - \mathbf{r}_k)$. O problema consiste determinar o parâmetro α_k tal que $\mathbf{r}_k \rightarrow \mathbf{0}$, i.e. $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}_k > \mathbf{0}$, quando $k \rightarrow \infty$	92
4.4	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.6) pelo sistema dinâmico (4.45) correspondente ao método CGP-CLF, sendo $\mathbf{f}_1(\mathbf{r}_k) = \alpha_k(\beta_k \mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{r}_k)$ e $\mathbf{f}_2(\mathbf{r}_k) = \alpha_k(\beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + \mathbf{r}_k - \mathbf{r}_k)$. O problema consiste determinar os parâmetros α_k e β_k tal que $\mathbf{r}_k \rightarrow \mathbf{0}$, i.e. $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}_k > \mathbf{0}$, quando $k \rightarrow \infty$	95
4.5	Curva dos dados da iris: Largura de pétala em função do Comprimento de pétala	99
4.6	Superfícies separadoras das classes <i>iris setosa</i> and <i>iris versicolor</i> . Todos os algoritmos considerados classificam corretamente todas as amostras. A margem de distância M obtida após a convergência é mostrado na tabela 4.1. O critério de parada é $\ \mathbf{r}_k\ \leq 0.001$ e o número máximo de iterações é 1000.	100
4.7	Superfícies separadoras das classes <i>iris setosa</i> and <i>iris virginica</i> . Todos os algoritmos considerados classificam corretamente todas as amostras. A margem de distância M obtida após a convergência é mostrado na tabela 4.1. O critério de parada é $\ \mathbf{r}_k\ \leq 0.001$ e o número máximo de iterações é 1000.	100
4.8	Superfícies separadoras das classes <i>iris versicolor</i> and <i>iris virginica</i> . A tabela 4.1 mostra a estatística das classificações e a margem M obtidas após alcançar o número máximo de iterações. O critério de parada é $\ \mathbf{r}_k\ \leq 0.001$ e o número máximo de iterações é 1000.	101

5.1	Diagrama de blocos ilustrativo do problema de regulação equivalente à resolução de (2.9) por métodos iterativos em aritmética de precisão finita. Os erros de precisão finita são representados como uma perturbação multiplicativa δ na entrada da planta.	109
5.2	Forma geral do diagrama de blocos ilustrativo do problema de regulação equivalente à resolução de (2.9) por métodos iterativos em aritmética de precisão finita. Os erros de precisão finita são representados como uma perturbação multiplicativa δ na entrada da planta. Para os métodos iterativos estacionários a planta \mathbf{P} e o controlador \mathbf{C} podem ser representados tanto no domínio do tempo quanto no domínio da frequência.	110
5.3	Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de Jacobi em aritmética de precisão finita no domínio do tempo. A perturbação δ representa os erros de precisão finita.	112
5.4	Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de Jacobi em aritmética de precisão finita no domínio da frequência. A perturbação δ representa os erros de precisão finita.	113
5.5	Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de Gauss-Seidel em aritmética de precisão finita no domínio do tempo. A perturbação δ representa os erros de precisão finita.	117
5.6	Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de Gauss-Seidel em aritmética de precisão finita no domínio da frequência. A perturbação δ representa os erros de precisão finita.	117
5.7	Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de SD em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	119
5.8	Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de CG em aritmética de precisão finita, sendo $\mathbf{f}_1(\mathbf{p}_k) = (\beta_k \mathbf{I} - \alpha_k \mathbf{A})\mathbf{p}_k$. A perturbação δ_k representa os erros de precisão finita.	121
5.9	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa (2.7) pelo sistema dinâmico (5.53) correspondente ao método SD-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	122
5.10	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa (2.7) pelo sistema dinâmico (5.56) correspondente ao método CG-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	123
5.11	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (5.57) correspondente ao método CSD-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	124

5.12	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (5.58) correspondente ao método CCG-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	125
5.13	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (5.59) correspondente ao método SDCLF-CMA em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita. . . .	126
5.14	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (5.60) correspondente ao método CGCLF-CMA em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita. . . .	127
5.15	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.5) pelo sistema dinâmico (5.61) correspondente ao método CSDCLF-CMA em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	128
5.16	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.5) pelo sistema dinâmico (5.62) correspondente ao método CCGCLF-CMA em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	128
5.17	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.6) pelo sistema dinâmico (5.63) correspondente ao método SDP-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	129
5.18	Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.6) pelo sistema dinâmico (5.64) correspondente ao método CGP-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.	130
5.19	Comparação da taxa de convergência dos métodos iterativos lineares para a matriz \mathbf{A}_2 na presença da perturbação máxima. Os métodos dinâmicos apresentam uma margem de estabilidade maior que os métodos estacionários porém, os últimos apresentam uma taxa de convergência maior na presença da perturbação máxima tolerável.	132
5.20	Curvas de aprendizado dos equalizadores adaptativos com $W = 2.9$. A tabela 5.2 apresenta os valores de δ_M , o maior valor da perturbação δ_k para o qual o erro médio quadrático de cada algoritmo continua convergindo mais para os valores de regime da tabela 2.1.	133
5.21	Curvas de aprendizado dos filtros adaptativos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 3 usuários, na presença da perturbação $\delta_k = 0.99$. Os métodos CCG-CLF e CSD-CLF apresentam um erro médio quadrático menor em regime enquanto o método CRLS diverge.	134
5.22	Erro médio quadrático dos métodos SDCLF-CMA, CGCLF-CMA, NLSDCMA e RLS-CMA, na presença da perturbação δ_M	136

5.23	Quociente entre o sinal e a interferência mais o ruído dos métodos CGCLF-CMA, RLS-CMA, SGD-CMA e NLSD-CMA na presença da perturbação δ_M	136
5.24	<i>Beampattern</i> normalizado dos métodos SDCLF-CMA, CGCLF-CMA, RLS-CMA e NLSD-CMA obtido na iteração 5000, na presença da perturbação δ_M	137
5.25	<i>Beampattern</i> normalizado dos métodos SDCLF-CMA, CGCLF-CMA, RLS-CMA e NLSD-CMA obtido na iteração 10000, na presença da perturbação δ_M	137
5.26	Curvas de aprendizado dos filtros adaptativos cegos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 5 usuários, na presença da perturbação $\delta_k = 0.99$. Os métodos CCGCLF-CMA e CSDCLF-CMA apresentam um erro médio quadrático menor em regime enquanto o método CRLS-CMA diverge.	138
5.27	Superfícies separadoras das classes <i>iris setosa</i> and <i>iris versicolor</i> na presença da perturbação δ_k . O critério de parada é $\ \mathbf{r}_k\ \leq 0.001$ e o número máximo de iterações é 2000.	139
5.28	Superfícies separadoras das classes <i>iris setosa</i> e <i>iris virginica</i> na presença da perturbação δ_k . O critério de parada é $\ \mathbf{r}_k\ \leq 0.001$ e o número máximo de iterações é 2000.	140

Lista de Tabelas

2.1	Erro médio quadrático dos filtros adaptativos na configuração de equalização adaptativa. Os métodos propostos neste trabalho apresentam um desempenho comparável aos métodos LMS, RLS e CG-CW.	42
2.2	Erro médio quadrático dos filtros adaptativos na configuração de identificação de sistema. Os novos métodos propostos nesta tese apresentam um desempenho comparável aos métodos RLS e CG-CW.	43
2.3	Erro médio quadrático dos filtros adaptativos na configuração de predição linear. Os métodos propostos neste trabalho apresentam um desempenho comparável ao método de [19] e ao método RLS.	47
2.4	Porcentagem de Desajuste $M(\%)$	47
2.5	Comparação do custo computacional dos métodos de filtragem adaptativa com ou sem restrições lineares.	50
3.1	Comparação dos sistemas dinâmicos correspondentes aos métodos de filtragem adaptativa com ou sem restrições lineares e de filtragem adaptativa cega com ou sem restrições lineares.	75
3.2	Comparação do custo computacional dos métodos de filtragem adaptativa cega com ou sem restrições lineares.	82
4.1	Desempenho dos algoritmos na separação das classes <i>iris setosa</i> , <i>iris versicolor</i> e <i>iris virginica</i> . A porcentagem $\%CC$ é dado por (4.57) e a margem M é o mínimo das distâncias dos pontos corretamente classificados ao hiperplano de separação. O número máximo de iterações é igual a 1000 e o critério de parada é $\ \mathbf{r}_k\ \leq 0.001$	102
4.2	Comparação de tempo médio e numero de iterações.	103
4.3	Comparação do desempenho dos métodos. O método HK falha na determinação da pseudo-inversa da matriz de dados do conjunto “ <i>Ionosphere</i> ”, e o método AHK diverge no caso dos conjuntos “ <i>BUPA</i> ” e “ <i>Diabetes</i> ”.	103
5.1	Comparação do desempenho dos métodos Jacobi, Gauss-Seidel, SD-CLF e CG-CLF. A perturbação δ representa os erros de precisão finita. A tolerância é escolhida como $\epsilon = 10^{-3}$, a dimensão das matrizes é $n = 1000$ e o número máximo de iterações é igual a 10000. Os métodos dinâmicos apresentam uma margem de estabilidade maior que os métodos estacionários porém, os últimos apresentam uma taxa de convergência maior na presença da perturbação máxima tolerável.	132

5.2	Para os métodos SD-CLF, CG-CLF, LMS e RLS compara-se o valor de δ_M , o maior valor da perturbação δ_k para o qual o erro médio quadrático de cada algoritmo continua convergindo para os valores de regime da tabela 2.1. . . .	133
5.3	Para os métodos SDCLF-CMA, CGCLF-CMA, RLS-CMA e NLS-D-CMA compara-se o valor de δ_M , o maior valor da perturbação δ_k para o qual o erro médio quadrático de cada algoritmo converge para os valores de regime da figura 3.14.	135
5.4	Comparação dos valores de δ_M , para os casos C_1 (separação dos conjuntos <i>iris setosa</i> e <i>iris versicolor</i>) e C_2 (separação dos conjuntos <i>iris setosa</i> e <i>iris virginica</i>).	139

Lista de Símbolos

$(\cdot)^*$	Valor conjugado de (\cdot) , p. 8
$(\cdot)^H$	Transposto conjugado de (\cdot) , p. 8
$(\cdot)^T$	Transposto de (\cdot) , p. 84
$(\cdot)^\dagger$	Pseudo-inversa de (\cdot) , p. 84
$E(\cdot)$	Valor esperado de (\cdot) , p. 8
$O(\epsilon)$	De ordem da precisão da máquina ϵ , p. 105
\approx	Aproximadamente igual a, p. 114
$\arg(\cdot)$	Argumento de (\cdot) , p. 112
$\inf_k(\cdot)$	Maior escalar menor que o menor valor de (\cdot) , p. 18
$\max_\theta(\cdot)$	Maximizar (\cdot) em relação a θ , p. 112
$\min_{\mathbf{w}_k}(\cdot)$	Minimizar (\cdot) em relação ao vetor \mathbf{w}_k , p. 9
\emptyset	Conjunto vazio, p. 100
\forall	Para todo, p. 89
$\frac{\partial(\cdot)}{\partial \mathbf{w}_k}$	Derivada parcial de (\cdot) em relação ao vetor \mathbf{w}_k , p. 9
\gg	Muito maior que, p. 114
∞	Infinito, p. 11
κ	Número de condicionamento, p. 38

λ_f	Fator de esquecimento, p. 15
λ_k	Vetor de multiplicadores de Lagrange, p. 26
$\lambda_{max}(\cdot)$	Maior autovalor de (\cdot) , p. 97
$\lambda_{min}(\cdot)$	Menor autovalor de (\cdot) , p. 18
$\langle \cdot, \cdot \rangle$	Produto interno, p. 11
$\mathbb{C}^{M+1 \times M+1}$	Anel das matrizes de dimensão $(M+1 \times M+1)$ e valores complexos, p. 9
\mathbb{C}^{M+1}	Conjunto dos vetores de dimensão $(M+1)$ e valores complexos, p. 9
$\nabla(\cdot)$	Vetor gradiente de (\cdot) , p. 9
\ominus	Subtração em ponto flutuante, p. 105
\oplus	Adição em ponto flutuante, p. 105
\otimes	Multiplicação em ponto flutuante, p. 105
$\bar{\sigma}\{\cdot\}$	Maior valor singular de $\{\cdot\}$, p. 109
$\partial(\cdot)$	Fronteira da região (\cdot) , p. 109
$\sup(\cdot)$	Menor escalar maior que o maior valor de (\cdot) , p. 109
$\underline{\sigma}\{\cdot\}$	Menor valor singular de $\{\cdot\}$, p. 111
$fl(\cdot)$	Representação de (\cdot) em aritmética de ponto flutuante, p. 104
op	Operações básicas $(+, -, *, /)$ em aritmética de ponto flutuante, p. 105

Lista de Abreviaturas

3G	<i>3rd Generation</i> , p. 145
4G	<i>4th Generation</i> , p. 145
AHK	Ho-Kashyap Adaptativo, p. 96
CCG	<i>Constrained Conjugate Gradient</i> , p. 34
CGP	<i>Conjugate Gradient Perceptron</i> , p. 91
CG	<i>Conjugate Gradient</i> (Gradiente Conjugado), p. 3
CLF	Função de Liapunov com Controle, p. 3
CLMS	<i>Constrained Least Mean Squares</i> , p. 25
CMA	<i>Constant Modulus Algorithm</i> , p. 52
CRLS	<i>Constrained Recursive Least Squares</i> , p. 27
CR	<i>Conjugate Residual</i> , p. 1
CSD	<i>Constrained Steepest Descent</i> , p. 31
CSGD	<i>Constrained Stochastic Gradient</i> , p. 63
DOA	<i>Direction Of Arrival</i> (Ângulo de incidência), p. 75
DS-CDMA	<i>Direct-Sequence Code Division Multiple Access</i> , p. 9
FIR	<i>Finite Impulse Response</i> , p. 8
FLOP	Número de operações de ponto flutuante, p. 10

FOM	<i>Full Orthogonalization Method</i> , p. 1
GCR	<i>Generalized Conjugate Residual</i> , p. 1
GMRES	<i>Generalized Minimum Residual</i> , p. 1
GSC	<i>Generalized Sidelobe Canceler</i> , p. 30
HK	Ho-Kashyap, p. 96
IEEE	<i>Institute of Electrical and Electronics Engineers</i> , p. 49
ISDS	<i>Input to State Dynamical Stability</i> , p. 116
ISS	<i>Input to State Stability</i> , p. 116
LMS	<i>Least Mean Squares</i> , p. 13
LS-SVM	<i>Least Square Support Vector Machine</i> , p. 97
MSE	<i>Mean Square Error</i> (Erro Médio Quadrático), p. 38
NLSD	<i>Non-Linear Steepest Descent</i> , p. 60
PD	Proporcinal de Derivativo, p. 2
PID	Proporcional Integral e Derivativo, p. 2
PI	Proporcional e Integral, p. 2
RLS	<i>Recursive Least Squares</i> , p. 14
SDP	<i>Steepest Descent Perceptron</i> , p. 88
SD	<i>Steepest Descent</i> (Declividade Máxima), p. 13
SGD	<i>Stochastic Gradient</i> , p. 52
SINR	<i>Signal-to-Interference-plus-Noise Ratio</i> , p. 74
SNR	<i>Signal-to-Noise Ratio</i> , p. 45
SOR	<i>Successive Over Relaxation</i> , p. 2

SVM *Support Vector Machine*, p. 97

WLAN *Wireless Local Area Network*, p. 145

Capítulo 1

Introdução

Análise numérica é uma área na qual a contribuição da teoria de controle é, até o presente, menos significativa. Nesta perspectiva é que se coloca o objetivo desta tese: a utilização da teoria e o ferramental de controle para: (i) analisar/justificar o uso dos algoritmos propostos em áreas como, por exemplo, filtragem adaptativa, processamento cego; (ii) desenvolver novos métodos numéricos a partir das técnicas propostas, por exemplo, algoritmos de treinamento de redes de perceptrons; (iii) estudar a robustez dos métodos iterativos lineares e não-lineares.

A seguir, realiza-se uma revisão bibliográfica mencionando os resultados principais existentes nas áreas estudadas nesta tese, contextualizando os resultados a serem apresentados ao longo da tese.

1.1 Revisão bibliográfica comentada

1.1.1 Métodos iterativos para sistemas lineares

A resolução de sistemas lineares do tipo $\mathbf{Ax} = \mathbf{b}$ por métodos iterativos começou a ser estudado nos anos quarenta no contexto de grandes sistemas de equações diferenciais parciais (EDP's) oriundos de física e engenharia nuclear e descrições bastante completas do progresso nesta área podem ser encontradas em Young [1] e Varga [2]. Nos anos 50, Hestenes e Stiefel [3] apresentaram o método do gradiente conjugado (CG) cuja idéia básica consiste em minimizar a função quadrática $\Phi(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Ax} - \mathbf{x}^T \mathbf{b}$ no espaço afim $\mathbf{x}_0 + \mathcal{K}_k$, sendo \mathcal{K}_k um subespaço de Krylov [4]. Vários outros métodos de projeção em subespaço

de Krylov foram desenvolvidos na mesma época, entre outros citam-se os métodos GMRES, FOM, CR, GCR, Arnoldi ([5], [6]). Para a classe de matrizes hermitianas, o método do gradiente conjugado é considerado um dos melhores métodos iterativos ([6], [7]).

1.1.2 Interpretação de métodos iterativos como sistemas dinâmicos com controle

A interpretação de métodos iterativos para sistemas lineares como sistemas dinâmicos foi feita pela primeira vez em Schaerer e Kaszkurewicz [8]. Posteriormente este trabalho foi ampliado e estendido a métodos iterativos em geral nos trabalhos Bhaya e Kaszkurewicz [9] e Bhaya e Kaszkurewicz [10]. Nestes trabalhos, o método do gradiente conjugado é representado e analisado como um sistema dinâmico constituído por uma planta (sistema dinâmico) que, a grosso modo, representa o problema a ser resolvido e um controlador (outro sistema dinâmico) representando o algoritmo empregado. Estes dois sistemas dinâmicos, planta e controlador, são interligados em uma configuração padrão denominada realimentação negativa, objeto fundamental de estudo na teoria de controle.

Alguns resultados destas referências foram estendidos em Diene [11] e Diene e Bhaya [12], obtendo-se alguns resultados novos. No trabalho Diene [11] os métodos iterativos estacionários (Jacobi, Gauss-Seidel, SOR, Richardson) e os métodos dinâmicos do tipo Krylov (Gradiente Conjugado, Resíduo Conjugado, Resíduo Conjugado Generalizado, Gradiente Conjugado com atraso, Gradiente Conjugado para a equação normal) foram analisados nessa perspectiva de sistemas dinâmicos. Desta análise resultou o desenvolvimento de novos métodos iterativos lineares através da utilização de controladores do tipo PI, PD e PID [12].

Em Ferreira [13] a abordagem de métodos numéricos sob um ponto de vista de sistemas dinâmicos foi estendida aos sistemas gradientes que resolvem problemas de otimização. Os sistemas considerados nesse trabalho foram obtidos a partir de um método de penalização exata, resultando em sistemas dinâmicos com segundo membro descontínuo, que podem ser interpretados como modelos de redes neurais com função de ativação descontínua. As análises de convergência, feitas através da forma Persidskii dos sistemas gradientes, utilizando funções de Liapunov não suaves do tipo Lure-Persidskii, resultaram em condições de convergência global tratáveis.

Em Pazos [14] a mesma abordagem foi utilizada para resolver diversos problemas de

otimização. Nesse trabalho os problemas abordados nessa perspectiva foram os de achar zeros de funções vetoriais não lineares, achar mínimos de funções escalares, o problema geral de otimização convexa e desigualdades variacionais.

1.1.3 Métodos iterativos em processamento de sinais: filtragem adaptativa

Na área de processamento de sinais os métodos numéricos são utilizados na resolução dos problemas de filtragem, filtragem adaptativa, codificação, estimação, detecção, análise, reconhecimento, sintetização, gravação e reprodução de sinais [15–17]. No caso da filtragem adaptativa, os métodos iterativos são utilizados para determinar os pesos do filtro. Em Boray e Srinath [18] o método do gradiente conjugado (CG) foi estendido à resolução de sistemas lineares variantes no tempo do tipo $\mathbf{A}_k \mathbf{x}_k = \mathbf{b}_k$, resultante da determinação dos pesos em linha. Nesse caso, várias iterações do CG são realizadas a cada amostra k . Chang e Willson [19] apresentaram uma outra extensão do CG para resolução de sistemas lineares variantes no tempo ($\mathbf{A}_k \mathbf{x}_k = \mathbf{b}_k$), em qual apenas uma iteração do CG é necessária a cada amostra k . Esse último trabalho foi estendido para o caso da filtragem adaptativa com restrições lineares em Apolinário Jr et al. [20]. Nos três últimos trabalhos referenciados acima, não há prova de convergência dos algoritmos propostos, os autores conjecturaram a convergência através de simulações numéricas. Nesta tese, mostrar-se-á como a representação do método CG como sistema dinâmico [21, 22] permite sua extensão para ambos os casos de sistemas lineares variantes no tempo com ou sem restrições lineares, com prova formal de convergência através de uma escolha adequada de função de Liapunov com controle (CLF).

1.1.4 Métodos iterativos em processamento de sinais: filtragem adaptativa cega

No caso do processamento cego de sinais, os métodos numéricos são utilizados por exemplo: para eliminar os efeitos dos caminhos múltiplos dos sinais [23]; para cancelar as interferências devidas a sinais em frequências próximas à frequência de um sinal transmitido [24]; para identificar o sinal de um usuário em um sistema de antenas multiusuários [25–28]; para equalizar sinais corrompidos por ruídos sem dispor do sinal desejado [29]. Nestes cenários

uma parte considerável dos sinais apresentam um módulo constante ou com variação lenta, essa propriedade é utilizada para definir uma função objetivo dito critério do módulo constante [30]. A minimização de tal critério permite resolver os problemas citados em um processamento em-linha. Novamente, em estrita analogia com o caso anterior de filtragem adaptativa, é possível projetar métodos do tipo CG para a resolução do problema de minimização definido, mantendo uma característica de solução iterativa em-linha, e esse fato será mostrado nesta tese.

1.1.5 Métodos iterativos para desigualdades lineares: projeto de perceptrons

No contexto de redes neurais, métodos iterativos podem ser utilizados para resolver as desigualdades lineares resultantes do treinamento de perceptrons [31–33]. Em Nagaraja e Krishna [34], foi proposta uma extensão do método CG para resolução de sistemas lineares com o lado direito variante no tempo, $\mathbf{Ax}_k = \mathbf{b}_k$, resultante do treinamento de perceptrons em modo batelada. Em Hassoun e Song [35], são apresentadas versões adaptativas do algoritmo Ho-Kashyap [31] para treinamento de perceptrons em ambos os modos batelada e em-linha. Recentemente, Nagaraja e Bose [36] propuseram um algoritmo CG para o treinamento em-linha de perceptrons, e conjecturaram a convergência através de várias simulações de reconhecimento de padrão. Nesse contexto também, nesta tese desenvolver-se-ão extensões do método CG para o treinamento de perceptrons em ambos os modos batelada [37] e em-linha, com prova formal de convergência.

1.1.6 Robustez de algoritmos numéricos

A estabilidade e o condicionamento são conceitos utilizados na análise numérica para relacionar a resposta de um algoritmo computacional às perturbações decorrentes dos dados ou da utilização da aritmética de precisão finita. Em Björck et al. [38], Paige et al. [39] e Giraud et al. [40] a estabilidade reversa de métodos do tipo Krylov foi estudada. Uma modelagem e um estudo dos erros de precisão finita foram feitas em Strakos e Tichý [41] e Meurant e Strakos [42] no contexto da estabilidade reversa dos métodos do gradiente conjugado e de Lanczos. Na linguagem de controle, os conceitos de estabilidade numérica seriam denominados, genericamente, de robustez do algoritmo numérico. Entretanto, há poucos es-

tudos de robustez de algoritmos numéricos do ponto de vista de controle robusto. A seguir, citam-se trabalhos existentes nesta área para contextualizar a nova proposta feita nesta tese. Em Bunch et al. [43] os autores propõem uma abordagem conceitual para aspectos de consistência, condicionamento e estabilidade numérica em processamento de sinais e sugerem uma terminologia para distinguir entre erros propagados pela natureza do problema e erros propagados pela utilização de aritmética de precisão finita. Em Bunch et al. [44] os mesmos autores desenvolvem esta análise para o caso do filtro transversal rápido, mostrando a divergência do mesmo, bem como possibilidades de consertar este problema à luz da análise proposta no artigo. Entretanto, o foco do artigo é a aplicação de técnicas de álgebra linear numérica à análise de algoritmos de processamento de sinais, e a complexidade destas técnicas, aliada à não-familiaridade das mesmas para a maioria dos engenheiros têm limitado a sua aplicação a outros algoritmos.

No contexto do estudo de sistemas dinâmicos, Grüne [45] apresentou uma nova variação do conceito de estabilidade entrada-estado para estimar e/ou descrever a influência das perturbações. Nesse artigo, o autor apresentou resultados quantitativos sobre a margem de estabilidade de sistemas não lineares e uma versão do teorema do pequeno ganho para sistemas não lineares. No entanto apenas os sistemas dinâmicos com perturbações decrescentes são considerados. Nesta tese, propõe-se uma nova modelagem de perturbações oriundas da utilização de aritmética de precisão finita, baseando-se na combinação do modelo proposto em [41, 42] com a modelagem usual de perturbações multiplicativas na entrada da planta, introduzidas no trabalho de Doyle e Stein [46]. Esta perspectiva de controle, aplicada ao problema clássico de perturbações numéricas (de precisão finita), permite uma aplicação inédita do teorema de pequeno ganho para determinar medidas de robustez ou estabilidade numérica dos algoritmos propostos nesta tese. Enfatiza-se que o objetivo deste exercício não é substituir a análise de estabilidade reversa (*backward stability*) ou os outros tipos de estabilidade, consistência etc. propostos em Bunch et al. [43], e sim prover novas maneiras e medidas quantitativas, familiares a usuários da comunidade de engenheiros, da robustez da estabilidade de métodos iterativos estudados nesta tese.

1.2 Organização da tese

Esta tese está estruturada da seguinte forma: no capítulo 2 os problemas de filtragem adaptativa e de filtragem adaptativa com restrições lineares são revisados e os métodos mais comumente encontrados na literatura são formulados e analisados como sistemas dinâmicos. No mesmo capítulo são propostas duas novas famílias de métodos de filtragem adaptativa e outras duas novas famílias de métodos de filtragem adaptativa com restrições lineares. No capítulo 3 os problemas de filtragem adaptativa cega e de filtragem adaptativa cega com restrições lineares são abordados e os métodos existentes são formulados e analisados como sistemas dinâmicos. Em seguida são propostas duas novas famílias de métodos de filtragem adaptativa cega e mais duas novas famílias de métodos de filtragem adaptativa cega com restrições lineares. O capítulo 4 apresenta uma formulação e uma análise dos métodos de treinamento de perceptrons na perspectiva de sistemas de controle. Dessa formulação desenvolvem-se duas novas famílias de treinamento de perceptrons em modo batelada. O capítulo 5 apresenta uma formulação dos efeitos dos erros de precisão finita como perturbações multiplicativas na entrada da planta. Também no mesmo capítulo medidas da robustez dos métodos numéricos considerados são determinadas. Finalmente o capítulo 6 apresenta as conclusões gerais, as contribuições e os trabalhos futuros decorrentes desta tese.

Capítulo 2

Análise e projeto de algoritmos de filtragem adaptativa

2.1 Introdução

Em várias aplicações da engenharia, tais como a telecomunicação, o controle de sistemas, a detecção de objetos por radar ou sonar, é necessário coletar, analisar e transmitir sinais analógicos e/ou digitais por meio de sensores, transmissores, receptores e processadores. Muitas vezes esses sinais são corrompidos por ruídos de medição dos sensores, distorções e desvanecimentos causados pelo canal de transmissão, interferências de outros sinais na emissão e na recepção dos sinais desejados, etc. Nesses casos é necessário realizar um pré e/ou pós processamento para suprimir ruídos, cancelar ecos, equalizar e realçar os sinais desejados. Esse processamento pode ser feito através de filtros adaptativos e de filtros adaptativos com restrições lineares. Neste capítulo estudar-se-ão os problemas de filtragem adaptativa e de filtragem adaptativa com restrições lineares. Novos métodos de filtragem adaptativa e de filtragem adaptativa com restrições lineares são propostos e analisados, a convergência e a otimalidade desses novos métodos é provada através de funções de Liapunov com controle. Por meio de simulações numéricas mostra-se que os novos métodos propostos têm um desempenho comparável ao de métodos consagrados na literatura.

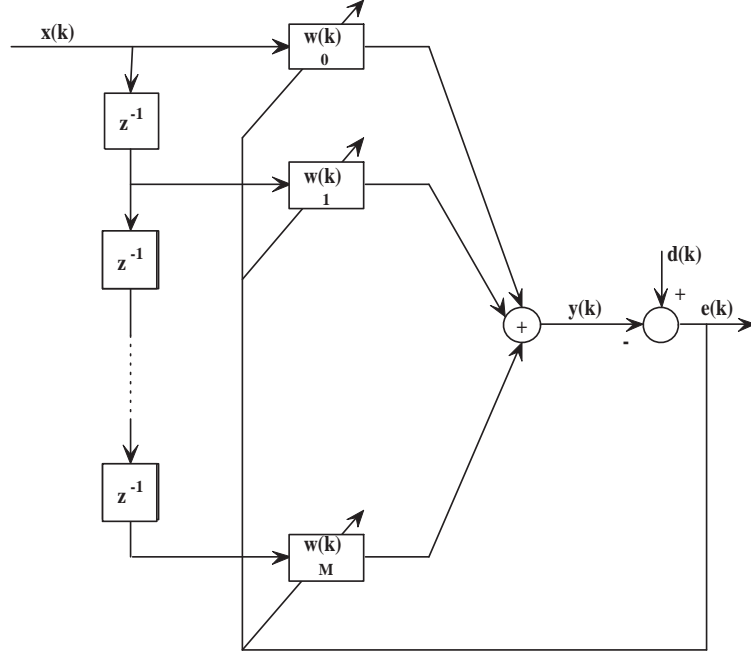


Figura 2.1: Filtro transversal adaptativo de resposta finita ao impulso (FIR).

2.2 Formulação do problema

2.2.1 Filtragem adaptativa

O problema de filtragem adaptativa consiste em reconstituir e/ou determinar o valor de um sinal variante no tempo, a partir de versões passadas e/ou de uma versão do mesmo distorcida por vários fatores e/ou corrompida por um ruído (também variante no tempo). A realização mais básica de um filtro adaptativo é obtida através da forma direta de filtro digital de resposta finita ao impulso (FIR) (como ilustrado na figura 2.1) cuja saída é dada por

$$y_k = \sum_{i=0}^M w_{ik} x_{k-i} = \mathbf{w}_k^H \mathbf{x}_k \quad (2.1)$$

sendo $\mathbf{x}_k = [x_k x_{k-1} \cdots x_{k-M}]^H \in \mathbb{C}^{M+1}$ e $\mathbf{w}_k = [w_{0k} w_{1k} \cdots w_{Mk}]^H \in \mathbb{C}^{M+1}$ os vetores de entrada e de pesos respectivamente, e M a dimensão do *tap* do filtro (número de atrasadores). Uma das funções objetivo mais utilizadas para ajustar os pesos de um filtro adaptativo é o erro médio quadrático definido como

$$F(e_k) = \xi_k = E[e_k e_k^*] = E[d_k d_k^* - y_k d_k^* - d_k y_k^* + y_k y_k^*] \quad (2.2)$$

sendo

$$e_k = d_k - y_k = d_k - \mathbf{w}_k^H \mathbf{x}_k, \quad (2.3)$$

y_k a saída do filtro adaptativo e d_k o sinal desejado [15]. Nesse caso, a função objetivo pode ser reescrita como

$$\begin{aligned} E[e_k e_k^*] = \xi_k &= E[d_k d_k^* - \mathbf{w}_k^H \mathbf{x}_k d_k^* - d_k \mathbf{x}_k^H \mathbf{w}_k + \mathbf{w}_k^H \mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k] \\ &= E[d_k d_k^*] - E[\mathbf{w}_k^H \mathbf{x}_k d_k^*] - E[d_k \mathbf{x}_k^H \mathbf{w}_k] + E[\mathbf{w}_k^H \mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k] \end{aligned}$$

Portanto a função do erro médio quadrático é dada por

$$\xi(\mathbf{w}_k) = E[d_k d_k^*] - \mathbf{w}_k^H \mathbf{b} - \mathbf{b}^H \mathbf{w}_k + \mathbf{w}_k^H \mathbf{R} \mathbf{w}_k \quad (2.4)$$

sendo $\mathbf{b} = E[\mathbf{x}_k d_k^*] \in \mathbb{C}^{M+1}$ o vetor de correlação cruzada entre a resposta desejada e o sinal de entrada, $\mathbf{R} = E[\mathbf{x}_k \mathbf{x}_k^H] \in \mathbb{C}^{(M+1) \times (M+1)}$ a matriz Hermitiana positiva definida de autocorrelação do sinal de entrada, e $E[d_k d_k^*]$ a variância do sinal desejado d_k (constante em relação a \mathbf{w}_k) [15]. Como a função objetivo ξ é uma função quadrática em relação ao vetor de pesos \mathbf{w}_k , o problema de minimizar ξ é equivalente ao problema de achar \mathbf{w}_k tal que

$$\nabla \xi = \frac{\partial \xi}{\partial \mathbf{w}_k} = -2\mathbf{b} + 2\mathbf{R}\mathbf{w}_k = \mathbf{0} \iff \mathbf{R}\mathbf{w}_k = \mathbf{b}. \quad (2.5)$$

Pela equação (2.5) e notando que $\frac{\partial^2 \xi}{\partial \mathbf{w}_k^2} = \mathbf{R} > 0$, pode-se dizer que o vetor \mathbf{w}_o tal que $\mathbf{R}\mathbf{w}_o = \mathbf{b}$ é o argumento que minimiza a função objetivo ξ . Em outras palavras, \mathbf{w}_o é solução de

$$\min_{\mathbf{w}_k} \xi(\mathbf{w}_k). \quad (2.6)$$

Dessa forma, define-se o problema da filtragem adaptativa da seguinte forma:

$$\text{Resolver } \mathbf{R}_k \mathbf{w}_k = \mathbf{b}_k, \quad (2.7)$$

sendo $\mathbf{b}_k = E[\mathbf{x}_k d_k^*]$ e $\mathbf{R}_k = E[\mathbf{x}_k \mathbf{x}_k^H]$, o vetor de correlação cruzada entre a resposta desejada e o sinal de entrada, e a matriz de autocorrelação do sinal de entrada, respectivamente determinados a cada iteração k .

2.2.2 Filtragem adaptativa com restrições lineares

Em algumas aplicações de filtragem adaptativa (comunicação DS-CDMA por exemplo), o vetor de pesos \mathbf{w}_k além de ser solução de (2.7), deve também satisfazer restrições lineares. Nesse caso o problema de filtragem adaptativa é redefinido como um problema de filtragem adaptativa com restrições lineares da seguinte forma:

$$\begin{aligned} \text{Resolver} \quad & \mathbf{R}_k \mathbf{w}_k = \mathbf{b}_k, \\ \text{sujeito a} \quad & \mathbf{C}^H \mathbf{w}_k = \mathbf{f} \end{aligned} \tag{2.8}$$

sendo $\mathbf{C} \in \mathbb{C}^{M+1 \times p}$ a matriz das restrições, e $\mathbf{f} \in \mathbb{C}^p$ o vetor de ganho. Os filtros adaptativos desenvolvidos a partir de (2.8) são ditos filtros de variância mínima com restrições lineares (LCMV, do inglês *Linearly Constrained Minimum Variance*).

2.3 Preliminares: formulação de algoritmos iterativos como sistemas de controle

Para resolver o sistema linear de equações (2.9)

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{2.9}$$

sendo $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{x}, \mathbf{b} \in \mathbb{R}^n$ existem métodos diretos tais como as fatorações **LU** e **QR** que calculam a solução exata (desprezando os erros da representação numérica) $\mathbf{x}_o = \mathbf{A}^{-1}\mathbf{b}$ sem determinar explicitamente a inversa da matriz \mathbf{A} . Porém, estes métodos diretos, como a operação da inversa, requerem um número de operações de ponto flutuante (FLOP) proporcional a n^3 . Este número de operações de ponto flutuante torna estes métodos diretos proibitivos para as matrizes esparsas de dimensão elevada que resultam, por exemplo, da discretização de equações diferenciais parciais. Para este caso existem métodos iterativos que calculam uma solução aproximada de (2.9) a partir de um ponto inicial \mathbf{x}_0 . Em geral, estes métodos iterativos lineares podem ser descritos pela seguinte equação

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k \tag{2.10}$$

Algoritmo 2.1 Método CG para sistemas lineares

Calcular $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$, $\mathbf{p}_0 = \mathbf{r}_0$

Para $k = 0, 1, \dots$, até convergência

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k$$

$$\beta_k = -\frac{\langle \mathbf{r}_{k+1}, \mathbf{A}\mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$$

Fim

sendo \mathbf{x}_k uma aproximação da solução de (2.9) e \mathbf{u}_k uma correção calculada conforme o algoritmo utilizado, para gerar em cada instante uma aproximação mais precisa \mathbf{x}_{k+1} da solução de (2.9) [5], e levar desta forma o resíduo, definido como $\mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k$, a uma pequena vizinhança de zero em um número pequeno de passos (pequena vizinhança definida em termos de uma tolerância; pequeno número de passos em relação à ordem da matriz). Estes métodos iterativos são divididos em dois grupos: os métodos estacionários e os métodos dinâmicos. Os métodos iterativos são ditos estacionários quando podem ser expressos da seguinte forma

$$\mathbf{x}_{k+1} = \mathbf{B}\mathbf{x}_k + \mathbf{c}, \quad (2.11)$$

sendo a matriz de iteração \mathbf{B} e o vetor \mathbf{c} constantes (i.e. não dependentes da iteração k); e dinâmicos quando podem ser expressos como

$$\mathbf{x}_{k+1} = \mathbf{B}_k \mathbf{x}_k + \mathbf{c}_k, \quad (2.12)$$

sendo que a matriz de iteração \mathbf{B}_k e o vetor \mathbf{c}_k dependem da iteração k [47]. O método do gradiente conjugado (CG), mostrado na forma de pseudocódigo no algoritmo 2.1 é considerado como um dos melhores algoritmos para resolver o problema (2.9) [4–7, 48]. O método CG atualiza a aproximação \mathbf{x}_{k+1} da solução de (2.9) através de direções conjugadas do resíduo [3, 49].

Na terminologia de controle, o problema da resolução do sistema (2.9) por métodos iterativos pode ser representado por um problema de rastreamento de sinal, conforme mostram a figura 2.2 e o sistema (2.13). Portanto, o problema da resolução de (2.9) por métodos iterativos consiste em projetar um controle \mathbf{u}_k para o qual, o erro de regime \mathbf{r}_k seja nulo. Em termos matemáticos isto é $\mathbf{r}_k \rightarrow \mathbf{0}$ e $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}$, quando $k \rightarrow \infty$. O sistema (2.13) representa

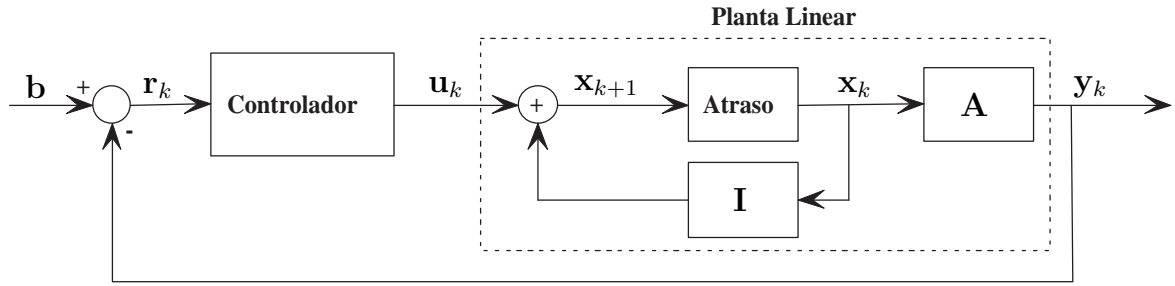


Figura 2.2: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (2.9) por métodos iterativos. O problema consiste em projetar um controlador tal que $\mathbf{r}_k \rightarrow \mathbf{0}$ e $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}$, quando $k \rightarrow \infty$.

a dinâmica deste problema de rastreamento;

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k \\ \mathbf{y}_k = \mathbf{A}\mathbf{x}_k \\ \mathbf{r}_k = \mathbf{b} - \mathbf{y}_k \\ \mathbf{u}_k = f(\mathbf{r}_k) \end{cases} \quad (2.13)$$

sendo que $f(\cdot)$ representa a relação entrada-saída do controlador (veja figura 2.2). Como já dito na introdução (capítulo 1), a interpretação de métodos iterativos para sistemas lineares como sistemas dinâmicos foi feita pela primeira vez em Schaerer e Kaszkurewicz [8] e para métodos iterativos lineares e não lineares em Bhaya e Kaszkurewicz [9, 10]. O método consiste basicamente em escolher uma função de Liapunov com controle (CLF, do inglês *Control Liapunov Function*) para determinar os parâmetros que tornam o equilíbrio do sistema (2.13) assintoticamente estável. Note que a cada escolha da estrutura do controlador \mathbf{u}_k corresponde uma família de métodos numéricos. Isso se deve ao fato de que, uma vez escolhida a estrutura do controlador, os parâmetros que garantem a estabilidade assintótica do equilíbrio do sistema são determinados a partir de uma escolha de CLF. A cada escolha de CLF correspondem parâmetros diferentes para uma mesma estrutura de controle \mathbf{u}_k . Vale ressaltar que a análise de métodos iterativos utilizando a técnica de CLFs de [9, 10] fornece uma idéia sobre a robustez dos métodos analisados e/ou desenvolvidos. Isto é devido ao fato de que a análise via CLF não utiliza as propriedades geométricas (ortogonalidade dos resíduos por exemplo) dos métodos considerados. A utilização dessa técnica em Diene [11] para a análise dos métodos iterativos do tipo Krylov mostrou que o método do gradiente con-

jugado (CG), com parâmetros determinados por CLF, converge mesmo quando a condição $\mathbf{r}_0 = \mathbf{p}_0$ (necessária para garantir a ortogonalidade dos resíduos) não é garantida. Essa situação ocorre nas aplicações práticas da filtragem adaptativa e do treinamento de perceptrons, que podem ser descritos como utilizando algoritmos “*em-linha*” do tipo CG para resolver sistemas do tipo $\mathbf{A}_k \mathbf{x}_k = \mathbf{b}_k$ e $\mathbf{A} \mathbf{x}_k = \mathbf{b}_k$. O termo “*em-linha*” refere-se à atualização da matriz \mathbf{A}_k e do segundo membro \mathbf{b}_k a cada iteração, e significa que, na prática, $\mathbf{r}_0 \neq \mathbf{p}_0$ depois da primeira iteração. Por esse motivo, utilizar-se-á, ao longo dessa tese, essa mesma técnica de CLF para analisar, projetar algoritmos numéricos para a filtragem adaptativa (seções 2.4 e 2.5), a filtragem adaptativa não supervisionada (capítulo 3) e para o treinamento de perceptrons (capítulo 4); e determinar uma medida da robustez dos algoritmos considerados (capítulo 5).

2.4 Algoritmos de filtragem adaptativa

2.4.1 Método LMS (*Least Mean Squares*)

Para resolver o problema da filtragem adaptativa definido por (2.7), os sistemas do tipo gradiente de descida são os mais utilizados devido à sua simplicidade de implementação [1, 4, 5]. O mínimo de (2.4), solução de (2.7), pode ser encontrado a partir de um ponto inicial arbitrário e uma seqüência na direção de descida do gradiente de (2.4) descrita da seguinte forma:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mu \nabla \xi = \mathbf{w}_k + \mu (\mathbf{b}_k - \mathbf{R}_k \mathbf{w}_k) \quad (2.14)$$

sendo o passo μ um pequeno valor real positivo. Utilizando estimativas instantâneas da matriz de autocorrelação \mathbf{R} e do vetor de correlação cruzada \mathbf{b} tais que

$$\mathbf{R}_k = \mathbf{x}_k \mathbf{x}_k^H \quad (2.15)$$

$$\mathbf{b}_k = \mathbf{x}_k d_k^*, \quad (2.16)$$

a equação (2.14) é reescrita conforme segue abaixo

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu (\mathbf{x}_k d_k^* - \mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k) = \mathbf{w}_k + \mu \mathbf{x}_k (d_k^* - \mathbf{x}_k^H \mathbf{w}_k) = \mathbf{w}_k + \mu \mathbf{x}_k e_k^* \quad (2.17)$$

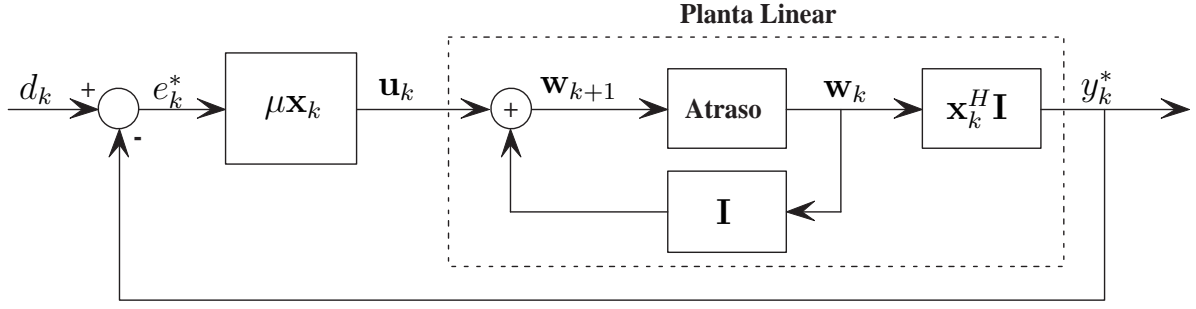


Figura 2.3: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema da filtragem adaptativa (2.7) pelo sistema dinâmico (2.18) correspondente ao método LMS. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $y_k \rightarrow d_k$, quando $k \rightarrow \infty$.

Algoritmo 2.2 Método LMS para filtros adaptativos

Para $k = 0, 1, \dots, N - 1$

$$e_k = d_k - \mathbf{w}_k^H \mathbf{x}_k$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu \mathbf{x}_k e_k^*$$

Fim

A equação (2.17) descreve o método conhecido como LMS (do inglês *Least Mean Squares*), apresentado na forma de pseudocódigo no algoritmo 2.2 [15–17, 50]. O método LMS pode ser interpretado como uma aproximação do método da declividade máxima (SD) utilizando estimativas instantâneas. Do ponto de vista de controle o método LMS pode ser descrito por um problema de rastreamento de sinal representado pela figura 2.3 e o sistema (2.18) abaixo

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \\ y_k = \mathbf{w}_k^H \mathbf{x}_k \\ e_k = d_k - y_k \\ \mathbf{u}_k = \mu \mathbf{x}_k e_k^* \end{cases} \quad (2.18)$$

O sistema dinâmico (2.18) é constituído pela planta linear variante no tempo $\{\mathbf{I}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, \mathbf{0}\}$ e o controlador proporcional ao erro $\{\mathbf{0}, \mathbf{0}, \mathbf{0}, \mu \mathbf{x}_k\}$ em realimentação negativa. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $y_k \rightarrow d_k$, quando $k \rightarrow \infty$. Note que a escolha planta/controlador não é única, porém a combinação da planta e do controlador escolhidos em malha fechada é única.

2.4.2 Método RLS (*Recursive Least Squares*)

O método LMS (algoritmo 2.2) apresenta um baixo custo computacional, porém a sua convergência depende da escolha do passo μ . A escolha de um passo grande leva à divergência do algoritmo enquanto a escolha de um passo pequeno resulta em uma convergência lenta do algoritmo LMS. Essa convergência lenta do algoritmo LMS é também observado quando o espalhamento dos autovalores da matriz \mathbf{R} é grande. Para esse caso, pode-se fazer uma estimativa da matriz de autocorrelação \mathbf{R} e do vetor de correlação cruzada \mathbf{b} levando em consideração as informações sobre os sinais em instantes anteriores através de uma janela de dados ponderada de forma exponencialmente decrescente por um fator λ_f , dito fator de esquecimento. Nesse caso \mathbf{R} e \mathbf{b} são obtidas conforme as equações (2.19) e (2.20)

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \quad (2.19)$$

$$\mathbf{b}_{k+1} = \lambda_f \mathbf{b}_k + \mathbf{x}_{k+1} d_{k+1}^* \quad (2.20)$$

A solução de (2.7) é tal que

$$\mathbf{w}_{k+1} = \mathbf{R}_{k+1}^{-1} \mathbf{b}_{k+1}, \quad (2.21)$$

porém a determinação direta da inversa de \mathbf{R}_{k+1} é computacionalmente caro (da ordem de $(M + 1)^3$). Essa inversa pode ser determinada de forma iterativa, através do lema da inversão de matrizes [51], da seguinte forma:

$$\mathbf{R}_{k+1}^{-1} = \lambda_f^{-1} \mathbf{R}_k^{-1} - \lambda_f^{-1} \frac{\mathbf{R}_k^{-1} \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1}}{\lambda_f + \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{x}_{k+1}}, \quad (2.22)$$

Definindo

$$\mathbf{v}_{k+1} = \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{x}_{k+1}}{1 + \lambda_f^{-1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{x}_{k+1}}, \quad (2.23)$$

que pode ser reescrito como

$$\begin{aligned} \mathbf{v}_{k+1} &= \lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{x}_{k+1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{x}_{k+1} \\ &= [\lambda_f^{-1} \mathbf{R}_k^{-1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1}] \mathbf{x}_{k+1} \\ &= \mathbf{R}_{k+1}^{-1} \mathbf{x}_{k+1}, \end{aligned} \quad (2.24)$$

a equação (2.22) pode ser reescrita de forma mais compacta conforme segue:

$$\mathbf{R}_{k+1}^{-1} = \lambda_f^{-1} \mathbf{R}_k^{-1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1}. \quad (2.25)$$

Substituindo (2.20), (2.25) e (2.24) em (2.21), resulta em

$$\begin{aligned} \mathbf{w}_{k+1} &= \mathbf{R}_{k+1}^{-1} \mathbf{b}_{k+1} \\ &= \lambda_f \mathbf{R}_{k+1}^{-1} \mathbf{b}_k + \mathbf{R}_{k+1}^{-1} \mathbf{x}_{k+1} d_{k+1}^* \\ &= \mathbf{R}_k^{-1} \mathbf{b}_k - \mathbf{v}_{k+1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{b}_k + \mathbf{v}_{k+1} d_{k+1}^* \\ &= \mathbf{w}_k + \mathbf{v}_{k+1} (d_{k+1}^* - \mathbf{x}_{k+1}^H \mathbf{w}_k) \\ &= \mathbf{w}_k + \mathbf{v}_{k+1} e_{k+1}^*, \end{aligned} \quad (2.26)$$

sendo que

$$e_{k+1} = d_{k+1} - \mathbf{w}_k^H \mathbf{x}_{k+1} \quad (2.27)$$

é o erro *a priori* [15](cap.8, pp.387). As equações (2.23), (2.25) e (2.26) descrevem o método conhecido na literatura como RLS (do inglês *Recursive Least Squares*) [15–17]. O método RLS, apresentado na forma de pseudocódigo no algoritmo 2.3, provê uma melhoria considerável na convergência do LMS, no entanto, o mesmo apresenta um custo computacional e uma necessidade de armazenamento maior que o LMS. Cabe enfatizar que uma fonte de divergência do método RLS é a perda da positividade da matriz inversa \mathbf{R}_{k+1}^{-1} devida aos erros numéricos. Uma forma de se evitar esse problema é manter a simetria de \mathbf{R}_{k+1}^{-1} utilizando a equação (2.22) ao invés da sua forma compacta (2.25) [52, 53]. Sob a perspectiva de controle, o método RLS pode ser interpretado como o sistema dinâmico descrito pelo sistema (2.28) e pela figura 2.4

$$\left\{ \begin{array}{l} \mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \\ y_{k+1} = \mathbf{w}_k^H \mathbf{x}_{k+1} \\ e_{k+1} = d_{k+1} - y_{k+1} \\ \mathbf{v}_{k+1} = \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{x}_{k+1}}{1 + \lambda_f^{-1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{x}_{k+1}} \\ \mathbf{u}_{k+1} = \mathbf{v}_{k+1} e_{k+1}^* \end{array} \right. \quad (2.28)$$

O sistema dinâmico (2.28) é composto pela planta linear variante no tempo $\{\mathbf{I}, \mathbf{I}, \mathbf{x}_{k+1}^H \mathbf{I}, \mathbf{0}\}$ e o controlador proporcional ao erro $\{\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{v}_{k+1}\}$ em realimentação negativa, sendo o vetor

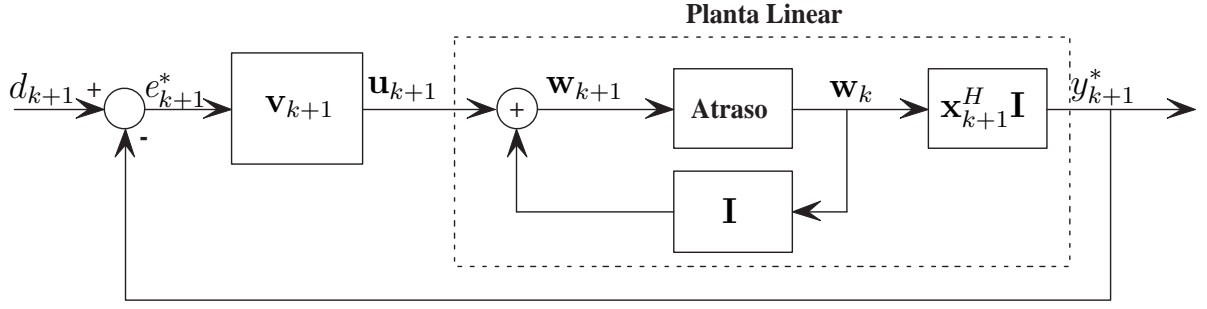


Figura 2.4: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (2.7) pelo sistema dinâmico (2.28) correspondente ao método RLS. O vetor de ganhos proporcionais \mathbf{v}_{k+1} é dado por (2.23).

Algoritmo 2.3 Método RLS para filtros adaptativos

Calcular $\mathbf{R}_0^{-1} = \mu^{-1}\mathbf{I}$, $\mu =$ pequena constante positiva

Para $k = 0, 1, \dots, N - 1$

$$\mathbf{v}_{k+1} = \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{x}_{k+1}}{1 + \lambda_f^{-1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{x}_{k+1}}$$

$$e_{k+1} = d_{k+1} - \mathbf{w}_k^H \mathbf{x}_{k+1}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{v}_{k+1} e_{k+1}^*$$

$$\mathbf{R}_{k+1}^{-1} = \lambda_f^{-1} \mathbf{R}_k^{-1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1}$$

Fim

de ganhos proporcionais \mathbf{v}_{k+1} dado por (2.23). Note que a planta é escolhida da mesma forma que o método LMS para poder comparar os dois métodos em uma base única.

2.4.3 Método SD-CLF (*Steepest Descent*)

Para resolver o problema da filtragem adaptativa (2.7), um novo método numérico, com desempenho comparável ao do LMS, pode ser desenvolvido considerando o problema (2.7) como um sistema dinâmico, conforme descrito na seção 2.3. Para tanto, reescreve-se a equação (2.14) tal que:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k (\mathbf{b}_k - \mathbf{R}_k \mathbf{w}_k) = \mathbf{w}_k + \alpha_k \mathbf{g}_k, \quad (2.29)$$

sendo α_k um passo variante a ser determinado e o resíduo $\mathbf{g}_k = \mathbf{b}_k - \mathbf{R}_k \mathbf{w}_k$. Para estimar a matriz \mathbf{R}_k e o vetor \mathbf{b}_k escolhe-se a janela de dados ponderada de forma exponencialmente

decrecente, dada pelas equações (2.19) e (2.20). Nesse caso, o resíduo é atualizado tal que:

$$\begin{aligned}
\mathbf{g}_{k+1} &= \mathbf{b}_{k+1} - \mathbf{R}_{k+1} \mathbf{w}_{k+1} \\
&= \lambda_f \mathbf{b}_k + \mathbf{x}_{k+1} d_{k+1}^* - (\lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H) (\mathbf{w}_k + \alpha_k \mathbf{g}_k) \\
&= \lambda_f (\mathbf{b}_k - \mathbf{R}_k \mathbf{w}_k) - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{x}_{k+1} (d_{k+1}^* - \mathbf{x}_{k+1}^H \mathbf{w}_k) \\
&= \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^*,
\end{aligned} \tag{2.30}$$

sendo o erro *a priori* e_{k+1} dado por (2.27). O sistema dinâmico (2.13) equivalente à resolução de (2.7) por métodos iterativos pode então ser reescrito como:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^* \end{cases} \tag{2.31}$$

Em termos de controle, o sistema (2.31), representado em forma de diagrama de blocos na figura 2.5, consiste em uma planta linear $\{\mathbf{I}, \mathbf{I}, \mathbf{R}_k, \mathbf{0}\}$ e um controlador proporcional ao erro em uma configuração de realimentação negativa. Note que, escolhendo \mathbf{g}_k e \mathbf{w}_k como variáveis de estado, e α_k como parâmetro de controle a ser determinado, o sistema (2.31) torna-se bilinear, isto é: linear no estado quando o controle é considerado constante e linear no controle quando o estado é considerado constante. Para determinar o equilíbrio do sistema (2.31), defina o erro ótimo

$$e_o = d_{k+1} - \mathbf{w}_o^H \mathbf{x}_{k+1}, \tag{2.32}$$

supondo estacionaridade dos sinais, existe um vetor ótimo de pesos \mathbf{w}_o tal que $\mathbf{R}_k \mathbf{w}_o = \mathbf{b}_k$ [54, 55]. Pelo princípio da ortogonalidade [15] tem-se que

$$E[\mathbf{x}_{k+1} e_o^*] = E[\mathbf{x}_{k+1} (d_{k+1}^* - \mathbf{x}_{k+1}^H \mathbf{w}_o)] = \mathbf{0}. \tag{2.33}$$

Utilizando $\mathbf{x}_{k+1} e_o^*$ como uma estimativa sem viés de $E[\mathbf{x}_{k+1} e_o^*]$ pode-se aproximar (2.33) da seguinte forma:

$$\mathbf{x}_{k+1} e_o^* = \mathbf{x}_{k+1} (d_{k+1}^* - \mathbf{x}_{k+1}^H \mathbf{w}_o) = \mathbf{0}. \tag{2.34}$$

Portanto, pode-se ver que o equilíbrio do sistema (2.31) é dado por $(\mathbf{w}_o, \mathbf{0})$. Para esse mesmo sistema (2.31), um resultado desta tese pode ser enunciado da seguinte forma [21]:

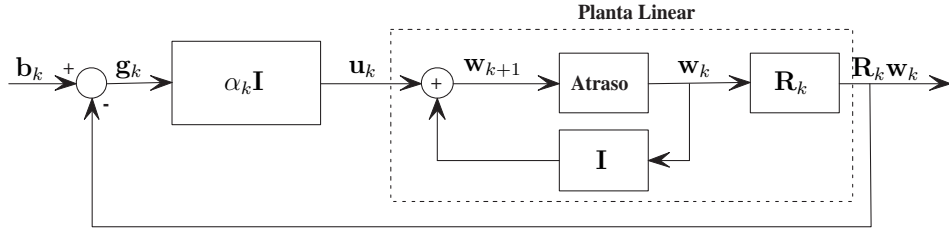


Figura 2.5: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa (2.7) pelo sistema dinâmico (2.31) correspondente ao método SD-CLF. O problema consiste em determinar o parâmetro de controle α_k tal que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{R}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$.

Teorema 1. Com a hipótese $\inf_k \lambda_{\min}(\mathbf{R}_k) = \zeta_1 > 0$, a escolha

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}{\langle \mathbf{R}_{k+1} \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}, \quad (2.35)$$

sendo $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^*$, é ótima para a função de Liapunov escolhida e garante a estabilidade assintótica do equilíbrio do sistema (2.31).

Prova do Teorema 1. Seja $\mathcal{X}_K = \{\mathbf{x}_k, k = 1, \dots, K\}$ o conjunto de todos os dados \mathbf{x}_k , sendo K um número grande. Para um processamento amostra por amostra em tempo real define-se um subconjunto de \mathcal{X}_k da seguinte forma:

$$\hat{\mathcal{X}}_k = \{\hat{\mathbf{x}}_i : \hat{\mathbf{x}}_i = \mathbf{x}_i, \text{ para } i \leq k, \text{ e } \hat{\mathbf{x}}_i = \mathbf{0} \text{ para } i > k\}.$$

Neste caso, na iteração $k + 1$, tem-se que

$$\hat{\mathcal{X}}_{k+1} = \{\hat{\mathbf{x}}_1 = \mathbf{x}_1, \dots, \hat{\mathbf{x}}_{k+1} = \mathbf{x}_{k+1}, \hat{\mathbf{x}}_{k+2} = \mathbf{0}, \dots, \hat{\mathbf{x}}_K = \mathbf{0}\}, \quad (2.36)$$

e quando K tende a infinito,

$$\hat{\mathcal{X}}_K = \mathcal{X}_K.$$

Portanto a cada iteração k o conjunto de dados a processar é dado por $\hat{\mathcal{X}}_k$.

Na iteração $k + 1$, (2.30) pode ser reescrita como

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k, \quad (2.37)$$

sendo

$$\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \hat{\mathbf{x}}_{k+1} e_{k+1}^* = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^*. \quad (2.38)$$

Note que o vetor $\tilde{\mathbf{g}}_k$ dado por (2.38) é calculado na iteração $k + 1$. Na iteração $k + 1$, de (2.38) e (2.37), pode-se escrever que

$$\begin{aligned} \tilde{\mathbf{g}}_{k+1} &= \lambda_f \mathbf{g}_{k+1} + \hat{\mathbf{x}}_{k+2} e_{k+2}^* \\ &= \lambda_f \mathbf{g}_{k+1} = \lambda_f \tilde{\mathbf{g}}_k - \alpha_k \lambda_f \mathbf{R}_{k+1} \mathbf{g}_k, \end{aligned} \quad (2.39)$$

uma vez que, de (2.36), $\hat{\mathbf{x}}_{k+2} = \mathbf{0}$ na iteração $k + 1$. Então, o sistema (2.31) pode ser reescrito como

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{g}_k \\ \tilde{\mathbf{g}}_{k+1} = \lambda_f \tilde{\mathbf{g}}_k - \alpha_k \lambda_f \mathbf{R}_{k+1} \mathbf{g}_k. \end{cases} \quad (2.40)$$

O equilíbrio do sistema (2.40) é dado por $(\mathbf{w}_o, \mathbf{0})$. Note que, no equilíbrio do sistema (2.31), tem-se que $\mathbf{w}_k = \mathbf{w}_o$ e $\mathbf{g}_k = \mathbf{0}$, o que implica que (de (2.34)) $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_o^* = 0$. Portanto a estabilidade assintótica do equilíbrio de (2.40) resulta na estabilidade assintótica de (2.31). O objetivo do restante dessa prova é mostrar a estabilidade assintótica do sistema (2.40). Escolha a CLF $V_{\tilde{\mathbf{g}}_k}(k) = \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle$, então

$$\Delta V_{\tilde{\mathbf{g}}} = \langle \tilde{\mathbf{g}}_{k+1}, \tilde{\mathbf{g}}_{k+1} \rangle - \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle. \quad (2.41)$$

Substituindo (2.39) em (2.41) resulta em

$$\Delta V_{\tilde{\mathbf{g}}} = \lambda_f^2 \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle - 2\alpha_k \lambda_f^2 \langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle + \alpha_k^2 \lambda_f^2 \langle \mathbf{R}_{k+1} \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle - \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle \quad (2.42)$$

O valor ótimo α_k (que torna $\Delta V_{\tilde{\mathbf{g}}}$ o mais negativo possível), dado por (2.35), é determinado resolvendo a seguinte equação:

$$\frac{\partial \Delta V_{\tilde{\mathbf{g}}}}{\partial \alpha_k} = -2\lambda_f^2 \langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle + 2\alpha_k \lambda_f^2 \langle \mathbf{R}_{k+1} \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle = 0$$

Substituindo (2.35) em (2.42) resulta em

$$\Delta V_{\tilde{\mathbf{g}}} = -\lambda_f^2 \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle^2}{\langle \mathbf{R}_{k+1} \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle} - (1 - \lambda_f^2) \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle \quad (2.43)$$

Algoritmo 2.4 Método SD-CLF para filtros adaptativos

Calcular $\mathbf{R}_0 = \mathbf{x}_0 \mathbf{x}_0^H$, $\mathbf{b}_0 = \mathbf{x}_0 d_0^*$, $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$

Para $k = 0, 1, \dots, N - 1$

$$e_{k+1} = d_{k+1} - \mathbf{w}_k^H \mathbf{x}_{k+1}$$

$$\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^*$$

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H$$

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}{\langle \mathbf{R}_{k+1} \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{g}_k$$

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k$$

Fim

Como, por hipótese $\langle \mathbf{R}_{k+1} \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle > 0$, segue-se que $\lambda_f \leq 1$ implica que

$$\Delta V_{\tilde{\mathbf{g}}} < 0, \quad (2.44)$$

o que implica que $\|\tilde{\mathbf{g}}_k\|_2^2$ é uma seqüência decrescente. Portanto $\tilde{\mathbf{g}}_k \rightarrow \mathbf{0}$ para $k \rightarrow \infty$, o que implica que $\lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^* \rightarrow 0$ para $k \rightarrow \infty$, que por sua vez, implica que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{w}_k \rightarrow \mathbf{w}_o$ para $k \rightarrow \infty$. ■

O novo método numérico baseado no sistema (2.31) e no teorema 1, apresentado na forma de pseudocódigo no algoritmo 2.4 e referenciado nesta tese como SD-CLF, é um método do tipo declividade máxima (*Steepest Descent*).

Observação 1. As simulações numéricas da seção 2.6 que verificam numericamente os resultados do teorema 1 consideram uma média aritmética de várias simulações.

Observação 2. Note que a escolha da CLF não é única, a escolha de outra CLF ($V_{\tilde{\mathbf{g}}_k}(k) = \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle$, por exemplo) resulta em uma fórmula diferente do parâmetro α_k ($\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{g}_k \rangle}{\langle \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}$, por exemplo) e em outro método da mesma família SD.

2.4.4 Método CG-CLF (*Conjugate Gradient*)

As simulações numéricas, apresentadas na seção 2.6, mostram que o método SD-CLF apresenta um desempenho superior ao LMS e inferior ao RLS. Sabe-se que a convergência do método de declividade máxima torna-se lenta para matrizes mal condicionadas. Essa convergência pode ser acelerada introduzindo um termo proporcional à derivada discreta do estado \mathbf{w}_k na equação (2.29). A introdução de um termo derivativo no método da declividade

mxima resulta no mtodo do gradiente conjugado (CG) [10, 12]. O mtodo CG foi utilizado em Boray e Srinath [18] e Chang e Willson [19] para projetar filtros adaptativos com uma convergncia comparvel  do mtodo RLS, com um custo computacional menor. Em Boray e Srinath [18] as matrizes \mathbf{R} e \mathbf{b} so estimadas utilizando uma janela deslizante de dados de tamanho n_w tal que

$$\mathbf{R}_k = \frac{1}{n_w} \sum_{j=k-n_w+1}^k \mathbf{x}_j \mathbf{x}_j^H \quad (2.45)$$

$$\mathbf{b}_k = \frac{1}{n_w} \sum_{j=k-n_w+1}^k \mathbf{x}_j d_j^* \quad (2.46)$$

Nesse caso o resduo, igual ao negativo do gradiente da funo objetivo (2.4),  estimado por uma mdia de n_w valores passados conforme a seguinte equao

$$\mathbf{g}_k = \left(\frac{2}{n_w} \right) \sum_{j=k-n_w+1}^k \mathbf{x}_j (d_j - \mathbf{w}_k^H \mathbf{x}_j)^* \quad (2.47)$$

Note que para cada amostra k so necessrios $\min(M, n_w)$ iteraes para a convergncia do algoritmo CG. Portanto, a escolha de n_w , a janela sobre qual a mdia do gradiente da funo custo  avaliada, tem um impacto direto sobre a convergncia dos pesos do filtro. Desta forma, um aumento na dimenso do *tap* e/ou dos dados de entrada causa diretamente um aumento no custo computacional do mtodo, podendo ultrapassar o custo computacional do mtodo RLS e da inverso da matriz \mathbf{R} . No trabalho de Chang e Willson [19], uma modificao  feita determinando a matriz de autocorrelao \mathbf{R} e o vetor de correlao cruzada \mathbf{b} atravs da mesma janela de dados ponderada de forma exponencialmente decrescente utilizada pelo mtodo RLS (equaes (2.19) e (2.20)). Esta modificao permite realizar apenas uma iterao do mtodo CG a cada amostragem, porm para manter a convergncia  necessrio a introduo de um fator η , determinado experimentalmente, no clculo do parmetro α_k (do mtodo CG), e do mtodo Polak-Ribiere para o clculo do parmetro β_k . Nesta tese mostra-se que, desenvolvendo mtodos de filtragem adaptativa atravs de CLF's, no  necessrio a introduo de nenhum fator emprico e/ou de nenhum outro mtodo no linear para garantir a convergncia dos mtodos em ambientes variantes no tempo [56]. Cabe enfatizar que a convergncia dos algoritmos propostos em Boray e Srinath [18], Chang e Willson [19]  conjecturada atravs de simulaes numricas enquanto

que os algoritmos de filtragem adaptativa propostos nesta tese são reforçados por uma prova formal de convergência, uma prova da otimalidade dos parâmetros escolhidos e simulações numéricas.

Para desenvolver um novo método CG para a filtragem adaptativa através de CLF's reescreve-se o sistema dinâmico (2.13), equivalente à resolução de (2.7) por métodos numéricos, da seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \mathbf{b}_{k+1} - \mathbf{R}_{k+1} \mathbf{w}_{k+1} \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k \end{cases} \quad (2.48)$$

Utilizando a janela de dados ponderada de forma exponencialmente decrescente (2.19, 2.20), pode-se escrever

$$\begin{aligned} \mathbf{g}_{k+1} &= \mathbf{b}_{k+1} - \mathbf{R}_{k+1} \mathbf{w}_{k+1} = \lambda_f \mathbf{b}_k + \mathbf{x}_{k+1} d_{k+1}^* - (\lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H) (\mathbf{w}_k + \alpha_k \mathbf{p}_k) \\ &= \lambda_f (\mathbf{b}_k - \mathbf{R}_k \mathbf{w}_k) - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} (d_{k+1}^* - \mathbf{x}_{k+1}^H \mathbf{w}_k) \\ &= \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} e_{k+1}^*, \end{aligned} \quad (2.49)$$

sendo novamente e_{k+1} o erro *a priori* dado por (2.27). Substituindo (2.49) em (2.48) resulta em:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} e_{k+1}^* \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k \end{cases} \quad (2.50)$$

Analogamente ao sistema (2.31), escolhendo \mathbf{w}_k , \mathbf{g}_k e \mathbf{p}_k como variáveis de estado, e α_k e β_k como parâmetros de controle a serem determinados, o sistema (2.50) torna-se bilinear. O equilíbrio do sistema (2.50) é dado por $(\mathbf{w}_o, \mathbf{0}, \mathbf{0})$, isto pode ser visto utilizando novamente a equação (2.34). Para o sistema (2.50), representado na forma de diagrama de blocos na figura 2.6, um dos principais resultados dessa tese pode ser enunciado da seguinte forma [21]:

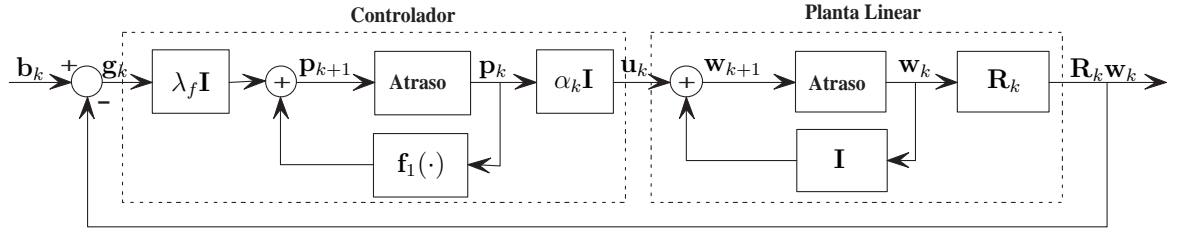


Figura 2.6: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa (2.7) pelo sistema dinâmico (2.50) correspondente ao método CG-CLF, sendo $f_1(\mathbf{p}_k) = \beta_k \mathbf{p}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} e_{k+1}^*$. O problema consiste em determinar os parâmetros de controle α_k e β_k tais que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{R}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$.

Teorema 2. Com as hipóteses $\inf_k \lambda_{\min}(\mathbf{R}_k^{-1}) = \zeta_1 > 0$ e $\inf_k \lambda_{\min}(\mathbf{R}_k) = \zeta_2 > 0$, as escolhas

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} \quad (2.51)$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}, \quad (2.52)$$

sendo $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^*$, são ótimas para as funções de Liapunov escolhidas e garantem a estabilidade assintótica do equilíbrio do sistema (2.50).

Prova do Teorema 2. Seja $\mathcal{X}_K = \{\mathbf{x}_k, k = 1, \dots, K\}$ o conjunto de todos os dados \mathbf{x}_k , sendo K um número grande. Para um processamento amostra por amostra em tempo real define-se um subconjunto de \mathcal{X}_k da seguinte forma:

$$\hat{\mathcal{X}}_k = \{\hat{\mathbf{x}}_i : \hat{\mathbf{x}}_i = \mathbf{x}_i, \text{ para } i \leq k, \text{ e } \hat{\mathbf{x}}_i = \mathbf{0} \text{ para } i > k\}.$$

Neste caso, na iteração $k + 1$, tem-se que

$$\hat{\mathcal{X}}_{k+1} = \{\hat{\mathbf{x}}_1 = \mathbf{x}_1, \dots, \hat{\mathbf{x}}_{k+1} = \mathbf{x}_{k+1}, \hat{\mathbf{x}}_{k+2} = \mathbf{0}, \dots, \hat{\mathbf{x}}_K = \mathbf{0}\}, \quad (2.53)$$

e quando K tende para infinito,

$$\hat{\mathcal{X}}_K = \mathcal{X}_K.$$

Portanto a cada iteração k o conjunto de dados a processar é dado por $\hat{\mathcal{X}}_k$.

Na iteração $k + 1$, (2.49) pode ser reescrita como

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k, \quad (2.54)$$

sendo

$$\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \hat{\mathbf{x}}_{k+1} e_{k+1}^* = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^*. \quad (2.55)$$

Note que o vetor $\tilde{\mathbf{g}}_k$ dado por (2.55) é calculado na iteração $k + 1$. Na iteração $k + 1$, de (2.55) e (2.54), pode-se escrever que

$$\begin{aligned} \tilde{\mathbf{g}}_{k+1} &= \lambda_f \mathbf{g}_{k+1} + \hat{\mathbf{x}}_{k+2} e_{k+2}^* \\ &= \lambda_f \mathbf{g}_{k+1} = \lambda_f \tilde{\mathbf{g}}_k - \alpha_k \lambda_f \mathbf{R}_{k+1} \mathbf{p}_k, \end{aligned} \quad (2.56)$$

uma vez que, de (2.53), $\hat{\mathbf{x}}_{k+2} = \mathbf{0}$ na iteração $k + 1$. Então, o sistema (2.50) pode ser reescrito como

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \tilde{\mathbf{g}}_{k+1} = \lambda_f \tilde{\mathbf{g}}_k - \alpha_k \lambda_f \mathbf{R}_{k+1} \mathbf{p}_k \\ \mathbf{p}_{k+1} = \frac{1}{\lambda_f} \tilde{\mathbf{g}}_{k+1} + \beta_k \mathbf{p}_k. \end{cases} \quad (2.57)$$

O equilíbrio do sistema (2.57) é dado por $(\mathbf{w}_o, \mathbf{0}, \mathbf{0})$. Note que, no equilíbrio do sistema (2.50), tem-se que $\mathbf{w}_k = \mathbf{w}_o$, $\mathbf{g}_k = \mathbf{0}$ e $\mathbf{p}_k = \mathbf{0}$, o que implica que (de (2.34)) $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_o^* = 0$. Portanto a estabilidade assintótica do equilíbrio de (2.57) resulta na estabilidade assintótica de (2.50). O objetivo do restante dessa prova é mostrar a estabilidade assintótica do sistema (2.57). Como a matriz \mathbf{R}_k é hermitiana positiva definida, ela é invertível e define uma norma, da mesma forma que sua inversa. Portanto escolha a CLF $V_{\tilde{\mathbf{g}}_k}(k) = \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle$, então

$$\Delta V_{\tilde{\mathbf{g}}} = \langle \tilde{\mathbf{g}}_{k+1}, \mathbf{R}_{k+1}^{-1} \tilde{\mathbf{g}}_{k+1} \rangle - \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle. \quad (2.58)$$

Substituindo (2.56) em (2.58) resulta em

$$\Delta V_{\tilde{\mathbf{g}}} = \lambda_f^2 \langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1}^{-1} \tilde{\mathbf{g}}_k \rangle - 2\alpha_k \lambda_f^2 \langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle + \alpha_k^2 \lambda_f^2 \langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle - \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle \quad (2.59)$$

O valor ótimo α_k (que torna $\Delta V_{\tilde{\mathbf{g}}}$ o mais negativo possível), dado por (2.51), é determinado

resolvendo a seguinte equação:

$$\frac{\partial \Delta V_{\tilde{\mathbf{g}}}}{\partial \alpha_k} = -2\lambda_f^2 \langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle + 2\alpha_k \lambda_f^2 \langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle = 0$$

Substituindo (2.51) em (2.59) resulta em

$$\Delta V_{\tilde{\mathbf{g}}} = -\lambda_f^2 \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} + \lambda_f^2 \langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1}^{-1} \tilde{\mathbf{g}}_k \rangle - \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle \quad (2.60)$$

Reescreva a equação da inversa de \mathbf{R}_{k+1} (2.25) tal que

$$\mathbf{R}_{k+1}^{-1} = \lambda_f^{-1} \mathbf{R}_k^{-1} - c \lambda_f^{-2} \mathbf{R}_k^{-1} \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1}, \quad (2.61)$$

sendo $c = (1/(1 + \lambda_f^{-1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{x}_{k+1})) > 0$. Inserindo (2.61) em (2.60) resulta em

$$\Delta V_{\tilde{\mathbf{g}}} = -\lambda_f^2 \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} - (1 - \lambda_f) \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle - c \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle. \quad (2.62)$$

Portanto, por hipótese, segue-se que $\lambda_f \leq 1$ implica que

$$\Delta V_{\tilde{\mathbf{g}}} < -\zeta_1 \|\tilde{\mathbf{g}}_k\|_2^2 < 0, \quad (2.63)$$

o que implica que $\|\tilde{\mathbf{g}}_k\|_{\mathbf{R}_k^{-1}}$ é uma seqüência decrescente. \square

Para a determinação de β_k , escolha a CLF $V_{\mathbf{p}_k}(k) = \langle \mathbf{p}_k, \mathbf{R}_k \mathbf{p}_k \rangle = \|\mathbf{p}_k\|_{\mathbf{R}_k}^2$. Então,

$$\Delta V_{\mathbf{p}} = \frac{1}{\lambda^2} \langle \tilde{\mathbf{g}}_{k+1}, \mathbf{R}_{k+1} \tilde{\mathbf{g}}_{k+1} \rangle + \frac{2}{\lambda} \beta_k \langle \tilde{\mathbf{g}}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle + \beta_k^2 \langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle - \langle \mathbf{p}_k, \mathbf{R}_k \mathbf{p}_k \rangle.$$

Nesse caso, o valor ótimo de β_k , dado por (2.52), é obtido resolvendo a seguinte equação

$$\frac{\partial \Delta V_{\mathbf{p}}}{\partial \beta_k} = \frac{2}{\lambda} \langle \tilde{\mathbf{g}}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle + 2\beta_k \langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle = 0.$$

De (2.57), inserindo (2.52) em $V_{\mathbf{p}_{k+1}}(k+1)$ resulta em

$$\begin{aligned} V_{\mathbf{p}_{k+1}}(k+1) &= \langle \mathbf{p}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_{k+1} \rangle = \frac{1}{\lambda^2} \|\tilde{\mathbf{g}}_{k+1}\|_{\mathbf{R}_{k+1}}^2 - \frac{1}{\lambda^2} \frac{\langle \tilde{\mathbf{g}}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} \\ &< \frac{1}{\lambda^2} \|\tilde{\mathbf{g}}_{k+1}\|_{\mathbf{R}_{k+1}}^2. \end{aligned} \quad (2.64)$$

Algoritmo 2.5 Método CG-CLF para filtros adaptativos

Calcular $\mathbf{R}_0 = \mathbf{x}_0 \mathbf{x}_0^H$, $\mathbf{b}_0 = \mathbf{x}_0 d_0^*$, $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$, $\mathbf{p}_0 = \mathbf{g}_0$

Para $k = 0, 1, \dots, N - 1$

$$e_{k+1} = d_{k+1} - \mathbf{w}_k^H \mathbf{x}_{k+1}$$

$$\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^*$$

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H$$

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k$$

Fim

De (2.63), da equivalência de normas e da hipótese (1), pode-se concluir que $\|\tilde{\mathbf{g}}_{k+1}\|_{\mathbf{R}_{k+1}}$ é uma seqüência decrescente. Portanto (2.64) implica que \mathbf{p}_{k+1} decresce na norma \mathbf{R}_{k+1} , não necessariamente monotonicamente, concluindo a prova. ■

Observação 3. Em Bottomley e Alexander [52] e de Campos [53] mostra-se que uma possibilidade de garantir que $\inf_k \lambda_{\min}(\mathbf{R}_k^{-1}) = \zeta_1 > 0$ é mantendo a simetria de \mathbf{R}_k^{-1} através da utilização da fórmula (2.22) ao invés da sua forma compacta (2.25).

O algoritmo 2.5, referenciado nesta tese como CG-CLF, apresenta o pseudocódigo da extensão do método CG, baseada no teorema 2, para o caso da filtragem adaptativa.

2.5 Algoritmos de filtragem adaptativa com restrições lineares

2.5.1 Método CLMS (Constrained Least Mean Squares)

O problema da filtragem adaptativa com restrições lineares (2.8) pode ser resolvido através de técnicas de otimização com restrições utilizando multiplicadores de Lagrange. Para tanto a função custo (2.4) é redefinida tal que

$$\psi = \frac{1}{2} (E[d_k d_k^*] - \mathbf{w}_k^H \mathbf{b}_k - \mathbf{b}_k^H \mathbf{w}_k + \mathbf{w}_k^H \mathbf{R}_k \mathbf{w}_k) + \lambda_k^H (\mathbf{C}^H \mathbf{w}_k - \mathbf{f}), \quad (2.65)$$

sendo λ_k o vetor de multiplicadores de Lagrange. Nesse caso o problema com restrições é transformado em um problema sem restrições da seguinte forma [57]:

$$\min_{\mathbf{w}_k} \{ \psi = \frac{1}{2} (E[d_k d_k^*] - \mathbf{w}_k^H \mathbf{b}_k - \mathbf{b}_k^H \mathbf{w}_k + \mathbf{w}_k^H \mathbf{R}_k \mathbf{w}_k) + \lambda_k^H (\mathbf{C}^H \mathbf{w}_k - \mathbf{f}) \}. \quad (2.66)$$

A solução de (2.66) é determinada a partir de uma aproximação inicial e uma seqüência na direção de descida do gradiente de ψ dada pela equação (2.67)

$$\begin{aligned} \mathbf{w}_{k+1} &= \mathbf{w}_k - \mu \nabla_{\mathbf{w}_k} \psi \\ &= \mathbf{w}_k - \mu (\mathbf{R}_k \mathbf{w}_k - \mathbf{b}_k + \mathbf{C} \lambda_k). \end{aligned} \quad (2.67)$$

O vetor de multiplicadores de Lagrange λ_k é escolhido tal que o vetor \mathbf{w}_{k+1} satisfaça a restrição linear $\mathbf{C}^H \mathbf{w}_{k+1} = \mathbf{f}$, isto é:

$$\mathbf{C}^H \mathbf{w}_{k+1} = \mathbf{f} \Leftrightarrow \mathbf{C}^H \mathbf{w}_k - \mu (\mathbf{C}^H \mathbf{R}_k \mathbf{w}_k - \mathbf{C}^H \mathbf{b}_k + \mathbf{C}^H \mathbf{C} \lambda_k) = \mathbf{f}.$$

Portanto, obtém-se o vetor de multiplicadores de Lagrange tal que:

$$\lambda_k = \frac{-1}{\mu} (\mathbf{C}^H \mathbf{C})^{-1} (\mathbf{f} - \mathbf{C}^H \mathbf{w}_k) - (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H \mathbf{R}_k \mathbf{w}_k + (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H \mathbf{b}_k. \quad (2.68)$$

Substituindo (2.68) em (2.67) resulta em

$$\begin{aligned} \mathbf{w}_{k+1} &= \mathbf{w}_k - \mu (\mathbf{R}_k \mathbf{w}_k - \mathbf{b}_k) + \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} (\mathbf{f} - \mathbf{C}^H \mathbf{w}_k) + \mu \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H (\mathbf{R}_k \mathbf{w}_k - \mathbf{b}_k) \\ &= (\mathbf{I} - \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H) \mathbf{w}_k - \mu (\mathbf{I} - \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H) (\mathbf{R}_k \mathbf{w}_k - \mathbf{b}_k) + \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{f} \\ &= \mathbf{P} \mathbf{w}_k - \mu \mathbf{P} (\mathbf{R}_k \mathbf{w}_k - \mathbf{b}_k) + \bar{\mathbf{f}}, \end{aligned} \quad (2.69)$$

sendo

$$\mathbf{P} = \mathbf{I} - \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H \quad (2.70)$$

$$\bar{\mathbf{f}} = \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{f}. \quad (2.71)$$

Note que \mathbf{P} é a matriz de projeção no complemento ortogonal do espaço coluna de \mathbf{C} , portanto \mathbf{P} é a matriz de projeção no espaço nulo a esquerda de \mathbf{C} [58]. Escolhendo as estima-

Algoritmo 2.6 Método CLMS para filtros adaptativos com restrições lineares

Calcular $\bar{\mathbf{f}} = \mathbf{C}(\mathbf{C}^H \mathbf{C})^{-1} \mathbf{f}$, $\mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H$ Para $k = 0, 1, \dots, N - 1$

$$e_k = d_k - \mathbf{w}_k^H \mathbf{x}_k$$

$$\mathbf{w}_{k+1} = \mathbf{P}(\mathbf{w}_k + \mu \mathbf{x}_k e_k^*) + \bar{\mathbf{f}}$$

Fim

tivas instantâneas (2.15) e (2.16) de \mathbf{R}_k e \mathbf{b}_k , a equação (2.69) resulta em:

$$\begin{aligned} \mathbf{w}_{k+1} &= \mathbf{P} \mathbf{w}_k - \mu \mathbf{P}(\mathbf{R}_k \mathbf{w}_k - \mathbf{b}_k) + \bar{\mathbf{f}} \\ &= \mathbf{P} \mathbf{w}_k - \mu \mathbf{P}(\mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k - \mathbf{x}_k d_k^*) + \bar{\mathbf{f}} \\ &= \mathbf{P} \mathbf{w}_k + \mu \mathbf{P} \mathbf{x}_k e_k^* + \bar{\mathbf{f}} = \mathbf{P}(\mathbf{w}_k + \mu \mathbf{x}_k e_k^*) + \bar{\mathbf{f}} \end{aligned} \quad (2.72)$$

As equações (2.70-2.72) descrevem o método conhecido como CLMS (do inglês *Constrained Least Mean Square*), proposto pela primeira vez em Frost III [59]. Como no caso do método LMS para a filtragem adaptativa, o método CLMS, apresentado na forma de pseudocódigo no algoritmo 2.6, é um dos mais utilizados para solucionar o problema da filtragem adaptativa com restrições lineares devido à sua simplicidade de implementação [60, 61]. Do ponto de vista de sistemas dinâmicos o método CLMS pode ser representado pelo sistema da figura 2.7 e o sistema de equações (2.73)

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \mathbf{u}_k \\ y_k = \mathbf{w}_k^H \mathbf{x}_k \\ e_k = d_k - \mathbf{w}_k^H \mathbf{x}_k \\ \mathbf{u}_k = \mu \mathbf{P} \mathbf{x}_k e_k^* + \bar{\mathbf{f}} \end{cases} \quad (2.73)$$

O sistema dinâmico da figura 2.7 é composto pela planta linear variante no tempo $\{\mathbf{P}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, \mathbf{0}\}$ em realimentação negativa com o controlador não linear $\mathbf{f}_1(e_k^*) = \mu \mathbf{P} \mathbf{x}_k e_k^* + \bar{\mathbf{f}}$ (correspondente a uma função afim do erro). Novamente o problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $y_k \rightarrow d_k$, quando $k \rightarrow \infty$.

2.5.2 Método CRLS (*Constrained Recursive Least Squares*)

O método CLMS apresenta as mesmas vantagens e desvantagens do método LMS: um baixo custo computacional e uma convergência dependente da escolha do passo μ e do es-

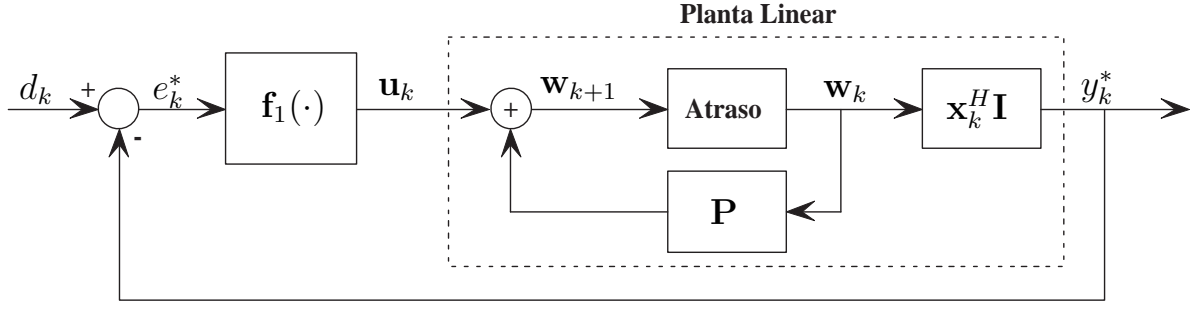


Figura 2.7: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (2.73) correspondente ao método CLMS. O controle é uma função afim do erro, $f_1(e_k^*) = \mu P x_k e_k^* + \bar{f}$. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $y_k \rightarrow d_k$, quando $k \rightarrow \infty$.

palhamento dos autovalores da matriz R_k . Para melhorar a taxa de convergência pode-se fazer estimativas melhores de R_k e b_k levando em consideração as informações dos sinais em instantes anteriores através da janela de dados ponderada de forma exponencialmente decrescente (2.19, 2.20). De (2.66) sabe-se que a solução de (2.8) é tal que:

$$w_{k+1} = R_{k+1}^{-1} b_{k+1} - R_{k+1}^{-1} C \lambda_{k+1}. \quad (2.74)$$

Da mesma forma que o caso do método CLMS, o vetor de multiplicadores de Lagrange λ_{k+1} é escolhido de tal forma que o vetor w_{k+1} satisfaça a restrição linear, isto é:

$$C^H w_{k+1} = f \Leftrightarrow C^H R_{k+1}^{-1} b_{k+1} - C^H R_{k+1}^{-1} C \lambda_{k+1} = f.$$

Então, determina-se o vetor de multiplicadores de Lagrange da seguinte forma:

$$\lambda_{k+1} = (C^H R_{k+1}^{-1} C)^{-1} (C^H R_{k+1}^{-1} b_{k+1} - f) \quad (2.75)$$

Substituindo (2.75) em (2.74) resulta em

$$w_{k+1} = R_{k+1}^{-1} b_{k+1} + R_{k+1}^{-1} C (C^H R_{k+1}^{-1} C)^{-1} (f - C^H R_{k+1}^{-1} b_{k+1}). \quad (2.76)$$

O vetor w_{k+1} pode ser determinado de recursivamente notando que $R_{k+1}^{-1} b_{k+1}$ corresponde à solução do problema irrestrito (2.7) e utilizando o lemma da inversão de matrizes (2.25)

para calcular as matrizes $\mathbf{R}_{k+1}^{-1}\mathbf{C}$ e $(\mathbf{C}^H\mathbf{R}_{k+1}^{-1}\mathbf{C})^{-1}$. Para tanto define-se

$$\begin{aligned}\mathbf{\Gamma}_{k+1} &:= \mathbf{R}_{k+1}^{-1}\mathbf{C} = \lambda_f^{-1}\mathbf{R}_k^{-1}\mathbf{C} - \lambda_f^{-1}\mathbf{v}_{k+1}\mathbf{x}_{k+1}^H\mathbf{R}_k^{-1}\mathbf{C} \\ &= \lambda_f^{-1}\mathbf{\Gamma}_k - \lambda_f^{-1}\mathbf{v}_{k+1}\mathbf{x}_{k+1}^H\mathbf{\Gamma}_k\end{aligned}\quad (2.77)$$

$$\begin{aligned}\mathbf{\Psi}_{k+1} &:= \mathbf{C}^H\mathbf{\Gamma}_{k+1} = \lambda_f^{-1}\mathbf{C}^H\mathbf{\Gamma}_k - \lambda_f^{-1}\mathbf{C}^H\mathbf{v}_{k+1}\mathbf{x}_{k+1}^H\mathbf{\Gamma}_k \\ &= \lambda_f^{-1}\mathbf{\Psi}_k - \lambda_f^{-1}\mathbf{C}^H\mathbf{v}_{k+1}\mathbf{x}_{k+1}^H\mathbf{\Gamma}_k.\end{aligned}\quad (2.78)$$

Então, a matriz inversa $\mathbf{\Psi}_{k+1}^{-1} = (\mathbf{C}^H\mathbf{R}_{k+1}^{-1}\mathbf{C})^{-1}$ é determinada através da seguinte equação

$$\mathbf{\Psi}_{k+1}^{-1} = \lambda_f\mathbf{\Psi}_k^{-1} + \lambda_f\mathbf{l}_{k+1}\mathbf{x}_{k+1}^H\mathbf{\Gamma}_k\mathbf{\Psi}_k^{-1},\quad (2.79)$$

sendo

$$\mathbf{l}_{k+1} = \frac{\mathbf{\Psi}_k^{-1}\mathbf{C}^H\mathbf{v}_{k+1}}{1 - \mathbf{x}_{k+1}^H\mathbf{\Gamma}_k\mathbf{\Psi}_k^{-1}\mathbf{C}^H\mathbf{v}_{k+1}}\quad (2.80)$$

A solução do problema irrestrito é determinada recursivamente pelo método RLS tal que

$$\mathbf{R}_{k+1}^{-1}\mathbf{b}_{k+1} = \mathbf{w}_k + \mathbf{v}_{k+1}e_{k+1}^*.\quad (2.81)$$

Inserindo (2.81) em (2.76) tem-se

$$\begin{aligned}\mathbf{w}_{k+1} &= \mathbf{w}_k + \mathbf{v}_{k+1}e_{k+1}^* + \mathbf{R}_{k+1}^{-1}\mathbf{C}(\mathbf{C}^H\mathbf{R}_{k+1}^{-1}\mathbf{C})^{-1}(\mathbf{f} - \mathbf{C}^H(\mathbf{w}_k + \mathbf{v}_{k+1}e_{k+1}^*)) \\ &= (\mathbf{I} - \mathbf{R}_{k+1}^{-1}\mathbf{C}(\mathbf{C}^H\mathbf{R}_{k+1}^{-1}\mathbf{C})^{-1}\mathbf{C}^H)(\mathbf{w}_k + \mathbf{v}_{k+1}e_{k+1}^*) + \mathbf{R}_{k+1}^{-1}\mathbf{C}(\mathbf{C}^H\mathbf{R}_{k+1}^{-1}\mathbf{C})^{-1}\mathbf{f} \\ &= \mathbf{P}_{k+1}(\mathbf{w}_k + \mathbf{v}_{k+1}e_{k+1}^*) + \bar{\mathbf{f}}_{k+1},\end{aligned}\quad (2.82)$$

sendo

$$\mathbf{P}_{k+1} = \mathbf{I} - \mathbf{R}_{k+1}^{-1}\mathbf{C}(\mathbf{C}^H\mathbf{R}_{k+1}^{-1}\mathbf{C})^{-1}\mathbf{C}^H = \mathbf{I} - \mathbf{\Gamma}_{k+1}\mathbf{\Psi}_{k+1}^{-1}\mathbf{C}^H\quad (2.83)$$

$$\bar{\mathbf{f}}_{k+1} = \mathbf{R}_{k+1}^{-1}\mathbf{C}(\mathbf{C}^H\mathbf{R}_{k+1}^{-1}\mathbf{C})^{-1}\mathbf{f} = \mathbf{\Gamma}_{k+1}\mathbf{\Psi}_{k+1}^{-1}\mathbf{f}\quad (2.84)$$

As equações (2.77), (2.79-2.80) e (2.82-2.84) descrevem o método conhecido como CRLS (do inglês *Constrained Recursive Least Squares*), proposto pela primeira vez em Resende et al. [62]. Analogamente ao caso da filtragem adaptativa, o método CRLS, apresentado na forma de pseudocódigo no algoritmo 2.7 provê uma melhoria significativa na convergên-

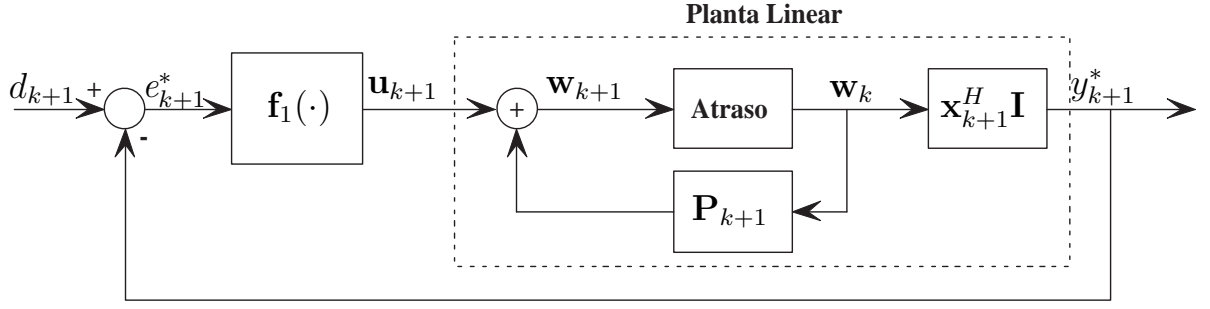


Figura 2.8: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (2.85) correspondente ao método CRLS. O controle é uma função afim do erro, $f_1(e_{k+1}) = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1}^* + \bar{\mathbf{f}}_{k+1}$ sendo \mathbf{v}_{k+1} dado por (2.23).

cia do método CLMS, porém apresenta um custo computacional e uma necessidade de armazenamento maior. Em termos de controle o método CRLS é representado pelo sistema dinâmico (2.28) e a figura 2.8

$$\left\{ \begin{array}{l} \mathbf{w}_{k+1} = \mathbf{P}_{k+1} \mathbf{w}_k + \mathbf{u}_k \\ y_{k+1} = \mathbf{x}_{k+1}^H \mathbf{w}_k \\ e_{k+1} = d_{k+1} - y_{k+1} \\ \mathbf{v}_{k+1} = \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{x}_{k+1}}{1 + \lambda_f^{-1} \mathbf{x}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{x}_{k+1}} \\ \mathbf{u}_{k+1} = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1}^* + \bar{\mathbf{f}}_{k+1} \end{array} \right. \quad (2.85)$$

O sistema é formado pela planta linear variante no tempo $\{\mathbf{P}_{k+1}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, 0\}$ em realimentação negativa com o controlador $f_1(e_k^*) = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1}^* + \bar{\mathbf{f}}_{k+1}$, correspondente a uma função afim do erro, sendo \mathbf{v}_{k+1} dado por (2.23).

2.5.3 Método com estrutura GSC

Um método alternativo foi proposto em Griffiths e Jim [61]; nesse caso o método consiste em uma estrutura denominada *generalized sidelobe canceler* (GSC) que transforma o problema de minimização com restrições lineares em um problema de minimização sem restrições, e portanto permite aplicar qualquer algoritmo adaptativo [20]. A utilização da estrutura GSC consiste em determinar uma matriz \mathbf{B} , dita matriz de bloqueio, tal que

$$\mathbf{B}^H \mathbf{C} = \mathbf{0}. \quad (2.86)$$

Algoritmo 2.7 Método CRLS para filtros adaptativos com restrições lineares

Calcular $\mathbf{R}_0^{-1} = \mu^{-1}\mathbf{I}$, $\mathbf{\Gamma}_0 = \mathbf{R}_0^{-1}\mathbf{C}$, $\mathbf{\Psi}_0^{-1} = \mu^{-1}\mathbf{I}$, $\mu =$ pequena constante positiva

Calcular $\mathbf{P}_0 = \mathbf{I} - \mathbf{\Gamma}_0\mathbf{\Psi}_0^{-1}\mathbf{C}^H$, $\bar{\mathbf{f}}_0 = \mathbf{\Gamma}_0\mathbf{\Psi}_0^{-1}\mathbf{f}$

Para $k = 0, 1, \dots, N - 1$

$$\begin{aligned}\mathbf{v}_{k+1} &= \frac{\lambda_f^{-1}\mathbf{R}_k^{-1}\mathbf{x}_{k+1}}{1 + \lambda_f^{-1}\mathbf{x}_{k+1}^H\mathbf{R}_k^{-1}\mathbf{x}_{k+1}} \\ \mathbf{R}_{k+1}^{-1} &= \lambda_f^{-1}\mathbf{R}_k^{-1} - \lambda_f^{-1}\mathbf{v}_{k+1}\mathbf{x}_{k+1}^H\mathbf{R}_k^{-1} \\ \mathbf{\Gamma}_{k+1} &= \lambda_f^{-1}\mathbf{\Gamma}_k - \lambda_f^{-1}\mathbf{v}_{k+1}\mathbf{x}_{k+1}^H\mathbf{\Gamma}_k \\ \mathbf{l}_{k+1} &= \frac{\mathbf{\Psi}_k^{-1}\mathbf{C}^H\mathbf{v}_{k+1}}{1 - \mathbf{x}_{k+1}^H\mathbf{\Gamma}_k\mathbf{\Psi}_k^{-1}\mathbf{C}^H\mathbf{v}_{k+1}} \\ \mathbf{\Psi}_{k+1}^{-1} &= \lambda_f\mathbf{\Psi}_k^{-1} + \lambda_f\mathbf{l}_{k+1}\mathbf{x}_{k+1}^H\mathbf{\Gamma}_k\mathbf{\Psi}_k^{-1} \\ e_{k+1} &= d_{k+1} - \mathbf{w}_k^H\mathbf{x}_{k+1} \\ \mathbf{P}_{k+1} &= \mathbf{I} - \mathbf{\Gamma}_{k+1}\mathbf{\Psi}_{k+1}^{-1}\mathbf{C}^H \\ \bar{\mathbf{f}}_{k+1} &= \mathbf{\Gamma}_{k+1}\mathbf{\Psi}_{k+1}^{-1}\mathbf{f} \\ \mathbf{w}_{k+1} &= \mathbf{P}_{k+1}(\mathbf{w}_k + \mathbf{v}_{k+1}e_{k+1}^*) + \bar{\mathbf{f}}_{k+1}\end{aligned}$$

Fim

Em seguida resolve-se um problema de filtragem adaptativa sem restrições através da seguinte equação:

$$\mathbf{R}'_k\mathbf{w}'_k = \mathbf{b}'_k, \quad (2.87)$$

sendo

$$\mathbf{x}'_k = \mathbf{B}^H\mathbf{x}_k, \quad (2.88)$$

e $\mathbf{b}'_k = E[\mathbf{x}'_k d_k^*]$, $\mathbf{R}'_k = E[\mathbf{x}'_k \mathbf{x}'_k{}^H]$. Finalmente a solução do problema com restrições (2.8) é então encontrada através da equação (2.89) abaixo

$$\mathbf{w}_k = \bar{\mathbf{f}} - \mathbf{B}\mathbf{w}'_k, \quad (2.89)$$

sendo $\bar{\mathbf{f}} = \mathbf{C}(\mathbf{C}^H\mathbf{C})^{-1}\mathbf{f}$. A equivalência entre a solução de (2.8) e a equação (2.89) é mostrada em vários trabalhos na literatura [60, 63–65]. Note que para resolver o problema sem restrições (2.87), pode-se utilizar os algoritmos LMS, RLS, SD-CLF e CG-CLF porém a determinação de matriz de bloqueio \mathbf{B} aumenta a complexidade do método com estrutura GSC.

2.5.4 Método CSD-CLF (*Constrained Steepest Descent*)

A relação, dada por (2.89), entre a solução dada pela estrutura GSC e a solução do problema com restrições pode ser utilizada para desenvolver novos métodos numéricos para

a resolução de (2.8) com desempenho comparável ao dos métodos CLMS e CRLS. Em Apolinário Jr et al. [20], uma versão do método CG de Chang e Willson [19] foi proposta para o caso da filtragem adaptativa com restrições lineares. Nesse caso, Apolinário Jr et al. mostraram que a matriz de projeção (2.70) no espaço nulo a esquerda da matriz das restrições pode ser reescrita como:

$$\mathbf{P} = \mathbf{B}(\mathbf{B}^H\mathbf{B})^{-1}\mathbf{B}^H, \quad (2.90)$$

sendo \mathbf{B} a matriz de bloqueio. Esse mesmo resultado foi obtido em Tseng e Griffiths [66]. Uma condição necessária e suficiente para a equivalência entre o método CLMS e o método LMS utilizado em uma estrutura GSC (LMS-GSC) é dada pela seguinte equação [20, 61]:

$$\mathbf{B}^H\mathbf{B} = \mathbf{I} \quad \implies \quad \mathbf{P} = \mathbf{B}\mathbf{B}^H. \quad (2.91)$$

Um novo método de resolução de (2.8) pode ser desenvolvido a partir da resolução de (2.87) pelo método SD-CLF e das relações (2.89) e (2.91). Para tanto, utiliza-se a janela exponencialmente decrescente de dados (2.19) para estimar a matriz de autocorrelação \mathbf{R}' do problema irrestrito (2.87) tal que

$$\begin{aligned} \mathbf{R}'_{k+1} &= \lambda_f \mathbf{R}'_k + \mathbf{x}'_{k+1} \mathbf{x}'_{k+1}{}^H \\ &= \sum_{i=1}^k \lambda_f^{k+1-i} \mathbf{B}^H \mathbf{x}_k \mathbf{x}_k{}^H \mathbf{B} + \mathbf{B}^H \mathbf{x}_{k+1} \mathbf{x}_{k+1}{}^H \mathbf{B} \\ &= \mathbf{B}^H \left(\sum_{i=1}^{k+1} \lambda_f^{k+1-i} \mathbf{x}_k \mathbf{x}_k{}^H \right) \mathbf{B} \\ &= \mathbf{B}^H \mathbf{R}_{k+1} \mathbf{B}. \end{aligned} \quad (2.92)$$

De (2.89) e do algoritmo 2.4 tem-se que

$$\begin{aligned} \mathbf{w}_{k+1} &= \bar{\mathbf{f}} - \mathbf{B}\mathbf{w}'_{k+1} \\ &= \bar{\mathbf{f}} - \mathbf{B}(\mathbf{w}'_k + \alpha_k \mathbf{g}'_k), \end{aligned} \quad (2.93)$$

sendo

$$\mathbf{g}'_{k+1} = \lambda_f \mathbf{g}'_k - \alpha_k \mathbf{R}'_{k+1} \mathbf{g}'_k + \mathbf{x}'_{k+1} e_{k+1}^*. \quad (2.94)$$

Note que, se $\mathbf{B}^H \mathbf{B} = \mathbf{I}$ então

$$\begin{aligned}
\mathbf{B} \mathbf{w}'_k = \bar{\mathbf{f}} - \mathbf{w}_k &\Leftrightarrow \mathbf{B}^H \mathbf{B} \mathbf{w}'_k = \mathbf{B}^H \bar{\mathbf{f}} - \mathbf{B}^H \mathbf{w}_k \\
&\Leftrightarrow \mathbf{w}'_k = \mathbf{B}^H \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{f} - \mathbf{B}^H \mathbf{w}_k \\
&\Leftrightarrow \mathbf{w}'_k = -\mathbf{B}^H \mathbf{w}_k.
\end{aligned} \tag{2.95}$$

Substituindo (2.95) em (2.93) resulta em

$$\begin{aligned}
\mathbf{w}_{k+1} &= \bar{\mathbf{f}} - \mathbf{B}(-\mathbf{B}^H \mathbf{w}_k + \alpha_k \mathbf{g}'_k) \\
&= \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k,
\end{aligned} \tag{2.96}$$

sendo $\mathbf{g}_k = \mathbf{B} \mathbf{g}'_k$. Então, de (2.94), (2.92) e (2.88), escreve-se

$$\begin{aligned}
\mathbf{g}_{k+1} = \mathbf{B} \mathbf{g}'_{k+1} &= \lambda_f \mathbf{B} \mathbf{g}'_k - \alpha_k \mathbf{B} \mathbf{R}'_{k+1} \mathbf{g}'_k + \mathbf{B} \mathbf{x}'_{k+1} e'_{k+1}{}^* \\
&= \lambda_f \mathbf{g}_k - \alpha_k \mathbf{B} \mathbf{B}^H \mathbf{R}_{k+1} \mathbf{B} \mathbf{B}^H \mathbf{g}_k + \mathbf{B} \mathbf{B}^H \mathbf{x}_{k+1} e'_{k+1}{}^* \\
&= \lambda_f \mathbf{g}_k - \alpha_k \mathbf{P} \mathbf{R}_{k+1} \mathbf{P}^H \mathbf{g}_k + \mathbf{P} \mathbf{x}_{k+1} e'_{k+1}{}^*.
\end{aligned} \tag{2.97}$$

Definindo

$$\bar{\mathbf{x}}_{k+1} = \mathbf{P} \mathbf{x}_{k+1} \tag{2.98}$$

$$\bar{\mathbf{R}}_{k+1} = \mathbf{P} \mathbf{R}_{k+1} \mathbf{P}^H = \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^H \tag{2.99}$$

$$\begin{aligned}
\bar{e}_{k+1} &= e'_{k+1} = d_{k+1} - \mathbf{w}_k^H \mathbf{x}'_{k+1} = d_{k+1} + \mathbf{w}_k^H \mathbf{B} \mathbf{B}^H \mathbf{x}_{k+1} \\
&= d_{k+1} + \mathbf{w}_k^H \mathbf{P} \mathbf{x}_{k+1} = d_{k+1} + \mathbf{w}_k^H \bar{\mathbf{x}}_{k+1},
\end{aligned} \tag{2.100}$$

a equação (2.97) resulta em

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k + \bar{\mathbf{x}}_{k+1} \bar{e}_{k+1}{}^*. \tag{2.101}$$

Das equações (2.96) e (2.101) escreve-se o sistema do tipo declividade máxima correspon-

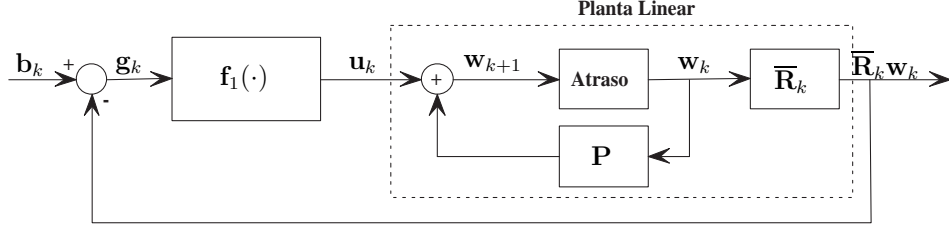


Figura 2.9: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (2.102) correspondente ao método CSD-CLF, sendo $f_1(\mathbf{g}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k$. O problema consiste em determinar o parâmetro de controle α_k tal que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\bar{\mathbf{R}}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$.

dente à resolução de (2.8) pelo método SD-CLF da seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P}\mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{e}}_{k+1}^*, \end{cases} \quad (2.102)$$

sendo $\bar{\mathbf{e}}_{k+1}$ o erro *a priori* definido por (2.100). Para o sistema (2.102), representado na forma de diagrama de blocos na figura 2.9, enuncia-se um resultado desta tese da seguinte forma [22]:

Corolário 1.1. Com a hipótese $\inf_k \lambda_{\min}(\bar{\mathbf{R}}_k) = \zeta_1 > 0$, a escolha

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}{\langle \bar{\mathbf{R}}_{k+1} \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}, \quad (2.103)$$

sendo $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{e}}_{k+1}^*$, é ótima para a função de Liapunov escolhida e garante a estabilidade assintótica do equilíbrio do sistema (2.102).

Prova do Corolário 1.1. Substituindo \mathbf{x}_{k+1} , \mathbf{R}_{k+1} e \mathbf{e}_{k+1} por $\bar{\mathbf{x}}_{k+1}$, $\bar{\mathbf{R}}_{k+1}$ e $\bar{\mathbf{e}}_{k+1}$ respectivamente, a prova segue da mesma forma que a prova do teorema 1. ■

O algoritmo 2.8 apresenta o pseudocódigo do novo método, referenciado nesta tese como CSD-CLF, baseado no sistema (2.102) e no corolário 1.1.

2.5.5 Método CCG-CLF (*Constrained Conjugate Gradient*)

Um outro novo método de resolução de (2.8) pode ser desenvolvido resolvendo (2.87) pelo método CG-CLF e utilizando as relações (2.89) e (2.91). Nesse caso, note que de (2.89)

Algoritmo 2.8 Método CSD-CLF para filtros adaptativos com restrições lineares

Calcular $\mathbf{w}_0 = \bar{\mathbf{f}} = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{f}$, $\bar{\mathbf{R}}_0 = \mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T$, $\mathbf{g}_0 = \mathbf{0}$

Para $k = 0, 1, \dots, N - 1$

$$\bar{\mathbf{x}}_{k+1} = \mathbf{P} \mathbf{x}_{k+1}$$

$$\bar{\mathbf{R}}_{k+1} = \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^H$$

$$\bar{\mathbf{e}}_{k+1} = d_{k+1} + \mathbf{w}_k^H \bar{\mathbf{x}}_{k+1}$$

$$\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{e}}_{k+1}^*$$

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}{\langle \bar{\mathbf{R}}_{k+1} \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k$$

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k$$

Fim

e do algoritmo 2.5 pode-se escrever

$$\begin{aligned} \mathbf{w}_{k+1} &= \bar{\mathbf{f}} - \mathbf{B} \mathbf{w}'_{k+1} \\ &= \bar{\mathbf{f}} - \mathbf{B}(\mathbf{w}'_k + \alpha_k \mathbf{p}'_k), \end{aligned} \quad (2.104)$$

sendo

$$\mathbf{g}'_{k+1} = \lambda_f \mathbf{g}'_k - \alpha_k \mathbf{R}'_{k+1} \mathbf{p}'_k + \mathbf{x}'_{k+1} e'_{k+1}{}^* \quad (2.105)$$

$$\mathbf{p}'_{k+1} = \mathbf{g}'_{k+1} + \beta_k \mathbf{p}'_k. \quad (2.106)$$

Substituindo (2.95) em (2.104) resulta em

$$\begin{aligned} \mathbf{w}_{k+1} &= \bar{\mathbf{f}} - \mathbf{B}(-\mathbf{B}^H \mathbf{w}_k + \alpha_k \mathbf{p}'_k) \\ &= \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k, \end{aligned} \quad (2.107)$$

sendo $\mathbf{p}_k = \mathbf{B} \mathbf{p}'_k$. Portanto, de (2.106) tem-se que

$$\mathbf{p}_{k+1} = \mathbf{B} \mathbf{p}'_{k+1} = \mathbf{B} \mathbf{g}'_{k+1} + \beta_k \mathbf{B} \mathbf{p}'_k = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k, \quad (2.108)$$

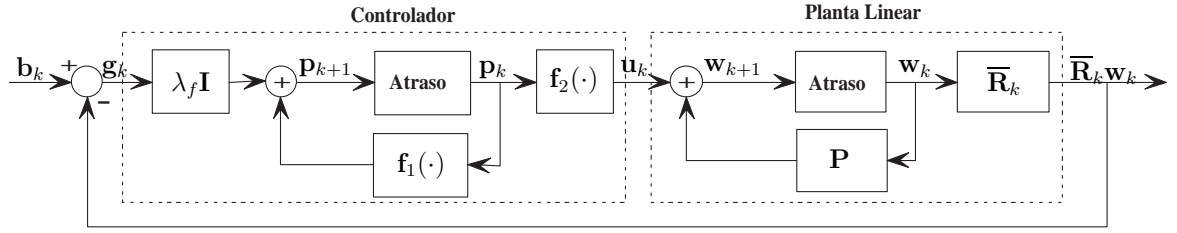


Figura 2.10: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (2.110) correspondente ao método CCG-CLF, sendo $\mathbf{f}_1(\mathbf{p}_k) = \beta_k \mathbf{p}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{e}}_{k+1}^*$ e $\mathbf{f}_2(\mathbf{p}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k$. O problema consiste em determinar os parâmetros de controle α_k e β_k tais que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\bar{\mathbf{R}}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$.

sendo $\mathbf{g}_{k+1} = \mathbf{B} \mathbf{g}'_{k+1}$. Então, de (2.105), (2.92), (2.88), (2.98-2.100) escreve-se

$$\begin{aligned}
 \mathbf{g}_{k+1} = \mathbf{B} \mathbf{g}'_{k+1} &= \lambda_f \mathbf{B} \mathbf{g}'_k - \alpha_k \mathbf{B} \mathbf{R}'_{k+1} \mathbf{p}'_k + \mathbf{B} \mathbf{x}'_{k+1} e'_{k+1}{}^* \\
 &= \lambda_f \mathbf{g}_k - \alpha_k \mathbf{B} \mathbf{B}^H \mathbf{R}_{k+1} \mathbf{B} \mathbf{B}^H \mathbf{p}_k + \mathbf{B} \mathbf{B}^H \mathbf{x}_{k+1} e_{k+1}{}^* \\
 &= \lambda_f \mathbf{g}_k - \alpha_k \mathbf{P} \mathbf{R}_{k+1} \mathbf{P}^H \mathbf{p}_k + \mathbf{P} \mathbf{x}_{k+1} e_{k+1}{}^* \\
 &= \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{e}}_{k+1}{}^*. \tag{2.109}
 \end{aligned}$$

Das equações (2.107-2.109) pode-se escrever o sistema do tipo gradiente conjugado correspondente à resolução de (2.8) pelo método CG da seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{e}}_{k+1}{}^* \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k, \end{cases} \tag{2.110}$$

sendo $\bar{\mathbf{e}}_{k+1}$ definido por (2.100). Em Apolinário Jr et al. [20], um sistema semelhante a (2.110) foi proposto; a diferença entre os dois casos consiste na atualização do resíduo \mathbf{g}_{k+1} . Apolinário Jr et al. determinaram os parâmetros α_k e β_k da mesma forma que Chang e Willson [19], introduzindo um fator empírico η e o método Polak-Ribiere. Para o sistema (2.110), representado na forma de diagrama de blocos na figura 2.10, outro resultado desta tese pode ser enunciado da seguinte forma [22]:

Algoritmo 2.9 Método CCG-CLF para filtros adaptativos com restrições lineares

Calcular $\mathbf{w}_0 = \bar{\mathbf{f}} = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{f}$, $\bar{\mathbf{R}}_0 = \mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T$, $\mathbf{g}_0 = \mathbf{p}_0 = \mathbf{0}$

Para $k = 0, 1, \dots, N - 1$

$$\begin{aligned}\bar{\mathbf{x}}_{k+1} &= \mathbf{P} \mathbf{x}_{k+1} \\ \bar{\mathbf{R}}_{k+1} &= \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^H \\ \bar{e}_{k+1} &= d_{k+1} + \mathbf{w}_k^H \bar{\mathbf{x}}_{k+1} \\ \tilde{\mathbf{g}}_k &= \lambda_f \mathbf{g}_k + \bar{\mathbf{x}}_{k+1} \bar{e}_{k+1}^* \\ \alpha_k &= \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \\ \mathbf{w}_{k+1} &= \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} &= \tilde{\mathbf{g}}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \\ \beta_k &= -\frac{\langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \\ \mathbf{p}_{k+1} &= \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k\end{aligned}$$

Fim

Corolário 2.1. Com as hipóteses $\inf_k \lambda_{\min}(\bar{\mathbf{R}}_k^{-1}) = \zeta_1 > 0$ e $\inf_k \lambda_{\min}(\bar{\mathbf{R}}_k) = \zeta_2 > 0$, as escolhas

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \quad (2.111)$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}, \quad (2.112)$$

sendo $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \bar{\mathbf{x}}_{k+1} \bar{e}_{k+1}^*$, são ótimas para as funções de Liapunov escolhidas e garantem a estabilidade assintótica do equilíbrio do sistema (2.110).

Prova do Corolário 2.1. Substituindo \mathbf{x}_{k+1} , \mathbf{R}_{k+1} e e_{k+1} por $\bar{\mathbf{x}}_{k+1}$, $\bar{\mathbf{R}}_{k+1}$ e \bar{e}_{k+1} respectivamente, a prova segue da mesma forma que a prova do teorema 2. ■

O algoritmo 2.9, referenciado nesta tese como CCG-CLF, apresenta o pseudocódigo da extensão do método CG, baseado no corolário 2.1, para o caso da filtragem adaptativa com restrições lineares.

2.6 Simulações Numéricas

Para verificar as propriedades de convergência dos novos métodos propostos nesta tese, simulam-se três cenários de filtragem adaptativa (equalização adaptativa, identificação de sistema e predição linear) e um cenário de filtragem adaptativa com restrições lineares (sistema de comunicação DS-CDMA).

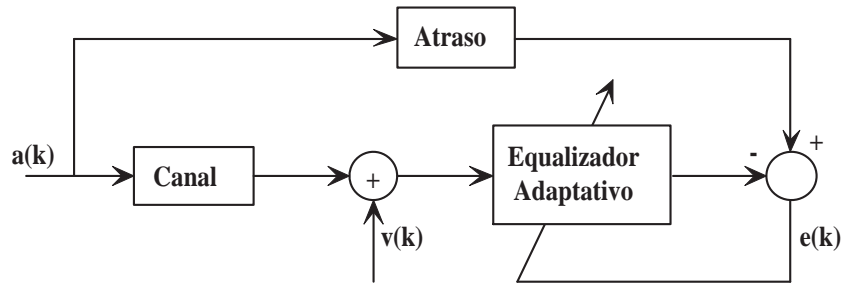


Figura 2.11: Diagrama de blocos do equalizador adaptativo.

2.6.1 Equalização Adaptativa

Para ilustrar as propriedades de convergência dos novos métodos de filtragem adaptativa propostos nesta tese (algoritmos 2.4 e 2.5), considere o equalizador adaptativo da figura 2.11. A entrada do sistema é uma seqüência aleatória $\{a(k)\}$ com valores ± 1 , portanto de média nula. A seqüência é distorcida por um canal de resposta em freqüência dada por

$$\begin{aligned} h(k) &= \frac{1}{2}[1 + \cos(2\pi(k-2)/W)], & k = 1, 2, 3 \\ &= 0, & k > 3 \end{aligned} \quad (2.113)$$

sendo que W controla a amplitude da distorção no canal. Um aumento em W causa um aumento da distorção do canal e resulta no aumento do espalhamento κ dos autovalores da matriz de autocorrelação \mathbf{R} do vetor de entrada do equalizador. O canal é corrompido por um vetor de ruído branco aditivo de média nula e variância σ^2 . O equalizador é composto por 11 *taps*. Como a resposta ao impulso do canal é simétrica em relação a $k = 2$, os pesos dos *taps* do equalizador são simétricos em relação ao sexto *tap*. Desta forma, a entrada $\{a(k)\}$ do canal é atrasada por 7 amostras para gerar a resposta desejada do equalizador. A experiência é feita com dois valores de W , 2.9 e 3.5, que resultam em espalhamentos κ de \mathbf{R} iguais a 6.07 e 46.821 respectivamente. Neste caso, os parâmetros são escolhidos tais que $\lambda_f = \eta = 0.99$, $\sigma^2 = 0.001$ e o *ensemble* é constituído por 200 amostras independentes. As figuras 2.12 e 2.13 apresentam uma comparação das *curvas de aprendizado*, gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k , do equalizador da figura 2.11 utilizando os métodos LMS, RLS, CG-CLF, SD-CLF. O desempenho dos algoritmos propostos é também comparado com outros dois métodos CG propostos em Boray e Srinath

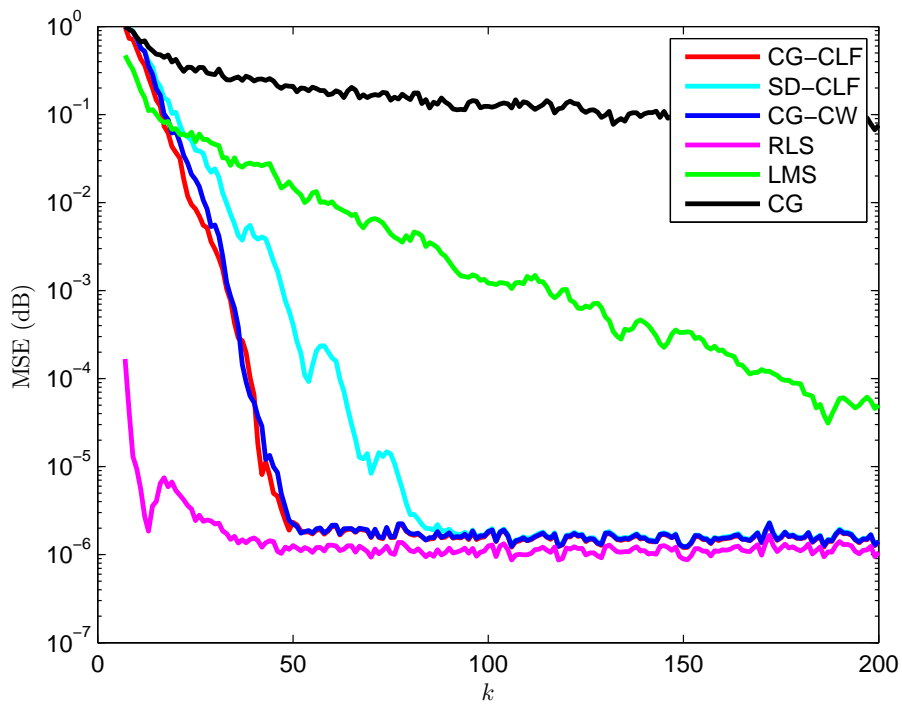


Figura 2.12: Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos equalizadores adaptativos com $W = 2.9$. Os valores de regime estão apresentados na tabela 2.1.

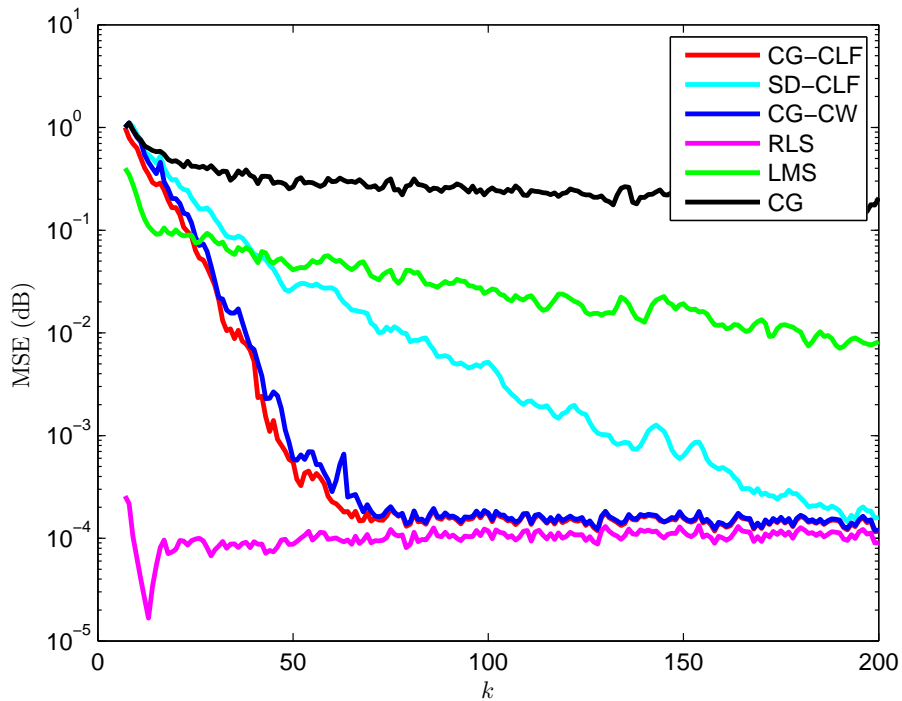


Figura 2.13: Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos equalizadores adaptativos com $W = 3.5$. Os valores de regime estão apresentados na tabela 2.1.

Método	$W = 2.9$	$W = 3.5$
LMS	1.23×10^{-3}	2.32×10^{-2}
SD-CLF	1.79×10^{-6}	2.44×10^{-3}
RLS	1.16×10^{-6}	1.03×10^{-4}
CG-CLF	1.65×10^{-6}	1.57×10^{-4}
CG-CW	1.68×10^{-6}	1.64×10^{-4}
CG	1.24×10^{-1}	2.42×10^{-1}

Tabela 2.1: Erro médio quadrático dos filtros adaptativos na configuração de equalização adaptativa. Os métodos propostos neste trabalho apresentam um desempenho comparável aos métodos LMS, RLS e CG-CW.

[18] e Chang e Willson [19] referenciados como CG e CG-CW, respectivamente. A tabela 2.1 apresenta os valores de regime do erro médio quadrático.

Nota-se que o método CG-CLF proposto nesta tese apresenta um desempenho comparável aos métodos RLS e CG-CW de [19]. No caso do método CG padrão utilizado em Boray e Srinath [18], como é realizado apenas uma iteração para cada par $(\mathbf{R}_k, \mathbf{b}_k)$, seria necessário fazer $\mathbf{p}_k = \mathbf{g}_k$ em cada iteração (isto corresponde a reiniciar o método CG tal que $\mathbf{p}_0 = \mathbf{g}_0$ a cada iteração k) para garantir a ortogonalidade dos resíduos gerados. Portanto, notando que $\mathbf{p}_k = \mathbf{g}_k + \beta_{k-1}\mathbf{p}_{k-1}$, pode-se dizer que ocorre uma perda de ortogonalidade dos resíduos ($\mathbf{g}_{k+1}^H \mathbf{g}_k \neq 0$) do método CG padrão [3], causando um desempenho pior do método. O método SD-CLF converge para o mesmo valor de erro médio quadrático em regime que os métodos RLS, CG-CW e CG-CLF, porém a uma taxa de convergência menor. A taxa de convergência do método SD-CLF é maior que a do LMS e é inversamente proporcional ao espalhamento κ dos autovalores da matriz de autocorrelação (quanto maior κ , menor será a taxa de convergência). Vale ressaltar que, mesmo que tenha uma taxa de convergência menor que o método RLS em um ambiente variante no tempo, o método SD-CLF pode ser utilizado em casos onde o método LMS (que possui uma taxa de convergência comparável, mas com erro médio quadrático de regime maior e custo computacional maior) é aplicável.

2.6.2 Identificação de Sistema

Na configuração de identificação de sistema consideram-se duas plantas desconhecidas que são filtros de resposta ao impulso finita (FIR) de ordem 10 e 20; o sinal de entrada é um ruído branco Gaussiano de variância $\sigma^2 = 1$. A figura 2.14 mostra o diagrama de blocos da configuração de identificação de sistema. As curvas de aprendizado do filtro adaptativo na

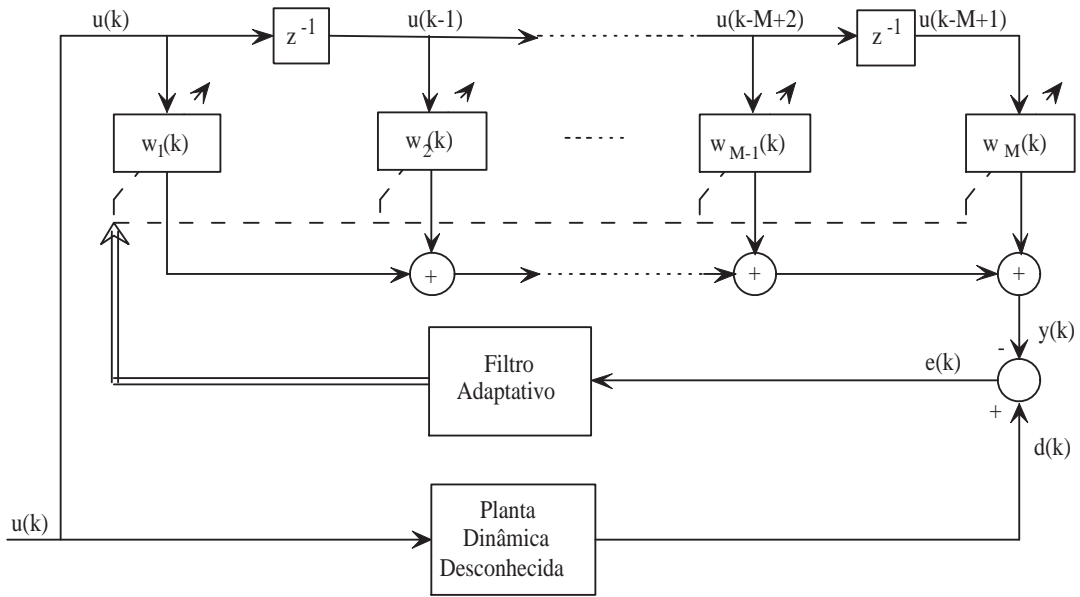


Figura 2.14: Identificação de uma planta dinâmica desconhecida por filtro adaptativo com *tap* atrasador. Os dados de entrada u e os pesos do *tap* w são reais [15].

Filtro Adaptativo	FIR 10	FIR 20
LMS	4.35×10^{-2}	4.74×10^{-2}
SD-CLF	1.36×10^{-2}	1.36×10^{-2}
RLS	8.06×10^{-3}	8.11×10^{-3}
CG-CLF	1.36×10^{-2}	1.36×10^{-2}
CG-CW	1.37×10^{-2}	1.36×10^{-2}
CG	4.66×10^{-1}	4.81×10^{-1}

Tabela 2.2: Erro médio quadrático dos filtros adaptativos na configuração de identificação de sistema. Os novos métodos propostos nesta tese apresentam um desempenho comparável aos métodos RLS e CG-CW.

configuração de identificador de sistema utilizando os métodos LMS, RLS, SD-CLF, CG-CLF, CG-CW e CG estão apresentadas nas figuras 2.15 e 2.16. Neste caso o *ensemble* é constituído por 200 amostras independentes. A tabela 2.2 apresenta os valores de regime para quais converge o erro médio quadrático dos filtros adaptativos. Como no caso do equalizador adaptativo, nota-se que os novos métodos propostos nesta tese apresentam um desempenho comparável ao método RLS. Por outro lado a perda de convergência do método CG padrão de [18], devido à perda de ortogonalidade dos resíduos, é mais acentuada na configuração de identificação de sistema.

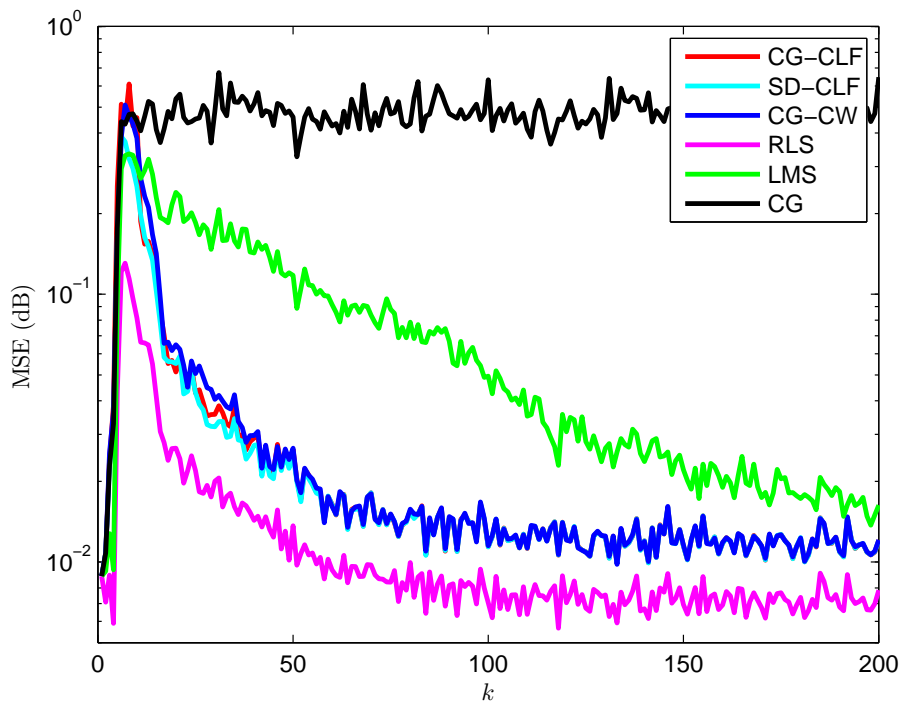


Figura 2.15: Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos na identificação de um FIR de ordem 10. Os valores de regime estão apresentados na tabela 2.2.

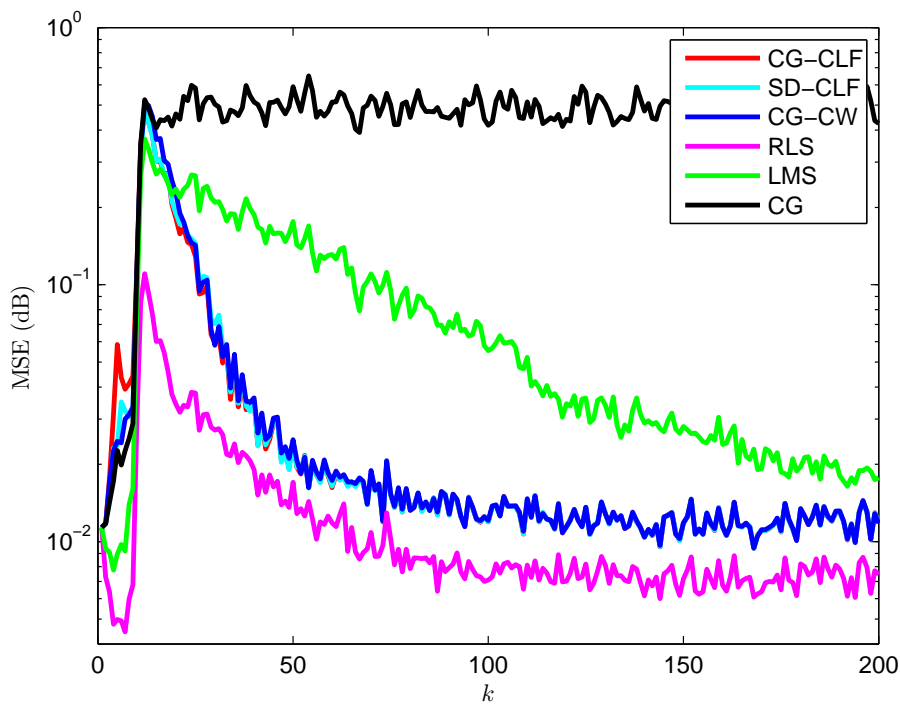


Figura 2.16: Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos na identificação de um FIR de ordem 20. Os valores de regime estão apresentados na tabela 2.2.

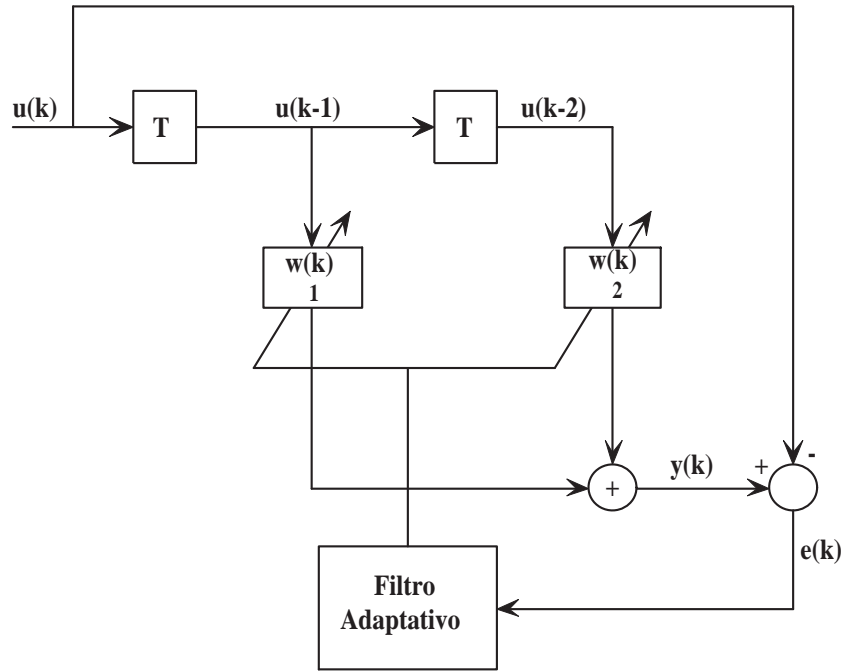


Figura 2.17: Diagrama de blocos de um filtro adaptativo na configuração de predição linear.

2.6.3 Predição Linear

A terceira configuração de filtro adaptativo simulada neste trabalho é o cenário de predição linear. A figura 2.17 mostra a configuração do filtro adaptativo de 2 *taps* no cenário de predição linear. As entradas dos *taps* $u(k-1)$ e $u(k-2)$ são extraídas do processo autoregressivo (AR) de valores reais dada pela seguinte equação

$$u(k) + a_1u(k-1) + a_2u(k-2) = v(k) \quad (2.114)$$

sendo $v(k)$ um ruído branco de média nula e variância σ_v^2 . Os valores de a_1 e a_2 são determinados pelo espalhamento dos autovalores da matriz de autocorrelação dos dados de entrada. Para uma tentativa do experimento, são obtidas 500 amostras do ruído branco $\{v(i)\}$ a partir de um gerador de números aleatórios de média nula e variância ajustável. O *ensemble* é constituído por 200 tentativas independentes e o experimento é feito para valores de espalhamento $\kappa = 3$ e $\kappa = 10$. As figuras 2.18 e 2.19 apresentam as curvas de aprendizagem dos filtros adaptativos no cenário de predição linear. A tabela 2.3 apresenta os valores de regime para quais converge o erro quadrático dos preditores lineares. Neste caso também, nota-se que os métodos propostos nesta tese apresentam propriedades de convergência comparáveis

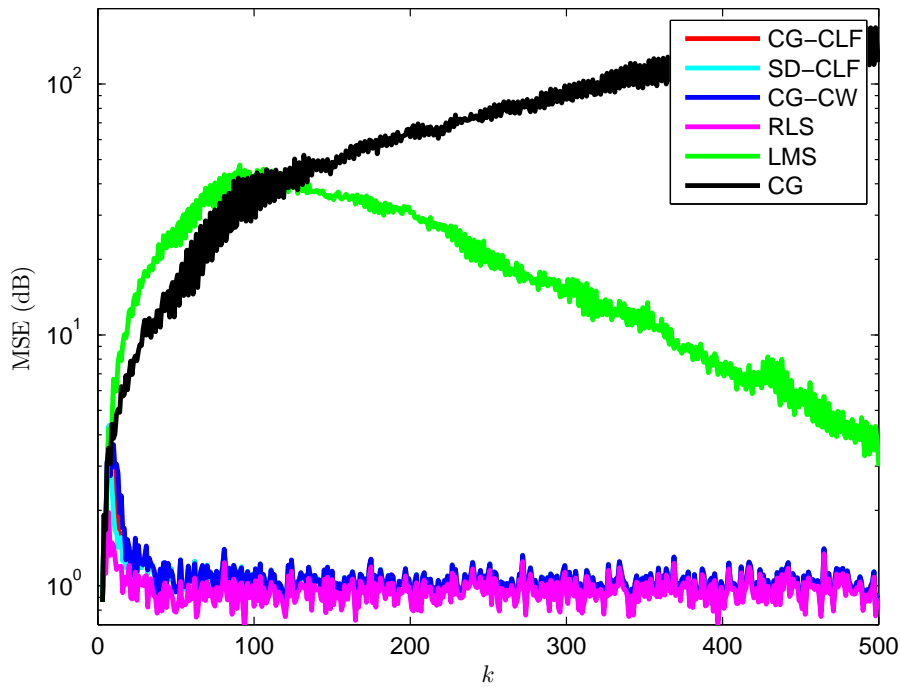


Figura 2.18: Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos na configuração de predição linear com $\kappa = 3$. Os valores de regime estão apresentados na tabela 2.3.

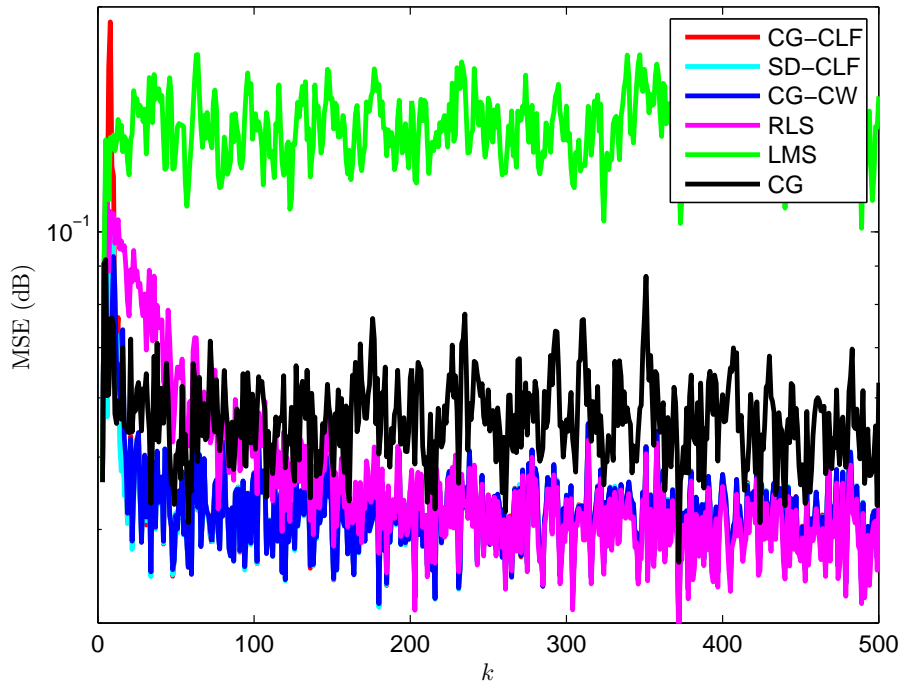


Figura 2.19: Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos na configuração de predição linear com $\kappa = 10$. Os valores de regime estão apresentados na tabela 2.3.

Preditor linear	$\kappa = 10$	$\kappa = 3$
LMS	1.37×10^{-1}	16.72
SD-CLF	4.18×10^{-2}	1.03
RLS	4.29×10^{-2}	0.96
CG-CLF	4.18×10^{-2}	1.03
CG-CW	4.18×10^{-2}	1.03
CG	5.58×10^{-2}	76.05

Tabela 2.3: Erro médio quadrático dos filtros adaptativos na configuração de predição linear. Os métodos propostos neste trabalho apresentam um desempenho comparável ao método de [19] e ao método RLS.

κ	J_{\min}	LMS	SD-CLF	RLS	CG-CLF	CG-CW	CG
3	0.0731	227.78	13.11	12.12	13.09	13.11	1039.37
10	0.0322	3.25	0.30	0.33	0.30	0.30	0.74

Tabela 2.4: Porcentagem de Desajuste $M(\%)$

aos métodos CG-CW e RLS. Vale ressaltar que a convergência lenta do método SD-CLF, que depende do espalhamento dos autovalores, observado no caso do equalizador adaptativo, não ocorre. Para o método CG padrão, a perda de ortogonalidade resulta em uma perda de convergência para pequenos valores de espalhamento. A tabela 2.4 compara a porcentagem de desajuste definida como

$$M = \frac{J_{\infty} - J_{\min}}{J_{\min}}, \quad (2.115)$$

onde J_{∞} é o valor esperado do erro médio quadrático em regime e J_{\min} é o erro médio quadrático mínimo [18].

2.6.4 Sistema de comunicação móvel DS-CDMA

Para testar os novos métodos CSD-CLF e CCG-CLF propostos, comparam-se esses com os métodos CLMS, CRLS e o método CG proposto em Apolinário Jr et al. [20] referenciado nesta tese como CCG-CW, aplicado ao caso da detecção mono-usuário em um sistema de comunicação móvel DS-CDMA. Nessa experiência, considera-se um modelo simplificado de transmissão síncrona *downlink* de K usuários, conforme mostrado na figura 2.20 [67]. O sinal contínuo recebido é transmitido através de um filtro digital e é amostrado a uma taxa tal que o vetor do sinal de entrada digital possa ser expressado como $\mathbf{x}_k = \mathbf{S}\mathbf{a}_k + \mathbf{n}_k$, sendo $\mathbf{S} = [s_1 \ s_2 \ \dots \ s_K]$ a matriz de transmissão contendo as seqüências transmitidas

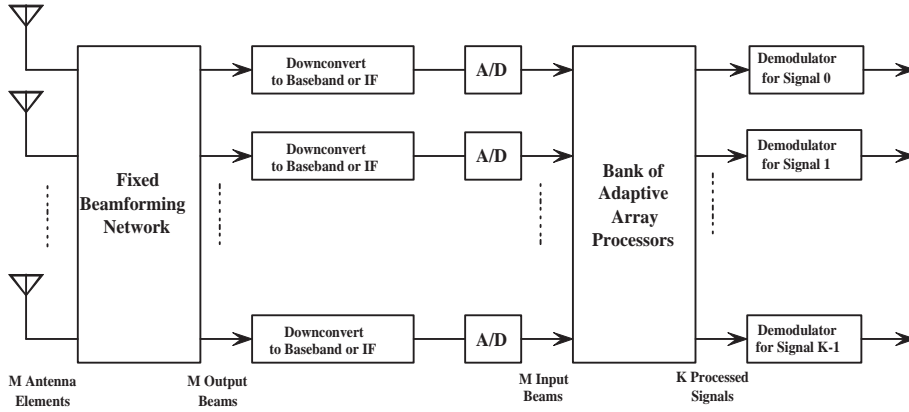


Figura 2.20: Modelo simplificado de transmissão síncrona *downlink* de K usuários.

amostradas dos usuários 1 a K , $\mathbf{A} = \text{diag}[A_1 \ A_2 \ \dots \ A_K]$ contém as amplitudes dos sinais de cada usuário, $\mathbf{c}_k = [c_{1k} \ c_{2k} \ \dots \ c_{Kk}]^T$ contém os bits da informação, e \mathbf{n}_k a sequência de ruído amostrado. Para esse exemplo, as restrições são tais que $\mathbf{C} = \mathbf{s}_1$, and $\mathbf{f} = 1$. O número de usuários é $K = 5$ e utiliza-se um *Gold Code* de comprimento 7 para a transmissão [20]. O quociente sinal/ruído (SNR) do usuário 1 é igual a 8 dB e a potência relativa dos usuários que interferem é igual a 20 dB, isto é, $10 \log(P_i/P_1) = 20$. Os mesmos parâmetros são escolhidos da mesma forma que em Apolinário Jr et al. [20]. As figuras 2.21 e 2.22 apresentam as curvas de aprendizado dos filtros adaptativos com restrições lineares aplicados à detecção mono-usuário em um sistema de comunicação móvel DS-CDMA. Pode ser observado que os novos métodos CLF propostos apresentam propriedades de convergência comparáveis com os métodos CRLS e CCG-CW.

2.7 Contribuições e Publicações

Neste capítulo, através da representação dos métodos iterativos como sistemas dinâmicos, estenderam-se os métodos SD e CG para os casos de sistemas lineares variantes no tempo com ou sem restrições lineares. Os problemas de filtragem adaptativa com ou sem restrições lineares foram revisados e colocados sob a perspectiva de sistemas dinâmicos. Essa abordagem permitiu o desenvolvimento de duas novas famílias de métodos iterativos para cada um dos dois problemas citados acima. A convergência e a otimalidade dos parâmetros dos novos métodos propostos foram provados através de escolhas apropriadas de funções de Liapunov com controle nos teoremas 1 e 2, e seus respectivos corolários 1.1 e 2.1. Ao contrário

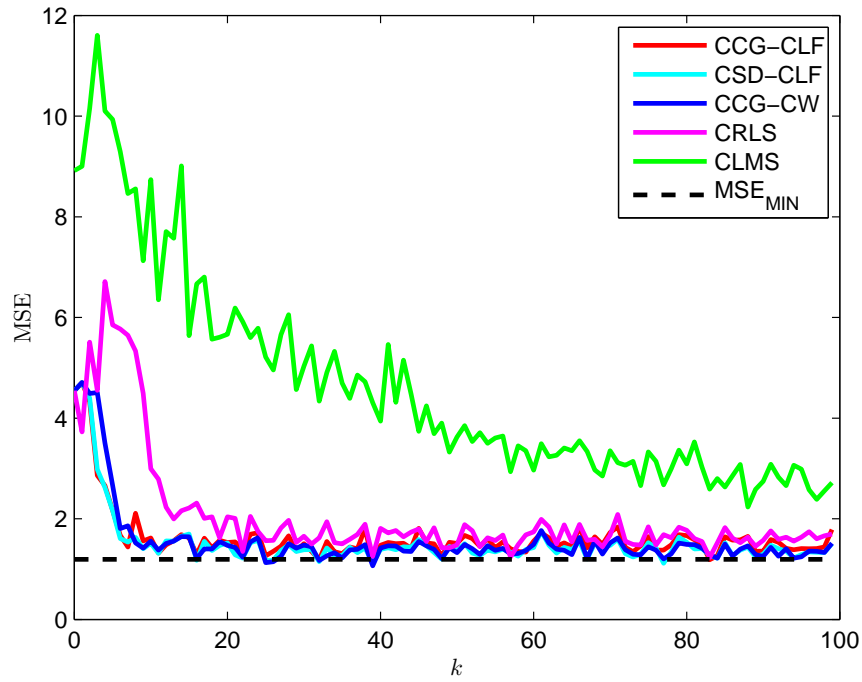


Figura 2.21: Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 3 usuários.

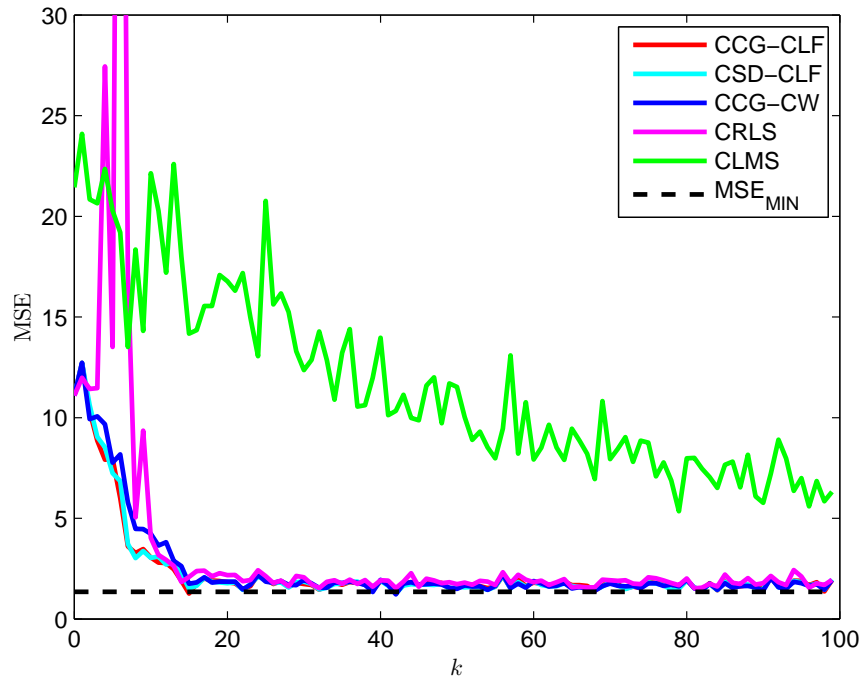


Figura 2.22: Curvas de aprendizado (gráfico do erro médio quadrático (MSE) $e^2(k) = (d_k - y_k)^2$ em função da iteração k) dos filtros adaptativos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 5 usuários.

Método	FLOP
LMS	$4M + 1$
RLS	$6M^2 + 7M + 2$
SD-CLF	$5M^2 + 12M - 1$
CG-CLF	$5M^2 + 16M$
CLMS	$2M^2 + 4M$
CRLS	$4M^3 + 26M^2 - 4M + 1$
CSD-CLF	$9M^2 + 11M - 1$
CCG-CLF	$9M^2 + 15M$

Tabela 2.5: Comparação do custo computacional dos métodos de filtragem adaptativa com ou sem restrições lineares.

de trabalhos anteriores [18–20], os novos métodos propostos não utilizam nenhum parâmetro empírico. O desempenho dos métodos propostos é superior ao do método LMS em ambos os casos, a um custo computacional maior, e comparável ou superior ao do método RLS a um custo computacional menor, conforme mostraram as simulações numéricas e a tabela 2.5 na qual compara-se o número de operações de ponto flutuante (FLOP) dos métodos.

As publicações decorrentes dos resultados e contribuições deste capítulo são enumerados a seguir:

1. DIENE, O., BHAYA, A., “Adaptive filtering algorithms designed using control Liapunov functions”. *IEEE Signal Processing Letters* v. 4, pp. 224-227, 2006. Correction in vol. 14 (12) (2007) 1047.
2. DIENE, O., BHAYA, A., “Métodos iterativos lineares projetados através de ferramentas da teoria de controle e suas aplicações”. *Revista de Controle e Automação* v. 17, n. 3, pp. 265-277, 2006. URL <http://www.scielo.br/pdf/ca/v17n3/a03v17n3.pdf>.
3. DIENE, O., BHAYA, A., “Linearly constrained adaptive filtering algorithms designed using control Liapunov functions”. In: *Proc. 16th Congresso Brasileiro de Automação*, Salvador, BA, Brazil, October 2006.

Capítulo 3

Análise e projeto de algoritmos de filtragem adaptativa cega

3.1 Introdução

No capítulo anterior viu-se que, em áreas como a telecomunicação, é necessário um pré/pós processamento, utilizando informações dos sinais desejados, de sinais transmitidos/recebidos corrompidos por vários fatores. No entanto, em muitos casos os sinais originais desejados não estão disponíveis. Nesses casos os pré/pós processamento podem ser feitos através de filtros adaptativos cegos e de filtros adaptativos cegos com restrições lineares. Neste capítulo os principais métodos de resolução desses dois problemas serão revisados e colocados em uma perspectiva de sistemas dinâmicos realimentados. Com base em teoremas, que provam a convergência através de funções de Liapunov com controle, novos métodos de resolução dos dois problemas de filtragem adaptativa cega serão apresentados e analisados. Também mostrar-se-á, simulando numericamente um sistema de antenas adaptativas e outro sistema de comunicação DS-CDMA, que os novos métodos propostos apresentam um desempenho superior ou igual ao dos métodos mais comumente utilizados.

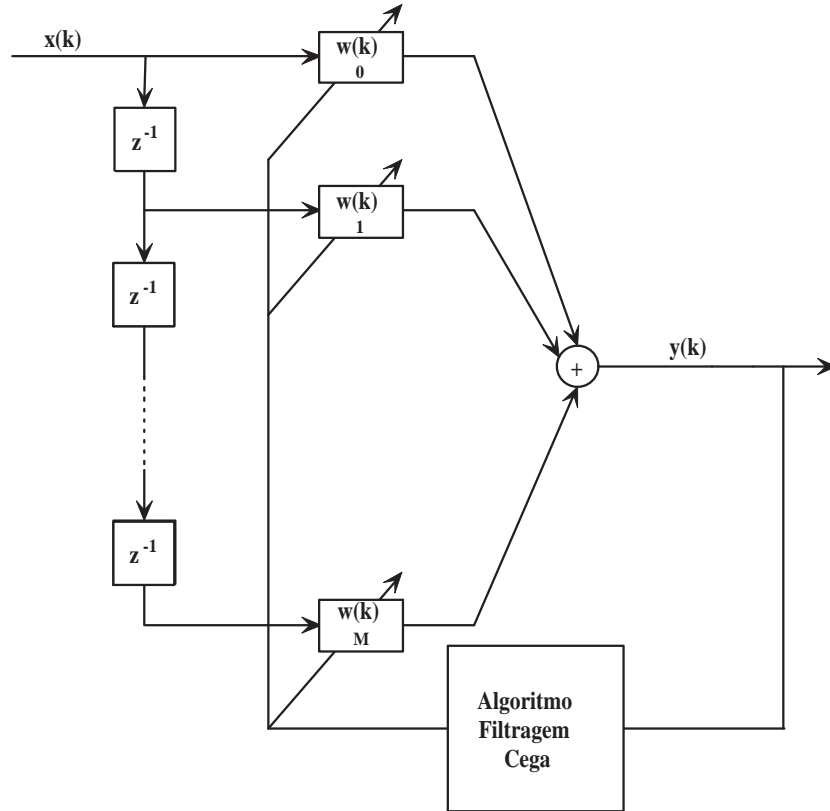


Figura 3.1: Filtro transversal adaptativo cego de resposta finita ao impulso (FIR).

3.2 Formulação do Problema

3.2.1 Filtragem adaptativa cega

O problema de filtragem adaptativa cega consiste em recuperar um sinal variante no tempo, a partir de uma versão distorcida desse mesmo, corrompido por um ruído (também variante no tempo) e misturado com outras interferências, sendo que o sinal desejado d_k original e a função que o mistura com as interferências são desconhecidos. Neste caso o processamento será feito sem supervisão uma vez que não se tem disponível o sinal desejado para fazer a comparação com a saída do filtro. A figura 3.10 mostra uma das realizações mais básicas de um filtro adaptativo cego. Nesse caso, o vetor \mathbf{w}_k de pesos do filtro é atualizado iterativamente a partir de uma função da saída y_k , uma vez que o sinal desejado não está disponível. Note que os vetores de entradas \mathbf{x}_k e de pesos \mathbf{w}_k , e a saída y_k são dadas pela equação 2.1 ($y_k = \mathbf{w}_k^H \mathbf{x}_k$), como no caso do filtro adaptativo. Em áreas como telecomunicações, por exemplo, uma parte importante dos sinais considerados apresentam um módulo

constante ou com variação lenta. Essa propriedade é utilizada para definir uma função objetivo bastante aplicada na literatura [29, 68]. Essa função custo denominada como critério do módulo constante (CM) é definida pela seguinte equação [30]:

$$J(p, 2) = E[(|y_k|^p - \rho)^2], \quad (3.1)$$

sendo

$$\rho = \frac{E[|d_k|^{2p}]}{E[|d_k|^p]}, \quad (3.2)$$

e p um parâmetro inteiro positivo. Dessa forma define-se um problema de filtragem adaptativa cega da seguinte forma:

$$\min_{\mathbf{w}_k} J(p, 2) = \min_{\mathbf{w}_k} E[(|y_k|^p - \rho)^2] = \min_{\mathbf{w}_k} E[(|\mathbf{w}_k^H \mathbf{x}_k|^p - \rho)^2]. \quad (3.3)$$

Os resultados em Mai e Sayed [69], Chen et al. [70] e Whitehead e Takawira [71] sugerem que o critério do módulo constante, da função custo (3.1), apresenta um desempenho melhor quando $p = 2$. Portanto nesta tese, somente o caso $p = 2$ será considerado, porém os resultados podem ser estendidos para outros valores de p . Nesse caso o problema de filtragem adaptativa cega (3.3) é reescrita como

$$\min_{\mathbf{w}} E[(\mathbf{w}^H \mathbf{x}_k \mathbf{x}_k^H \mathbf{w} - \rho)^2]. \quad (3.4)$$

3.2.2 Filtragem adaptativa cega com restrições lineares

Analogamente ao caso do problema de filtragem adaptativa, em algumas aplicações de filtragem adaptativa cega o vetor de pesos \mathbf{w}_k deve satisfazer restrições lineares, além de ser solução de (3.4). Nesse caso redefine-se o problema de filtragem adaptativa cega como um problema de filtragem adaptativa cega com restrições lineares da seguinte forma:

$$\begin{aligned} \min_{\mathbf{w}_k} E[(|\mathbf{w}_k^H \mathbf{x}_k|^p - \rho)^2], \\ \text{sujeito a } \mathbf{C}^H \mathbf{w}_k = \mathbf{f} \end{aligned} \quad (3.5)$$

sendo $\mathbf{C} \in \mathbb{C}^{M+1 \times p}$ a matriz das restrições, e $\mathbf{f} \in \mathbb{C}^p$ o vetor de ganho.

3.3 Algoritmos de filtragem adaptativa cega

3.3.1 Método SGD-CMA (*Stochastic Gradient Constant Modulus*)

Os sistemas do tipo gradiente de descida podem ser utilizados para resolver de forma iterativa o problema de filtragem adaptativa cega (3.4) [1, 4, 5]. Nesse caso, o mínimo de (3.1) é encontrado a partir de um ponto inicial arbitrário e uma seqüência na direção de descida do gradiente de (3.1) tal que:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mu' \nabla J(2, 2) = \mathbf{w}_k - \mu' \nabla E[(|\mathbf{w}_k^H \mathbf{x}_k|^2 - \rho)^2], \quad (3.6)$$

sendo o passo μ' um pequeno valor real positivo. O valor esperado da função custo (3.1) pode ser estimado a partir dos valores instantâneos estocásticos tal que

$$J(2, 2) = (|y_k|^2 - \rho)^2 = (|\mathbf{w}_k^H \mathbf{x}_k|^2 - \rho)^2 = (\mathbf{w}_k^H \mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k - \rho)^2. \quad (3.7)$$

Então o gradiente de $J(2, 2)$ é determinado pela seguinte equação

$$\nabla J(2, 2) = \frac{\partial J(2, 2)}{\partial \mathbf{w}_k} = 4\mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k (\mathbf{w}_k^H \mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k - \rho) = 4\mathbf{x}_k y_k^* (|y_k|^2 - \rho). \quad (3.8)$$

Portanto, a equação (3.6) de atualização do vetor de pesos \mathbf{w}_k é reescrita da seguinte forma

$$\mathbf{w}_{k+1} = \mathbf{w}_k - 4\mu' \mathbf{x}_k y_k^* (|y_k|^2 - \rho) = \mathbf{w}_k - \mu \mathbf{x}_k y_k^* (|y_k|^2 - \rho) = \mathbf{w}_k + \mu \mathbf{x}_k y_k^* (\rho - |y_k|^2), \quad (3.9)$$

sendo μ um pequeno passo de valor real positivo. A equação (3.9) descreve o método conhecido na literatura como SGD-CMA (do inglês *Stochastic Gradient Constant Modulus Algorithm*) [30]. Do ponto de vista de controle o método SGD-CMA pode ser representado por um problema de rastreamento de sinal descrito pela figura 3.2 e pelo seguinte sistema de equações:

$$\left\{ \begin{array}{l} \mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \\ y_k = \mathbf{w}_k^H \mathbf{x}_k \\ e_k = \rho - |y_k|^2 \\ \mathbf{v}_k = \mathbf{x}_k y_k^* \\ \mathbf{u}_k = \mu \mathbf{v}_k e_k \end{array} \right. \quad (3.10)$$

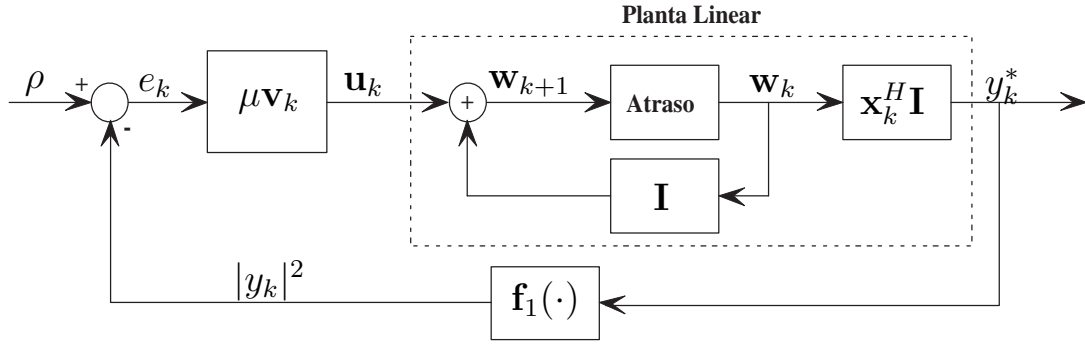


Figura 3.2: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.4) pelo sistema dinâmico (3.10) correspondente ao método SGD-CMA, sendo $\mathbf{v}_k = \mathbf{x}_k y_k^*$ e $f_1(y_k^*) = |y_k|^2$. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $|y_k|^2 \rightarrow \rho$, quando $k \rightarrow \infty$.

Algoritmo 3.1 Método SGD-CMA para filtros adaptativos cegos [30]

Para $k = 0, 1, \dots, N - 1$

$$y_k = \mathbf{w}_k^H \mathbf{x}_k$$

$$e_k = \rho - |y_k|^2$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu \mathbf{x}_k y_k^* e_k$$

Fim

O sistema dinâmico (3.10) é constituído pela planta linear variante no tempo $\{\mathbf{I}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, \mathbf{0}\}$ e o controlador proporcional ao erro $\{\mathbf{0}, \mathbf{0}, \mathbf{0}, \mu \mathbf{x}_k y_k^*\}$ em realimentação negativa. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $|y_k|^2 \rightarrow \rho$, quando $k \rightarrow \infty$. O algoritmo 3.1 apresenta o método SGD-CMA em forma de pseudocódigo.

O método SGD-CMA apresenta um baixo custo computacional, porém, como no caso do método LMS (algoritmo 2.2), a sua convergência depende da escolha do passo μ . Como o sistema (3.10) é não linear e variante no tempo, a escolha de um valor fixo μ que garanta a estabilidade assintótica de (3.10) torna-se difícil. A escolha de um passo grande leva à instabilidade do equilíbrio do sistema (3.10) que corresponde à divergência do algoritmo 3.1, enquanto a escolha de um passo pequeno resulta em uma convergência lenta. Essa convergência lenta do método SGD-CMA é também observada no caso de sinais em ambientes altamente corrompidos [72, 73].

3.3.2 Método RLS-CMA (*Recursive Least Squares Constant Modulus*)

Para os casos onde a convergência do método SGD-CMA é lenta, pode-se desenvolver outro método de filtragem adaptativa cega através de uma estimativa do valor esperado da

função custo (3.1) que leva em consideração informações dos sinais em instantes anteriores. Isso já foi feito em Pickholtz e Elbarbary [74] para o caso em que os parâmetros da função custo (3.1) são tais que $p = 2$ e $q = 1$. Nesta seção um método do tipo RLS será desenvolvido para a resolução do problema (3.4). Isso é feito através da utilização da janela de dados ponderada de forma exponencialmente decrescente do método RLS para a filtragem adaptativa, nesse caso a função custo (3.1) é redefinida tal que

$$J(2, 2) = \sum_{i=1}^k \lambda_f^{k-i} (\mathbf{w}_k^H \mathbf{x}_i \mathbf{x}_i^H \mathbf{w}_k - \rho)^2, \quad (3.11)$$

sendo λ_f o fator de esquecimento. Note que em ambientes com sinais estacionários ou com variação lenta, a diferença entre $\mathbf{x}_i^H \mathbf{w}_k$ e $\mathbf{x}_i^H \mathbf{w}_{i-1}$ é pequena quando i tende para k , enquanto que as diferenças maiores para i distante de k são atenuadas de forma exponencial pelo fator λ_f^{k-i} . Portanto, a função custo (3.11) pode ser aproximada da seguinte forma [70]

$$J'(2, 2) = \sum_{i=1}^k \lambda_f^{k-i} (\mathbf{w}_k^H \mathbf{z}_i - \rho)^2, \quad (3.12)$$

sendo

$$\mathbf{z}_i = \mathbf{x}_i \mathbf{x}_i^H \mathbf{w}_{i-1}. \quad (3.13)$$

Para um processamento amostra por amostra, definindo

$$\mathbf{R}_k = \lambda \mathbf{R}_{k-1} + \mathbf{z}_k \mathbf{z}_k^H \quad (3.14)$$

$$\mathbf{b}_k = \lambda \mathbf{b}_{k-1} + \rho \mathbf{z}_k, \quad (3.15)$$

a equação (3.12) é reescrita como

$$J'(2, 2) = \mathbf{w}_k^H \mathbf{R}_k \mathbf{w}_k - \mathbf{w}_k^H \mathbf{b}_k - \mathbf{b}_k^H \mathbf{w}_k + \sum_{i=1}^k \lambda^{k-i} \rho^2 \quad (3.16)$$

Note que \mathbf{R}_k e \mathbf{b}_k dependem da variável \mathbf{z}_k que, por sua vez, depende de \mathbf{w}_{k-1} . Entretanto, para um processamento em tempo real amostra por amostra, os outros vetores de pesos \mathbf{w}_i ($i = 1, \dots, k-1$) são calculados nas iterações anteriores e podem ser considerados constantes na iteração k [70]. Para ser equivalente ao critério do módulo constante (3.4), a função

(3.16) deve ser minimizada a cada iteração k em relação a \mathbf{w}_k (a função custo varia com k e precisa ser minimizada em relação a \mathbf{w}_k a cada iteração). Portanto, (3.16) pode ser considerado como quadrático em \mathbf{w}_k , uma vez que os outros vetores de pesos $\mathbf{w}_i (i = 1, \dots, k-1)$ são considerados como termos constantes conhecidos. Então, o problema da filtragem adaptativa cega utilizando o critério do módulo constante pode ser redefinido da seguinte forma:

$$\min_{\mathbf{w}_k} J'(2, 2) = \min_{\mathbf{w}_k} \left(\mathbf{w}_k^H \mathbf{R}_k \mathbf{w}_k - \mathbf{w}_k^H \mathbf{b}_k - \mathbf{b}_k^H \mathbf{w}_k + \sum_{i=1}^k \lambda^{k-i} \rho^2 \right). \quad (3.17)$$

A equação (3.16) é quadrática em \mathbf{w}_k , portanto a solução de (3.17)

$$\mathbf{w}_{k+1} = \mathbf{R}_{k+1}^{-1} \mathbf{b}_{k+1}. \quad (3.18)$$

Todavia, a determinação direta da inversa de \mathbf{R}_{k+1} é computacionalmente caro (da ordem de $(M + 1)^3$). Analogamente ao método RLS para filtros adaptativos (algoritmo 2.3), essa inversa pode ser determinada de forma iterativa, através do lemma da inversão de matrizes [51], da seguinte forma:

$$\mathbf{R}_{k+1}^{-1} = \lambda_f^{-1} \mathbf{R}_k^{-1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1}, \quad (3.19)$$

sendo

$$\mathbf{v}_{k+1} = \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{z}_{k+1}}{1 + \lambda_f^{-1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{z}_{k+1}}, \quad (3.20)$$

que pode ser reescrito como

$$\begin{aligned} \mathbf{v}_{k+1} &= \lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{z}_{k+1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{z}_{k+1} \\ &= [\lambda_f^{-1} \mathbf{R}_k^{-1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1}] \mathbf{z}_{k+1} \\ &= \mathbf{R}_{k+1}^{-1} \mathbf{z}_{k+1} \end{aligned} \quad (3.21)$$

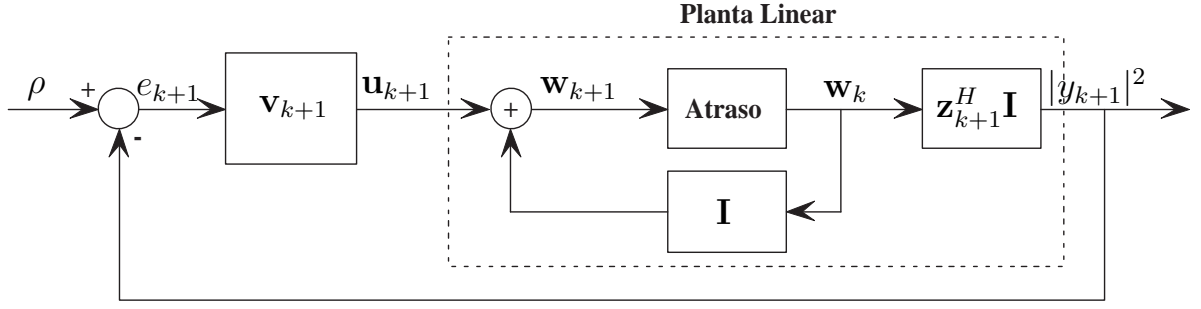


Figura 3.3: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (3.24) correspondente ao método RLS-CMA, sendo \mathbf{v}_{k+1} dada por (3.20).

Substituindo (3.15), (3.19) e (3.21) em (3.18), resulta em

$$\begin{aligned}
 \mathbf{w}_{k+1} &= \mathbf{R}_{k+1}^{-1} \mathbf{b}_{k+1} \\
 &= \lambda_f \mathbf{R}_{k+1}^{-1} \mathbf{b}_k + \mathbf{R}_{k+1}^{-1} \mathbf{z}_{k+1} d_{k+1}^* \\
 &= \mathbf{R}_k^{-1} \mathbf{b}_k - \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{b}_k + \mathbf{v}_{k+1} \rho \\
 &= \mathbf{w}_k + \mathbf{v}_{k+1} (\rho - \mathbf{z}_{k+1}^H \mathbf{w}_k) \\
 &= \mathbf{w}_k + \mathbf{v}_{k+1} e_{k+1}^*,
 \end{aligned} \tag{3.22}$$

sendo o erro *a priori* definido como

$$e_{k+1} = \rho - \mathbf{w}_k^H \mathbf{z}_{k+1}. \tag{3.23}$$

As equações (3.20), (3.19) e (3.22) descrevem o método conhecido na literatura como RLS-CMA (do inglês *Recursive Least Squares Constant Modulus Algorithm*) proposto pela primeira vez em Chen et al. [70]. O método RLS-CMA, apresentado na forma de pseudocódigo no algoritmo 3.2, provê uma melhoria considerável na convergência do SGD-CMA, no entanto, o mesmo apresenta um custo computacional e uma necessidade de armazenamento maior que o SGD-CMA. Do ponto de vista de controle, o método RLS-CMA pode ser representado por um problema de rastreamento de sinal descrito pela figura 3.3 e

Algoritmo 3.2 Método RLS-CMA para filtros adaptativos cegos [70]

Calcular $\mathbf{R}_0^{-1} = \mu^{-1}\mathbf{I}$, $\mu =$ pequena constante positiva

Para $k = 0, 1, \dots, N - 1$

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{w}_k \\ \mathbf{v}_{k+1} &= \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{z}_{k+1}}{1 + \lambda_f^{-1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{z}_{k+1}} \\ e_{k+1} &= \rho - \mathbf{w}_k^H \mathbf{z}_{k+1} \\ \mathbf{w}_{k+1} &= \mathbf{w}_k + \mathbf{v}_{k+1} e_{k+1}^* \\ \mathbf{R}_{k+1}^{-1} &= \lambda_f^{-1} \mathbf{R}_k^{-1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \end{aligned}$$

Fim

pelo seguinte sistema de equações:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \\ |y_{k+1}|^2 = \mathbf{z}_{k+1}^H \mathbf{w}_k \\ e_{k+1} = \rho - |y_{k+1}|^2 \\ \mathbf{v}_{k+1} = \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{z}_{k+1}}{1 + \lambda_f^{-1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{z}_{k+1}} \\ \mathbf{u}_{k+1} = \mathbf{v}_{k+1} e_{k+1} \end{cases} \quad (3.24)$$

O sistema dinâmico (3.24) é composto pela planta linear variante no tempo $\{\mathbf{I}, \mathbf{I}, \mathbf{z}_{k+1}^H \mathbf{I}, \mathbf{0}\}$ e o controlador proporcional ao erro $\{\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{v}_{k+1}\}$ em realimentação negativa, sendo o vetor de ganhos proporcionais \mathbf{v}_{k+1} dado por (3.20).

3.3.3 Método CGCLF-CMA (*Conjugate Gradient Constant Modulus*)

Para resolver o problema (3.17), pode-se desenvolver um novo método com propriedades de convergência comparáveis ao método RLS-CMA [70] a um custo computacional menor. O método CG tem sido amplamente utilizado, na sua versão estocástica, na literatura para o processamento cego de sinais [23, 75–79]. Nesta seção apresenta-se um método de gradiente conjugado para um processamento amostra por amostra em tempo real, (isto é: os dados são tratados assim que se tornam disponíveis na entrada do receptor), utilizando uma janela de dados ponderada de forma exponencialmente decrescente. Para tanto note que, das equações (3.14-3.15) e (3.23), pode-se escrever

$$\mathbf{g}_{k+1} = \mathbf{b}_{k+1} - \mathbf{R}_{k+1} \mathbf{w}_{k+1} = \lambda \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{z}_{k+1} e_{k+1}^*, \quad (3.25)$$

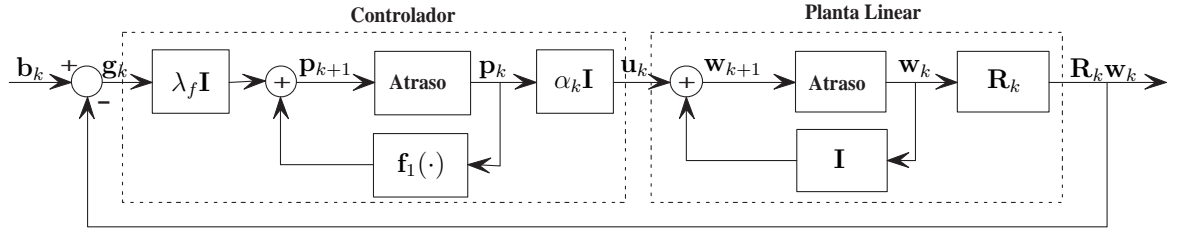


Figura 3.4: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (3.26) correspondente ao método CGCLF-CMA, sendo $f_1(\mathbf{p}_k) = \beta_k \mathbf{p}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{z}_{k+1} e_{k+1}^*$. O problema consiste em determinar os parâmetros de controle α_k e β_k tais que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{R}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$.

sendo o erro *a priori* e_{k+1} dado por (3.23). Portanto, para o caso do problema de filtragem adaptativa cega (3.17), o sistema do tipo gradiente conjugado (2.50) pode ser reescrito como

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \lambda \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{z}_{k+1} e_{k+1}^* \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k, \end{cases} \quad (3.26)$$

sendo \mathbf{z}_{k+1} e e_{k+1} dados por (3.13) e (3.23) respectivamente. Para o sistema (3.26), representado na forma de diagrama de blocos na figura 3.4, um resultado dessa tese por ser enunciado da seguinte forma [80]:

Corolário 2.2. *Com as hipóteses $\inf_k \lambda_{\min}(\mathbf{R}_k^{-1}) = \zeta_1 > 0$ e $\inf_k \lambda_{\min}(\mathbf{R}_k) = \zeta_2 > 0$, as escolhas*

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} \quad (3.27)$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}, \quad (3.28)$$

sendo $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{z}_{k+1} e_{k+1}^*$, são ótimas para as funções de Liapunov escolhidas e garantem a estabilidade assintótica do equilíbrio do sistema (3.26).

Prova do Corolário 2.2. *Substituindo \mathbf{x}_{k+1} por \mathbf{z}_{k+1} e definindo $e_{k+1} = \rho - \mathbf{w}_{k+1}^H \mathbf{z}_{k+1}$, a prova segue da mesma forma que a prova do teorema 2. ■*

O algoritmo 3.3, designado nesta tese como método CGCLF-CMA, apresenta o pseudo-código da extensão do método CG padrão de Hestenes e Stiefel [3] para o caso da filtragem

Algoritmo 3.3 Método CGCLF-CMA para filtros adaptativos cegos

Calcular $\mathbf{z}_0 = \mathbf{x}_0 \mathbf{x}_0^H \mathbf{w}_0$, $\mathbf{R}_0 = \mathbf{z}_0 \mathbf{z}_0^H$, $\mathbf{b}_0 = \rho \mathbf{z}_0$, $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$, $\mathbf{p}_0 = \mathbf{g}_0$

Para $k = 0, 1, \dots, N - 1$

$$\begin{aligned}\mathbf{z}_{k+1} &= \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{w}_k \\ e_{k+1} &= \rho - \mathbf{w}_k^H \mathbf{z}_{k+1} \\ \tilde{\mathbf{g}}_k &= \lambda_f \mathbf{g}_k + \mathbf{z}_{k+1} e_{k+1}^* \\ \alpha_k &= \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} \\ \mathbf{w}_{k+1} &= \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \mathbf{R}_{k+1} &= \lambda_f \mathbf{R}_k + \mathbf{z}_{k+1} \mathbf{z}_{k+1}^H \\ \mathbf{g}_{k+1} &= \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k \\ \beta_k &= -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} \\ \mathbf{p}_{k+1} &= \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k\end{aligned}$$

Fim

adaptativa cega. Note que o método CGCLF-CMA proposto não utiliza nenhum parâmetro empírico e apresenta $M^2 - 9M + 2$ operações de ponto flutuante a menos que o RLS-CMA de Chen et al. [70].

3.3.4 Método NLSD-CMA (*Non-Linear Steepest Descent Constant Modulus*)

Para validar a função custo aproximada (3.12) utilizada pelos métodos RLS-CMA [70] e CGCLF-CMA, compara-se a solução desses dois métodos com a solução obtida aplicando um método de otimização – declividade máxima – diretamente à equação não linear (3.11), sem fazer nenhuma hipótese simplificadora (os resultados dessa comparação estão apresentados na seção 3.6). Denomina-se essa variante de NLSD-CMA, sendo que as letras NL representam as iniciais de “não linear” (para lembrar que a equação não linear (3.11) está sendo minimizada) e as letras SD representam as iniciais de “steepest descent” (declividade máxima em inglês). A equação da atualização recursiva do método NLSD-CMA é obtida da seguinte forma

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mu \nabla J(2, 2), \quad (3.29)$$

Algoritmo 3.4 Método NLSD-CMA para filtros adaptativos cegos

Calcular $\mathbf{z}_0 = \mathbf{x}_0 \mathbf{x}_0^H \mathbf{w}_0$, $\mathbf{R}_0 = \mathbf{z}_0 \mathbf{z}_0^H$, $\mathbf{b}_0 = \rho \mathbf{z}_0$, $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$

Para $k = 0, 1, \dots, N - 1$

$$\begin{aligned} \mathbf{w}_{k+1} &= \mathbf{w}_k + \mu \mathbf{g}_k \\ \mathbf{z}_{k+1} &= \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{w}_{k+1} \\ e_{k+1} &= \rho - \mathbf{w}_{k+1}^H \mathbf{z}_{k+1} \\ \mathbf{R}_{k+1} &= \lambda_f \mathbf{R}_k + \mathbf{z}_{k+1} \mathbf{z}_{k+1}^H \\ \mathbf{g}_{k+1} &= \lambda_f \mathbf{g}_k - \mu \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{z}_{k+1} e_{k+1}^* \end{aligned}$$

Fim

onde μ é um passo pequeno de valor positivo e $\nabla J(2, 2)$ é o vetor gradiente, da função custo $J(2, 2)$ de (3.11), calculado através da diferenciação de $J(2, 2)$ em relação a \mathbf{w}_k como segue

$$\begin{aligned} \nabla J(2, 2) &= \frac{\partial J(2, 2)}{\partial \mathbf{w}_k} = \frac{\partial}{\partial \mathbf{w}_k} \sum_{i=1}^k \lambda_f^{k-i} (\mathbf{w}_k^H \mathbf{x}_i \mathbf{x}_i^H \mathbf{w}_k - \rho)^2 \\ &= 4 \sum_{i=1}^k \lambda^{k-i} \mathbf{x}_i \mathbf{x}_i^H \mathbf{w}_k (\mathbf{w}_k^H \mathbf{x}_i \mathbf{x}_i^H \mathbf{w}_k - \rho) \\ &= 4 \sum_{i=1}^k \lambda^{k-i} (\mathbf{z}_i \mathbf{z}_i^H \mathbf{w}_k - \rho \mathbf{z}_i), \end{aligned} \quad (3.30)$$

sendo \mathbf{z}_i redefinido como

$$\mathbf{z}_i = \mathbf{x}_i \mathbf{x}_i^H \mathbf{w}_k. \quad (3.31)$$

De (3.14) e (3.15), o vetor gradiente (3.30) pode ser reescrito como

$$\nabla J(2, 2) = 4(\mathbf{R}_k \mathbf{w}_k - \mathbf{b}_k). \quad (3.32)$$

Neste caso, note que \mathbf{R}_k depende de \mathbf{z}_k que, por sua vez, depende de \mathbf{w}_k e portanto a equação $\mathbf{R}_k \mathbf{w}_k = \mathbf{b}_k$ (zero de $\nabla J(2, 2)$ em (3.32)) é não linear em relação a \mathbf{w}_k . Das equações (3.32), (3.14) e (3.15), o método NLSD-CMA pode ser resumido pelo seguinte sistema

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \mu \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \mu \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{z}_{k+1} e_{k+1}^* \end{cases} \quad (3.33)$$

sendo o erro *a priori* e_{k+1} dado por (3.23). O algoritmo 3.4 apresenta o método NLSD-CMA na forma de pseudocódigo. O método NLSD-CMA é relativamente simples de implementar. Contudo, a escolha do parâmetro μ é crítica, pois a convergência do método depende de uma boa escolha.

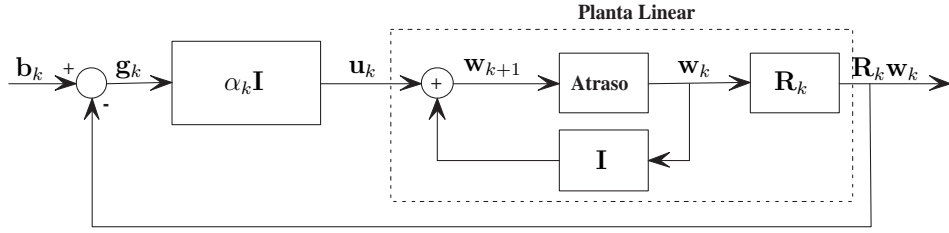


Figura 3.5: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (3.34) correspondente ao método SDCLF-CMA, sendo \mathbf{R}_k e \mathbf{b}_k dados por (3.14) e (3.15) respectivamente. O problema consiste em determinar o parâmetro de controle α_k tal que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\mathbf{R}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$.

3.3.5 Método SDCLF-CMA (*Steepest Descent Constant Modulus*)

O método NLSD-CMA apresenta um custo computacional menor que os métodos RLS-CMA e CGCLF-CMA, e também minimiza a função custo (3.11) sem fazer nenhuma hipótese simplificadora. Porém a escolha do parâmetro fixo μ torna-se difícil em um ambiente com processamento em tempo real uma vez que teria que se saber a localização dos autovalores da matriz de autocorrelação \mathbf{R}_k de todos os dados, antes mesmo que estes estejam disponíveis. Para contornar esse problema pode-se substituir o parâmetro fixo μ por um parâmetro variante no tempo α_k e utilizar a técnica de CLF para determinar um valor ótimo para α_k . Nesse caso o sistema (3.33) é reescrito como

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{z}_{k+1} e_{k+1}^* \end{cases} \quad (3.34)$$

Para o sistema (3.34), representado na forma de diagrama de blocos na figura 3.5, outro resultado desta tese pode ser enunciado da seguinte forma [80]:

Corolário 1.2. Com a hipótese $\inf_k \lambda_{\min}(\mathbf{R}_k) = \zeta_1 > 0$, a escolha

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}{\langle \mathbf{R}_{k+1} \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}, \quad (3.35)$$

sendo $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{z}_{k+1} e_{k+1}^*$, é ótima para a função de Liapunov escolhida e garante a estabilidade assintótica do equilíbrio do sistema (3.33).

Algoritmo 3.5 Método SDCLF-CMA para filtros adaptativos cegos

Calcular $\mathbf{z}_0 = \mathbf{x}_0 \mathbf{x}_0^H \mathbf{w}_0$, $\mathbf{R}_0 = \mathbf{z}_0 \mathbf{z}_0^H$, $\mathbf{b}_0 = \rho \mathbf{z}_0$, $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$

Para $k = 0, 1, \dots, N - 1$

$$\begin{aligned}\mathbf{z}_{k+1} &= \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{w}_k \\ \mathbf{R}_{k+1} &= \lambda_f \mathbf{R}_k + \mathbf{z}_{k+1} \mathbf{z}_{k+1}^H \\ e_{k+1} &= \rho - \mathbf{w}_k^H \mathbf{z}_{k+1} \\ \tilde{\mathbf{g}}_k &= \lambda_f \mathbf{g}_k + \mathbf{z}_{k+1} e_{k+1}^* \\ \alpha_k &= \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}{\langle \mathbf{R}_{k+1} \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle} \\ \mathbf{w}_{k+1} &= \mathbf{w}_k + \alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} &= \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k\end{aligned}$$

Fim

Prova do Corolário 1.2. Substituindo \mathbf{x}_{k+1} por \mathbf{z}_{k+1} e definindo $e_{k+1} = \rho - \mathbf{w}_k^H \mathbf{z}_{k+1}$, a prova segue da mesma forma que a prova do teorema 1. ■

O método baseado no sistema (3.33) e no corolário 1.2, referenciado como SDCLF-CMA, está apresentado no algoritmo 3.5 na forma de pseudocódigo.

3.4 Algoritmos de filtragem adaptativa cega com restrições lineares

3.4.1 Método CSGD-CMA (*Constrained Stochastic Gradient Constant Modulus*)

Analogamente ao problema da filtragem adaptativa com restrições lineares (2.8), o problema da filtragem adaptativa cega com restrições lineares (3.5) pode ser resolvido através de técnicas de otimização com restrições utilizando multiplicadores de Lagrange. Para tanto redefine-se a função custo de (3.5), tomando os valores instantâneos como estimativas do valor esperado tal que

$$J = \frac{1}{4} (\mathbf{w}_k^H \mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k - \rho)^2 + \boldsymbol{\lambda}_k^H (\mathbf{C}^H \mathbf{w}_k - \mathbf{f}), \quad (3.36)$$

sendo $\boldsymbol{\lambda}_k$ o vetor de multiplicadores de Lagrange. Portanto (3.5) da seguinte forma:

$$\min_{\mathbf{w}_k} \left\{ J = \frac{1}{4} (\mathbf{w}_k^H \mathbf{x}_k \mathbf{x}_k^H \mathbf{w}_k - \rho)^2 + \boldsymbol{\lambda}_k^H (\mathbf{C}^H \mathbf{w}_k - \mathbf{f}) \right\}. \quad (3.37)$$

Novamente, o problema (3.37) é resolvido iterativamente

$$\begin{aligned}\mathbf{w}_{k+1} &= \mathbf{w}_k - \mu \nabla_{\mathbf{w}_k} J \\ &= \mathbf{w}_k - \mu(\mathbf{x}_k y_k^* (|y_k|^2 - \rho) + \mathbf{C} \boldsymbol{\lambda}_k).\end{aligned}\quad (3.38)$$

O vetor de multiplicadores de Lagrange $\boldsymbol{\lambda}_k$ é escolhido tal que o vetor \mathbf{w}_{k+1} satisfaça a restrição linear $\mathbf{C}^H \mathbf{w}_{k+1} = \mathbf{f}$, isto é:

$$\mathbf{C}^H \mathbf{w}_{k+1} = \mathbf{f} \Leftrightarrow \mathbf{C}^H \mathbf{w}_k - \mu(\mathbf{C}^H \mathbf{x}_k y_k^* (|y_k|^2 - \rho) + \mathbf{C}^H \mathbf{C} \boldsymbol{\lambda}_k) = \mathbf{f}.$$

Portanto, obtém-se o vetor de multiplicadores de Lagrange tal que:

$$\boldsymbol{\lambda}_k = \frac{-1}{\mu} (\mathbf{C}^H \mathbf{C})^{-1} (\mathbf{f} - \mathbf{C}^H \mathbf{w}_k) - (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H \mathbf{x}_k y_k^* (|y_k|^2 - \rho).\quad (3.39)$$

Substituindo (3.39) em (3.38) resulta em

$$\begin{aligned}\mathbf{w}_{k+1} &= \mathbf{w}_k - \mu \mathbf{x}_k y_k^* (|y_k|^2 - \rho) + \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} (\mathbf{f} - \mathbf{C}^H \mathbf{w}_k) + \mu \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H \mathbf{x}_k y_k^* (|y_k|^2 - \rho) \\ &= (\mathbf{I} - \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H) \mathbf{w}_k - \mu (\mathbf{I} - \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H) \mathbf{x}_k y_k^* (|y_k|^2 - \rho) + \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{f} \\ &= \mathbf{P} \mathbf{w}_k - \mu \mathbf{P} \mathbf{x}_k y_k^* (|y_k|^2 - \rho) + \bar{\mathbf{f}} \\ &= \mathbf{P} (\mathbf{w}_k - \mu \mathbf{x}_k y_k^* (|y_k|^2 - \rho)) + \bar{\mathbf{f}},\end{aligned}\quad (3.40)$$

sendo

$$\begin{aligned}\mathbf{P} &= \mathbf{I} - \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H \\ \bar{\mathbf{f}} &= \mathbf{C} (\mathbf{C}^H \mathbf{C})^{-1} \mathbf{f}.\end{aligned}$$

A equação (3.40) descreve o método referenciado nesta tese como CSGD-CMA (do inglês *Constrained Stochastic Gradient Constant Modulus Algorithm*). Como no caso do método SGD para a filtragem adaptativa cega, o método CSGD-CMA, apresentado na forma de pseudocódigo no algoritmo 2.6, é simples de implementar. Em termos de controle o método CSGD-CMA pode ser representado por um problema de rastreamento de sinal descrito pela

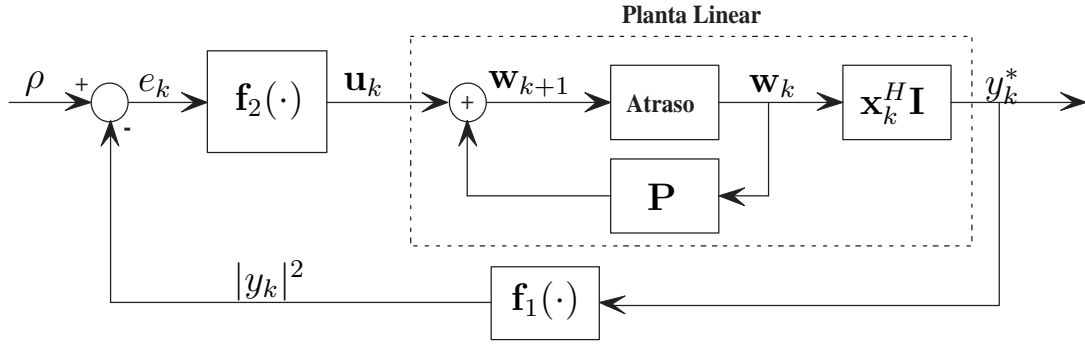


Figura 3.6: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega com restrições lineares (3.37) pelo sistema dinâmico (3.41) correspondente ao método CSGD-CMA, sendo $\mathbf{f}_1(y_k^*) = |y_k|^2$ e $\mathbf{f}_2(e_k) = \mu \mathbf{P} \mathbf{x}_k y_k^* e_k + \bar{\mathbf{f}}$. O problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $|y_k|^2 \rightarrow \rho$, quando $k \rightarrow \infty$.

Algoritmo 3.6 Método CSGD-CMA para filtros adaptativos cegos com restrições lineares

Calcular $\bar{\mathbf{f}} = \mathbf{C}(\mathbf{C}^H \mathbf{C})^{-1} \mathbf{f}$, $\mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H$

Para $k = 0, 1, \dots, N - 1$

$$y_k = \mathbf{w}_k^H \mathbf{x}_k$$

$$e_k = \rho - y_k y_k^*$$

$$\mathbf{w}_{k+1} = \mathbf{P}(\mathbf{w}_k + \mu \mathbf{x}_k y_k^* e_k) + \bar{\mathbf{f}}$$

Fim

figura 3.6 e pelo seguinte sistema de equações:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \mathbf{u}_k \\ y_k = \mathbf{w}_k^H \mathbf{x}_k \\ e_k = \rho - y_k^* y_k \\ \mathbf{u}_k = \mu \mathbf{P} \mathbf{x}_k y_k^* e_k + \bar{\mathbf{f}} \end{cases} \quad (3.41)$$

O sistema dinâmico (3.41) é constituído pela planta linear variante no tempo $\{\mathbf{P}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, \mathbf{0}\}$ e o controlador $\mathbf{f}_2(e_k) = \mu \mathbf{P} \mathbf{x}_k y_k^* e_k + \bar{\mathbf{f}}$, função afim do erro em realimentação negativa. Novamente o problema consiste em determinar o parâmetro de controle μ tal que $e_k \rightarrow 0$ e $|y_k|^2 \rightarrow \rho$, quando $k \rightarrow \infty$.

3.4.2 Método CRLS-CMA (*Constrained Recursive Least Squares Constant Modulus*)

O método CSGD-CMA herda todas as vantagens e desvantagens do método SGD-CMA ou seja, simplicidade de implementação e convergência dependente do passo μ e do condicionamento de \mathbf{R}_k . A taxa de convergência pode ser melhorada através da utilização da janela de dados ponderada de forma exponencialmente decrescente do método RLS-CMA. Nesse caso penaliza-se a função custo (3.16) através de multiplicadores de Lagrange da seguinte forma

$$J'(2, 2) = \frac{1}{2}(\mathbf{w}_k^H \mathbf{R}_k \mathbf{w}_k - \mathbf{w}_k^H \mathbf{b}_k - \mathbf{b}_k^H \mathbf{w}_k + \sum_{i=1}^k \lambda^{k-i} \rho^2) + \boldsymbol{\lambda}_k^H (\mathbf{C}^H \mathbf{w}_k - \mathbf{f}), \quad (3.42)$$

sendo \mathbf{R}_k e \mathbf{b}_k dados por (3.14) e (3.15). Portanto o problema (3.5) resulta em

$$\min_{\mathbf{w}_k} \{J'(2, 2) = \frac{1}{2}(\mathbf{w}_k^H \mathbf{R}_k \mathbf{w}_k - \mathbf{w}_k^H \mathbf{b}_k - \mathbf{b}_k^H \mathbf{w}_k + \sum_{i=1}^k \lambda^{k-i} \rho^2) + \boldsymbol{\lambda}_k^H (\mathbf{C}^H \mathbf{w}_k - \mathbf{f})\}. \quad (3.43)$$

A solução de (3.43) pode ser encontrada facilmente por analogia ao método CRLS para filtros adaptativos com restrições lineares. Para tanto reescrevem-se as equações (2.77), (2.78) e (2.79) substituindo \mathbf{x}_{k+1} por \mathbf{z}_{k+1} conforme segue:

$$\begin{aligned} \boldsymbol{\Gamma}_{k+1} &:= \mathbf{R}_{k+1}^{-1} \mathbf{C} = \lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{C} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{C} \\ &= \lambda_f^{-1} \boldsymbol{\Gamma}_k - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \boldsymbol{\Gamma}_k \end{aligned} \quad (3.44)$$

$$\begin{aligned} \boldsymbol{\Psi}_{k+1} &:= \mathbf{C}^H \boldsymbol{\Gamma}_{k+1} = \lambda_f^{-1} \mathbf{C}^H \boldsymbol{\Gamma}_k - \lambda_f^{-1} \mathbf{C}^H \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \boldsymbol{\Gamma}_k \\ &= \lambda_f^{-1} \boldsymbol{\Psi}_k - \lambda_f^{-1} \mathbf{C}^H \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \boldsymbol{\Gamma}_k \end{aligned} \quad (3.45)$$

$$\boldsymbol{\Psi}_{k+1}^{-1} = \lambda_f \boldsymbol{\Psi}_k^{-1} + \lambda_f \mathbf{l}_{k+1} \mathbf{z}_{k+1}^H \boldsymbol{\Gamma}_k \boldsymbol{\Psi}_k^{-1}, \quad (3.46)$$

sendo \mathbf{v}_{k+1} dado pela equação (3.20) e

$$\mathbf{l}_{k+1} = \frac{\boldsymbol{\Psi}_k^{-1} \mathbf{C}^H \mathbf{v}_{k+1}}{1 - \mathbf{z}_{k+1}^H \boldsymbol{\Gamma}_k \boldsymbol{\Psi}_k^{-1} \mathbf{C}^H \mathbf{v}_{k+1}} \quad (3.47)$$

Finalmente a solução de (3.43) é encontrada através do seguinte processo iterativo

$$\mathbf{w}_{k+1} = \mathbf{P}_{k+1}(\mathbf{w}_k + \mathbf{v}_{k+1}e_{k+1}^*) + \bar{\mathbf{f}}_{k+1}, \quad (3.48)$$

sendo

$$e_{k+1} = \rho - \mathbf{w}_k^H \mathbf{z}_{k+1} \quad (3.49)$$

$$\mathbf{P}_{k+1} = \mathbf{I} - \Gamma_{k+1} \Psi_{k+1}^{-1} \mathbf{C}^H \quad (3.50)$$

$$\bar{\mathbf{f}}_{k+1} = \Gamma_{k+1} \Psi_{k+1}^{-1} \mathbf{f}. \quad (3.51)$$

As equações (3.44-3.51) descrevem o método conhecido como CRLS-CMA (do inglês *Constrained Recursive Least Squares Constant Modulus Algorithm*) [81], apresentada no algoritmo 3.7 na forma de pseudocódigo. Em termos de controle, o método CRLS-CMA é representado por sistema dinâmico constituído por uma planta linear $\{\mathbf{P}_{k+1}, \mathbf{I}, \mathbf{z}_{k+1}^H \mathbf{I}, \mathbf{0}\}$ com um controlador adaptativa não linear em configuração de realimentação negativa descrito pela figura 3.7 e pelo seguinte sistema de equações:

$$\left\{ \begin{array}{l} \mathbf{w}_{k+1} = \mathbf{P}_{k+1} \mathbf{w}_k + \mathbf{u}_k \\ |y_{k+1}|^2 = \mathbf{z}_{k+1}^H \mathbf{w}_k \\ e_{k+1} = \rho - |y_{k+1}|^2 \\ \mathbf{v}_{k+1} = \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{z}_{k+1}}{1 + \lambda_f^{-1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{z}_{k+1}} \\ \mathbf{u}_{k+1} = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1} + \bar{\mathbf{f}}_{k+1} \end{array} \right. \quad (3.52)$$

O sistema dinâmico (3.52) é composto pela planta linear variante no tempo $\{\mathbf{P}_{k+1}, \mathbf{I}, \mathbf{z}_{k+1}^H \mathbf{I}, \mathbf{0}\}$ e o controlador $\mathbf{f}_1(e_{k+1}) = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1} + \bar{\mathbf{f}}_{k+1}$ em realimentação negativa, sendo o vetor de ganhos proporcionais \mathbf{v}_{k+1} dado por (3.20).

3.4.3 Método com estrutura GSC

A estrutura GSC proposta em Griffiths e Jim [61] como método alternativo de resolução do problema da filtragem adaptativa com restrições lineares (2.8) pode ser utilizada na resolução do problema da filtragem adaptativa cega com restrições lineares (3.5) [24, 82, 83]. Para tanto determina-se uma matriz de bloqueio tal que $\mathbf{B}^H \mathbf{C} = \mathbf{0}$, define-se $\mathbf{x}'_k = \mathbf{B}^H \mathbf{x}_k$ para

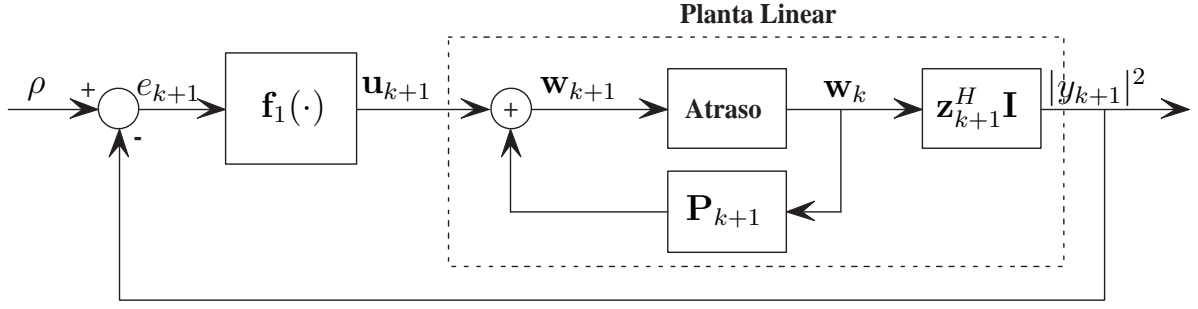


Figura 3.7: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.43) pelo sistema dinâmico (3.52) correspondente ao método CRLS-CMA, sendo $f_1(e_{k+1}) = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1} + \bar{\mathbf{f}}_{k+1}$ e \mathbf{v}_{k+1} dado por (3.20).

Algoritmo 3.7 Método CRLS-CMA para filtros adaptativos cegos com restrições lineares

Calcular $\mathbf{R}_0^{-1} = \mu^{-1} \mathbf{I}$, $\mathbf{\Gamma}_0 = \mathbf{R}_0^{-1} \mathbf{C}$, $\mathbf{\Psi}_0^{-1} = \mu^{-1} \mathbf{I}$, $\mu =$ pequena constante positiva

Calcular $\mathbf{P}_0 = \mathbf{I} - \mathbf{\Gamma}_0 \mathbf{\Psi}_0^{-1} \mathbf{C}^H$, $\bar{\mathbf{f}}_0 = \mathbf{\Gamma}_0 \mathbf{\Psi}_0^{-1} \mathbf{f}$

Para $k = 0, 1, \dots, N - 1$

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{w}_k \\ \mathbf{v}_{k+1} &= \frac{\lambda_f^{-1} \mathbf{R}_k^{-1} \mathbf{z}_{k+1}}{1 + \lambda_f^{-1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \mathbf{z}_{k+1}} \\ \mathbf{R}_{k+1}^{-1} &= \lambda_f^{-1} \mathbf{R}_k^{-1} - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \mathbf{R}_k^{-1} \\ \mathbf{\Gamma}_{k+1} &= \lambda_f^{-1} \mathbf{\Gamma}_k - \lambda_f^{-1} \mathbf{v}_{k+1} \mathbf{z}_{k+1}^H \mathbf{\Gamma}_k \\ \mathbf{l}_{k+1} &= \frac{\mathbf{\Psi}_k^{-1} \mathbf{C}^H \mathbf{v}_{k+1}}{1 - \mathbf{z}_{k+1}^H \mathbf{\Gamma}_k \mathbf{\Psi}_k^{-1} \mathbf{C}^H \mathbf{v}_{k+1}} \\ \mathbf{\Psi}_{k+1}^{-1} &= \lambda_f \mathbf{\Psi}_k^{-1} + \lambda_f \mathbf{l}_{k+1} \mathbf{z}_{k+1}^H \mathbf{\Gamma}_k \mathbf{\Psi}_k^{-1} \\ e_{k+1} &= \rho - \mathbf{w}_k^H \mathbf{z}_{k+1} \\ \mathbf{P}_{k+1} &= \mathbf{I} - \mathbf{\Gamma}_{k+1} \mathbf{\Psi}_{k+1}^{-1} \mathbf{C}^H \\ \bar{\mathbf{f}}_{k+1} &= \mathbf{\Gamma}_{k+1} \mathbf{\Psi}_{k+1}^{-1} \mathbf{f} \\ \mathbf{w}_{k+1} &= \mathbf{P}_{k+1} (\mathbf{w}_k + \mathbf{v}_{k+1} e_{k+1}) + \bar{\mathbf{f}}_{k+1} \end{aligned}$$

Fim

resolver o problema irrestrito (3.4) pelo método SGD-CMA, ou então define-se $\mathbf{z}'_k = \mathbf{B}^H \mathbf{z}_k$ para resolver (3.4) pelos métodos RLS-CMA, CGCLF-CMA, NLSD-CMA e SDCLF-CMA. Uma vez encontrada a solução \mathbf{w}'_k do problema irrestrito (3.4), a solução do problema restrito (3.5) é determinada através da equação (2.89), ou seja $\mathbf{w}_k = \bar{\mathbf{f}} - \mathbf{B} \mathbf{w}'_k$.

3.4.4 Método CCGCLF-CMA (*Constrained Conjugate Gradient Constant Modulus*)

Analogamente ao caso do problema da filtragem adaptativa com restrições lineares (2.8), a equação (2.89) pode ser utilizada para desenvolver novos métodos numéricos para a reso-

lução de (3.5) com desempenho comparável ao dos métodos CRLS-CMA. De fato, através da resolução do problema irrestrito (3.4) pelo método CGCLF-CMA e das equações (2.89) e (2.91), pode-se desenvolver um novo método numérico de resolução do problema restrito (3.5). Nesse caso, de (2.89) e do algoritmo 3.3, reescrevem-se as equações (2.104-2.106) da seguinte forma

$$\mathbf{w}_{k+1} = \bar{\mathbf{f}} - \mathbf{B}(\mathbf{w}'_k + \alpha_k \mathbf{p}'_k) \quad (3.53)$$

$$\mathbf{g}'_{k+1} = \lambda_f \mathbf{g}'_k - \alpha_k \mathbf{R}'_{k+1} \mathbf{p}'_k + \mathbf{z}'_{k+1} e'^*_{k+1} \quad (3.54)$$

$$\mathbf{p}'_{k+1} = \mathbf{g}'_{k+1} + \beta_k \mathbf{p}'_k. \quad (3.55)$$

Substituindo (2.95) em (3.53) e definindo $\mathbf{p}_k = \mathbf{B}\mathbf{p}'_k$ e $\mathbf{g}_{k+1} = \mathbf{B}\mathbf{g}'_{k+1}$ resulta em

$$\mathbf{w}_{k+1} = \mathbf{P}\mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k \quad (3.56)$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k \quad (3.57)$$

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{z}}_{k+1} \bar{e}^*_{k+1}, \quad (3.58)$$

sendo $\bar{\mathbf{R}}_{k+1} = \mathbf{P}\mathbf{R}_{k+1}\mathbf{P}^H$, $\bar{\mathbf{z}}_{k+1} = \mathbf{P}\mathbf{z}_{k+1}$ e $\bar{e}_{k+1} = \rho - \mathbf{w}_k^H \bar{\mathbf{z}}_{k+1}$. As equações (3.56-3.58) descrevem o sistema do tipo gradiente conjugado correspondente à resolução de (3.5) pelo método CG

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P}\mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{z}}_{k+1} \bar{e}^*_{k+1} \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k. \end{cases} \quad (3.59)$$

Para o sistema (3.59), representado na forma de diagrama de blocos na figura 3.8, outro resultado desta tese pode ser enunciado da seguinte forma:

Corolário 2.3. *Com as hipóteses $\inf_k \lambda_{\min}(\bar{\mathbf{R}}_k^{-1}) = \zeta_1 > 0$ e $\inf_k \lambda_{\min}(\bar{\mathbf{R}}_k) = \zeta_2 > 0$, as escolhas*

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \quad (3.60)$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}, \quad (3.61)$$

sendo $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \bar{\mathbf{z}}_{k+1} \bar{e}^*_{k+1}$, são ótimas para as funções de Liapunov escolhidas e garantem a estabilidade assintótica do equilíbrio do sistema (3.59).

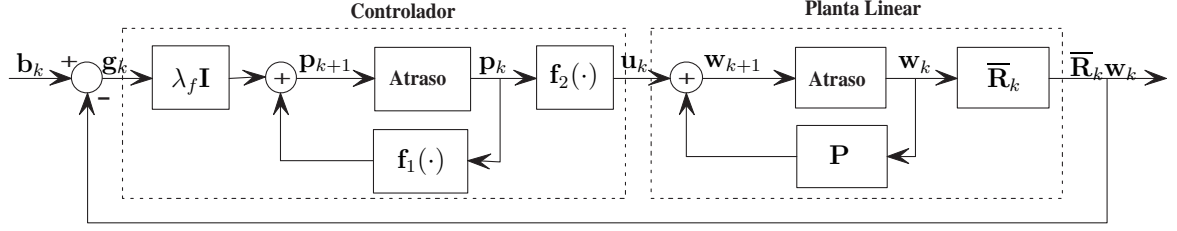


Figura 3.8: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.5) pelo sistema dinâmico (3.59) correspondente ao método CCGCLF-CMA, sendo $f_1(\mathbf{p}_k) = \beta_k \mathbf{p}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{z}}_{k+1} \bar{\mathbf{e}}_{k+1}^*$ e $f_2(\mathbf{p}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k$. O problema consiste em determinar os parâmetros de controle α_k e β_k tais que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\bar{\mathbf{R}}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$.

Algoritmo 3.8 Método CCGCLF-CMA para filtros adaptativos cegos com restrições lineares

Calcular $\mathbf{w}_0 = \bar{\mathbf{f}} = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \bar{\mathbf{f}}$, $\bar{\mathbf{R}}_0 = \mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T$, $\mathbf{g}_0 = \mathbf{p}_0 = \mathbf{0}$

Para $k = 0, 1, \dots, N - 1$

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{w}_k \\ \bar{\mathbf{z}}_{k+1} &= \mathbf{P} \mathbf{z}_{k+1} \\ \bar{\mathbf{R}}_{k+1} &= \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{z}}_{k+1} \bar{\mathbf{z}}_{k+1}^H \\ \bar{\mathbf{e}}_{k+1} &= d_{k+1} + \mathbf{w}_k^H \bar{\mathbf{z}}_{k+1} \\ \tilde{\mathbf{g}}_k &= \lambda_f \mathbf{g}_k + \bar{\mathbf{z}}_{k+1} \bar{\mathbf{e}}_{k+1}^* \\ \alpha_k &= \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \\ \mathbf{w}_{k+1} &= \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} &= \tilde{\mathbf{g}}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \\ \beta_k &= -\frac{\langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \\ \mathbf{p}_{k+1} &= \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k \end{aligned}$$

Fim

Prova do Corolário 2.3. Substituindo \mathbf{x}_{k+1} , \mathbf{R}_{k+1} e \mathbf{e}_{k+1} por $\bar{\mathbf{z}}_{k+1}$, $\bar{\mathbf{R}}_{k+1}$ e $\bar{\mathbf{e}}_{k+1}$ respectivamente, a prova segue da mesma forma que a prova do teorema 2. ■

O algoritmo 3.8, referenciado nesta tese como CCGCLF-CMA, apresenta o pseudocódigo da extensão do método CG, baseado no corolário 2.3, para o caso da filtragem adaptativa cega com restrições lineares.

3.4.5 Método CSDCLF-CMA (*Constrained Steepest Descent Constant Modulus*)

O método CCGCLF-CMA baseia-se no método CGCLF-CMA que, por sua vez, minimiza a função custo aproximada (3.16). Para validar a utilização da função custo aproximada (3.16) no problema restrito (3.5), compara-se a solução resultante do método com a solução

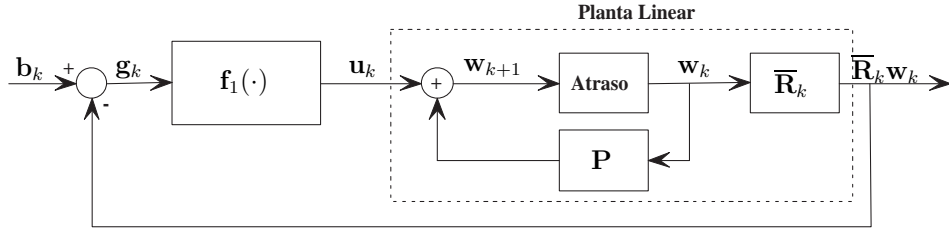


Figura 3.9: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.5) pelo sistema dinâmico (3.66) correspondente ao método CSDCLF-CMA, sendo $f_1(\mathbf{g}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k$. O problema consiste em determinar o parâmetro de controle α_k tal que $\mathbf{g}_k \rightarrow \mathbf{0}$ e $\bar{\mathbf{R}}_k \mathbf{w}_k \rightarrow \mathbf{b}_k$, quando $k \rightarrow \infty$.

proveniente de um outro novo método numérico (análogo ao método CSD-CLF para filtros adaptativos com restrições lineares) baseado no método SDCLF-CMA e na relação (2.89).

Da equação (2.89) e do algoritmo 3.5, reescrevem-se as equações (2.93) e (2.94) tal que

$$\mathbf{w}_{k+1} = \bar{\mathbf{f}} - \mathbf{B}(\mathbf{w}'_k + \alpha_k \mathbf{g}'_k) \quad (3.62)$$

$$\mathbf{g}'_{k+1} = \lambda_f \mathbf{g}'_k - \alpha_k \mathbf{R}'_{k+1} \mathbf{g}'_k + \mathbf{z}'_{k+1} e'^*_{k+1}. \quad (3.63)$$

Substituindo (2.95) em (3.62) e definindo $\mathbf{g}_{k+1} = \mathbf{B} \mathbf{g}'_{k+1}$ resulta em

$$\mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k \quad (3.64)$$

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k + \bar{\mathbf{z}}_{k+1} \bar{e}^*_{k+1}, \quad (3.65)$$

sendo $\bar{\mathbf{R}}_{k+1} = \mathbf{P} \mathbf{R}_{k+1} \mathbf{P}^H$, $\bar{\mathbf{z}}_{k+1} = \mathbf{P} \mathbf{z}_{k+1}$ e $\bar{e}_{k+1} = \rho - \mathbf{w}_k^H \bar{\mathbf{z}}_{k+1}$. As equações (3.64-3.65) descrevem o sistema do tipo declividade máxima correspondente à resolução de (3.5) pelo método SD

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k + \bar{\mathbf{z}}_{k+1} \bar{e}^*_{k+1}. \end{cases} \quad (3.66)$$

Para o sistema (3.66), representado na forma de diagrama de blocos na figura 3.9, outro resultado desta tese pode ser enunciado da seguinte forma:

Corolário 1.3. Com a hipótese $\inf_k \lambda_{\min}(\bar{\mathbf{R}}_k) = \zeta_1 > 0$, a escolha

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}{\langle \bar{\mathbf{R}}_{k+1} \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}, \quad (3.67)$$

sendo $\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \bar{\mathbf{z}}_{k+1} \bar{e}^*_{k+1}$, é ótima para a função de Liapunov escolhida e garante a

Algoritmo 3.9 Método CSDCLF-CMA para filtros adaptativos cegos com restrições lineares

Calcular $\mathbf{w}_0 = \bar{\mathbf{f}} = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{f}$, $\bar{\mathbf{R}}_0 = \mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T$, $\mathbf{g}_0 = \mathbf{0}$

Para $k = 0, 1, \dots, N - 1$

$$\mathbf{z}_{k+1} = \mathbf{x}_{k+1} \mathbf{x}_{k+1}^H \mathbf{w}_k$$

$$\bar{\mathbf{z}}_{k+1} = \mathbf{P} \mathbf{z}_{k+1}$$

$$\bar{\mathbf{R}}_{k+1} = \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{z}}_{k+1} \bar{\mathbf{z}}_{k+1}^H$$

$$\bar{e}_{k+1} = \rho - \mathbf{w}_k^H \bar{\mathbf{z}}_{k+1}$$

$$\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \bar{\mathbf{z}}_{k+1} \bar{e}_{k+1}^*$$

$$\alpha_k = \frac{\langle \tilde{\mathbf{g}}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}{\langle \bar{\mathbf{R}}_{k+1} \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k$$

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k$$

Fim

estabilidade assintótica do equilíbrio do sistema (3.66).

Prova do Corolário 1.3. *Substituindo \mathbf{x}_{k+1} , \mathbf{R}_{k+1} e e_{k+1} por $\bar{\mathbf{x}}_{k+1}$, $\bar{\mathbf{R}}_{k+1}$ e \bar{e}_{k+1} respectivamente, a prova segue da mesma forma que a prova do teorema 1.3. ■*

O algoritmo 3.9 apresenta o pseudocódigo do novo método, referenciado nesta tese como CSDCLF-CMA, baseado no sistema (3.66) e no corolário 1.3.

3.5 Síntese dos controladores

Nesta seção apresentar-se-á, na tabela 3.1, uma breve comparação dos sistemas dinâmicos correspondentes aos métodos de filtragem adaptativa com ou sem restrições lineares e de filtragem adaptativa cega com ou sem restrições lineares estudados nesta tese. Pode-se ver que o método LMS é equivalente a uma planta linear variante no tempo em realimentação negativa com um controlador proporcional ao erro sendo que o ganho proporcional é determinado apenas com informações atuais do sinal \mathbf{x}_k . Os métodos RLS e SDCLF correspondem a uma planta linear variante no tempo em realimentação negativa com controlador proporcional ao erro sendo que o ganho proporcional é determinado com informações atuais e passadas do sinal \mathbf{x}_k . O método CGCLF é equivalente a uma planta linear variante no tempo em realimentação negativa com um controlador proporcional e integral ao erro sendo que os ganhos proporcional e integral são determinados com informações atuais e passadas do sinal \mathbf{x}_k . As diferenças entre os métodos LMS, RLS, SDCLF, CGCLF e suas respectivas versões com restrições (CLMS, CRLS, CSDCLF, CCGCLF) consistem na introdução da

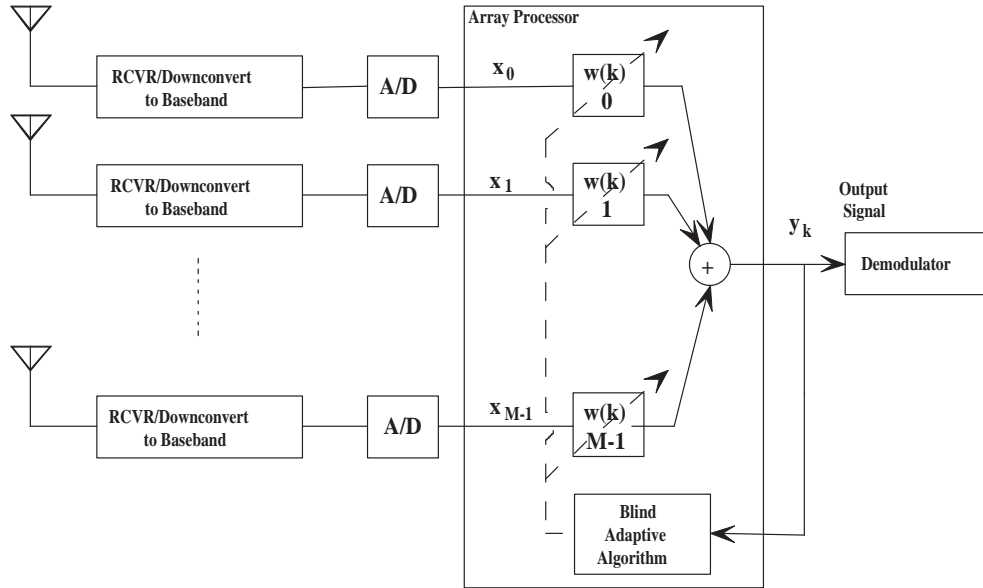


Figura 3.10: Conjunto de antenas adaptativas não-supervisionadas.

matriz de projeção \mathbf{P} como matriz do sistema na substituição dos controladores correspondentes por funções afins com termos em \mathbf{P} e $\bar{\mathbf{f}}$. Para o caso dos algoritmos cegos, o sinal \mathbf{x}_k é substituído por \mathbf{z}_k no caso dos métodos (C)RLS-CMA, (C)SDCLF-CMA e (C)CGCLF-CMA, enquanto a variável y_k é introduzida na função do controlador no caso dos métodos (C)SGD-CMA.

3.6 Simulações numéricas

3.6.1 Sistema de antenas adaptativas

Considere o sistema de antenas adaptativas (*beamformer*), mostrado na figura 3.10, com K usuários simultâneos, transmitindo sinais complexos $s_i(t)$, $1 \leq i \leq K$ com forma de onda de módulos constantes ($|s_i(t)| = 1$) em um cenário de comunicação sem fio. Os sinais contínuos no tempo são recebidos por um conjunto de L antenas, transmitidos e amostrados através de um circuito digital, resultando no sinal de entrada discreto no tempo $x_l(k)$, $1 \leq l \leq L$. Os sinais discretos no tempo recebidos por cada antena elementar são multiplicados por um vetor de pesos de valores complexos $w_l(k)$, $1 \leq l \leq L$, e são adicionados um com o

Método	Controlador	Planta	Comentários
LMS	$\mathbf{f}(e_k) = \mu \mathbf{x}_k e_k^*$	$\{\mathbf{I}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, \mathbf{0}\}$	Cont. Propor., ganho c / inform. instantâneas.
RLS	$\mathbf{f}(e_{k+1}) = \mathbf{v}_{k+1} e_{k+1}^*$	$\{\mathbf{I}, \mathbf{I}, \mathbf{x}_{k+1}^H \mathbf{I}, \mathbf{0}\}$	Cont. Propor., ganho c / inform. do passado.
SDCLF	$\mathbf{f}(\mathbf{g}_k) = \alpha_k \mathbf{g}_k$	$\{\mathbf{I}, \mathbf{I}, \mathbf{R}_k, \mathbf{0}\}$	Cont. Propor., ganho c / inform. do passado.
CGCLF	$\mathbf{f}(\mathbf{p}_k) = \alpha_k \mathbf{p}_k$	$\{\mathbf{I}, \mathbf{I}, \mathbf{R}_k, \mathbf{0}\}$	Con. Prop. Int., ganhos c / informações do passado.
CLMS	$\mathbf{f}(e_k) = \mu \mathbf{P} \mathbf{x}_k e_k^* + \bar{\mathbf{f}}$	$\{\mathbf{P}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, \mathbf{0}\}$	Cont. → fun. afim Prop. ao erro, ganho c / inform. instantâneas.
CRLS	$\mathbf{f}(e_{k+1}) = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1}^* + \bar{\mathbf{f}}_{k+1}$	$\{\mathbf{P}_{k+1}, \mathbf{I}, \mathbf{x}_{k+1}^H \mathbf{I}, \mathbf{0}\}$	Cont. → fun. afim Prop. ao erro, ganho c / inform. do passado.
CSDCLF	$\mathbf{f}(\mathbf{g}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k$	$\{\mathbf{P}, \mathbf{I}, \bar{\mathbf{R}}_k, \mathbf{0}\}$	Cont. → fun. afim Prop. ao erro, ganho c / inform. do passado.
CCGCLF	$\mathbf{f}(\mathbf{p}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k$	$\{\mathbf{P}, \mathbf{I}, \bar{\mathbf{R}}_k, \mathbf{0}\}$	Cont. → fun. afim Prop. Int. ao erro, ganho c / inform. do passado.
SGD-CMA	$\mathbf{f}(e_k) = \mu \mathbf{x}_k y_k^* e_k$	$\{\mathbf{I}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, \mathbf{0}\}$	Cont. Propor., ganho c / inform. instantâneas.
RLS-CMA	$\mathbf{f}(e_{k+1}) = \mathbf{v}_{k+1} e_{k+1}^*$	$\{\mathbf{I}, \mathbf{I}, \mathbf{z}_{k+1}^H \mathbf{I}, \mathbf{0}\}$	Cont. Propor., ganho c / inform. do passado.
SDCLF-CMA	$\mathbf{f}(\mathbf{g}_k) = \alpha_k \mathbf{g}_k$	$\{\mathbf{I}, \mathbf{I}, \mathbf{R}_k, \mathbf{0}\}$	Cont. Propor., ganho c / inform. do passado.
CGCLF-CMA	$\mathbf{f}(\mathbf{p}_k) = \alpha_k \mathbf{p}_k$	$\{\mathbf{I}, \mathbf{I}, \mathbf{R}_k, \mathbf{0}\}$	Con. Prop. Int., ganhos c / informações do passado.
CSGD-CMA	$\mathbf{f}(e_k) = \mu \mathbf{P} \mathbf{x}_k y_k^* e_k + \bar{\mathbf{f}}$	$\{\mathbf{P}, \mathbf{I}, \mathbf{x}_k^H \mathbf{I}, \mathbf{0}\}$	Cont. → fun. afim Prop. ao erro, ganho c / inform. instantâneas.
CRLS-CMA	$\mathbf{f}(e_{k+1}) = \mathbf{P}_{k+1} \mathbf{v}_{k+1} e_{k+1}^* + \bar{\mathbf{f}}_{k+1}$	$\{\mathbf{P}_{k+1}, \mathbf{I}, \mathbf{z}_{k+1}^H \mathbf{I}, \mathbf{0}\}$	Cont. → fun. afim Prop. ao erro, ganho c / inform. do passado.
CSDCLF-CMA	$\mathbf{f}(\mathbf{g}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{g}_k$	$\{\mathbf{P}, \mathbf{I}, \bar{\mathbf{R}}_k, \mathbf{0}\}$	Cont. → fun. afim Prop. ao erro, ganho c / inform. do passado.
CCGCLF-CMA	$\mathbf{f}(\mathbf{p}_k) = \bar{\mathbf{f}} - \alpha_k \mathbf{p}_k$	$\{\mathbf{P}, \mathbf{I}, \bar{\mathbf{R}}_k, \mathbf{0}\}$	Cont. → fun. afim Prop. Int. ao erro, ganho c / inform. do passado.

Tabela 3.1: Comparação dos sistemas dinâmicos correspondentes aos métodos de filtragem adaptativa com ou sem restrições lineares e de filtragem adaptativa cega com ou sem restrições lineares.

outro para formar a saída do *beamformer* y_k tal que

$$y_k = \sum_{l=1}^L w_l^* x_l(k).$$

Utilizando o formato vetorial para designar os pesos do *beamformer* e os sinais induzidos em cada antena elementar, isto é,

$$\begin{aligned} \mathbf{w} &= [w_1, w_2, \dots, w_L]^T \\ \mathbf{x}_k &= [x_1(k), x_2(k), \dots, x_L(k)]^T, \end{aligned}$$

a saída do *beamformer* resulta em

$$y_k = \mathbf{w}^H \mathbf{x}_k.$$

O objetivo do conjunto de antenas adaptativas é extrair o sinal desejado, $s_1(t)$ por exemplo, encontrando um vetor \mathbf{w} de pesos ótimos (ótimo no contexto de alguma norma) a partir do critério do módulo constante, sendo que em uma implementação prática os sinais recebidos são corrompidos pelo desvanecimento devido aos caminhos múltiplos, interferências e outros ruídos que causam variações na amplitude.

Para avaliar o desempenho dos algoritmos propostos, um conjunto uniforme de dez antenas elementares é utilizado, da mesma forma que em Chen et al. [70]. A distância entre os elementos é igual à metade do comprimento de onda do sinal desejado. O desempenho do “*beamformer*” é medido a partir do quociente entre o sinal e a interferência mais o ruído SINR (do inglês *signal-to-interference-plus-noise ratio*) definido como

$$\text{SINR}_i = \frac{\mathbf{w}^H \mathbf{r}_{a_i} r_{s_i} \mathbf{r}_{a_i}^H \mathbf{w}}{\mathbf{w}^H \mathbf{R}_I \mathbf{w}}, \quad (3.68)$$

sendo r_{s_i} a potência da i -ésima fonte, \mathbf{r}_{a_i} é o vetor associado de transmissão, e \mathbf{R}_I é a matriz de autocorrelação da interferência no ambiente e contém o ruído do ambiente na sua diagonal. Compara-se a taxa de convergência e a propriedade de rastreamento dos métodos propostos CGCLF-CMA, NLSD-CMA e SDCLF-CMA com os métodos RLS-CMA e SGD-CMA. A propriedade de rastreamento é estudada através de uma mudança súbita do número de fontes de interferência, após o período inicial de convergência. A potência do ruído é $\sigma_n^2 = 0.1$, a fase $\psi_i(n)$, $i \leq d$ de cada sinal $s_i(n) = e^{j\psi_i(n)}$ é distribuída de forma indepen-

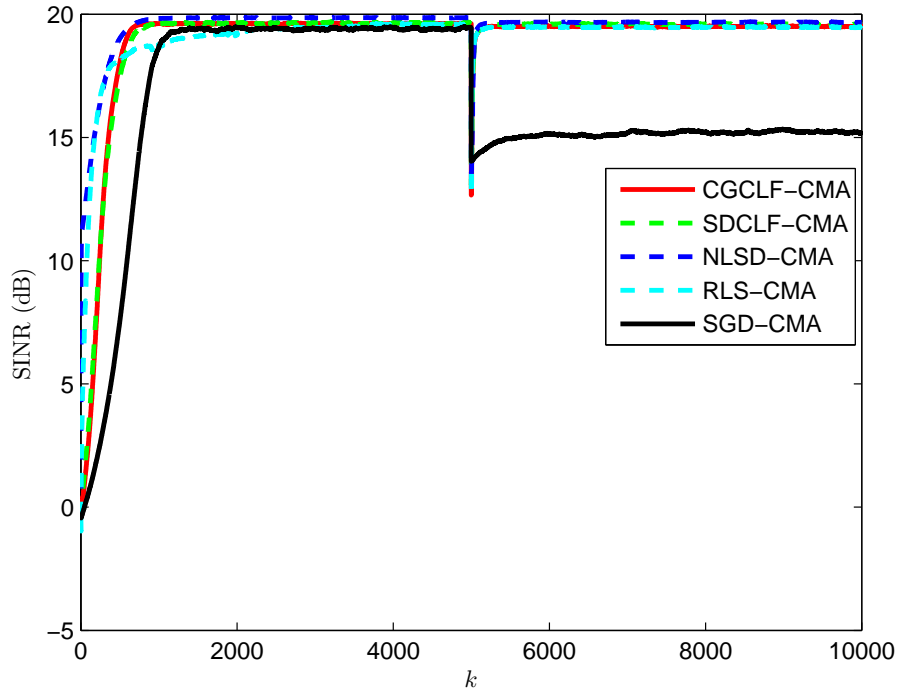


Figura 3.11: Quociente entre o sinal e a interferência mais o ruído dos métodos CGCLF-CMA, RLS-CMA, SGD-CMA e NLS-D-CMA

dente e uniforme no intervalo $[-\pi, \pi]$, onde d é o número de fontes de sinais. Nas primeiras 5000 iterações, existem dois sinais com ângulo de incidência DOA (do inglês *Direction Of Arrival*) $\theta_1 = 15^\circ$ (sinal desejado) e $\theta_2 = -20^\circ$ (interferência); nas 5000 iterações subsequentes, são adicionados mais dois sinais de interferência com DOA $\theta_3 = -50^\circ$ e $\theta_4 = 30^\circ$. O fator de esquecimento é escolhido como $\lambda_f = 0.99$ e o SGD-CMA é implementado com um passo μ_k variante da mesma forma que em Miguez e Castedo [26]. A figura 3.11 mostra as propriedades de convergência e rastreamento dos algoritmos. Pode-se observar que os métodos CGCLF-CMA, NLS-D-CMA e SDCLF-CMA propostos nesta tese apresentam propriedades de convergência e rastreamento comparáveis ao método RLS-CMA de Chen et al. [70] que, por sua vez, apresenta um desempenho melhor que o SGD-CMA. Todos os algoritmos rastreiam rapidamente as mudanças nas fontes de sinais, no entanto o SGD-CMA apresenta um valor menor de SINR em regime. Nota-se também que o método RLS-CMA, baseado na função custo aproximada (3.12), apresenta um transitório mais lento. Isso é devido ao fato de que no início do processo iterativos o produto entre o fator de atenuação λ_f^{k-1} e a diferença entre os produtos internos $\mathbf{x}_i^H \mathbf{w}_k$ e $\mathbf{x}_i^H \mathbf{w}_{i-1}$ (base da função custo (3.12)) não é desprezível. As figuras 3.12 e 3.13 mostram os ganhos normalizados do espectro do sinal reconstituído (“*beam pattern*”) pelo *beamformer* nas iterações 5000 e 10000, pode-se notar

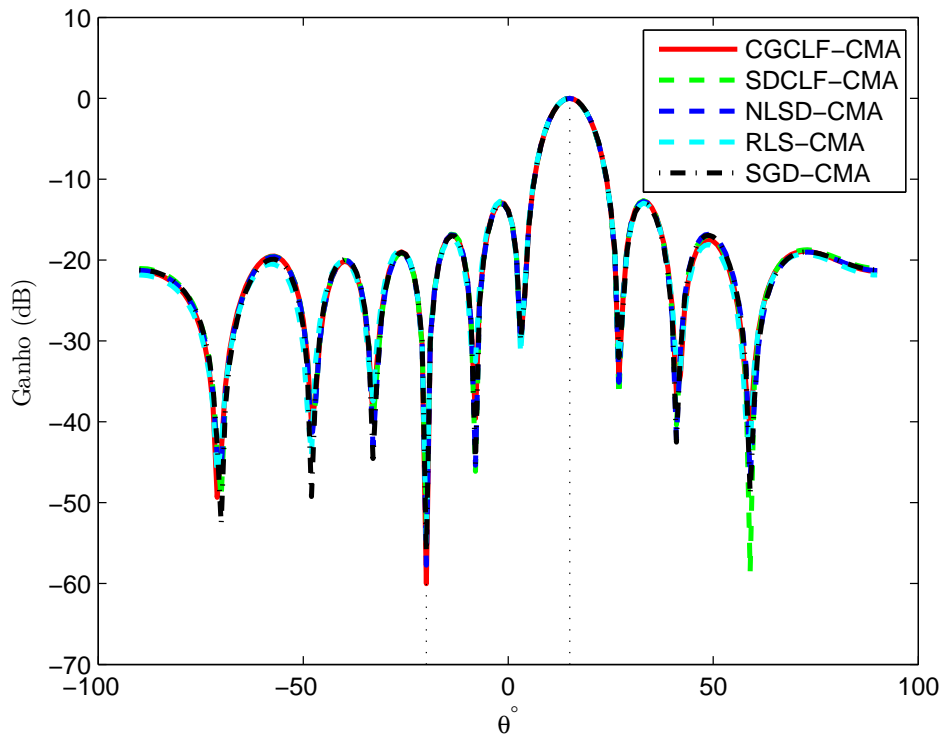


Figura 3.12: *Beampattern* normalizado dos métodos CGCLF-CMA, RLS-CMA, SGD-CMA e NLSD-CMA obtido na iteração 5000.

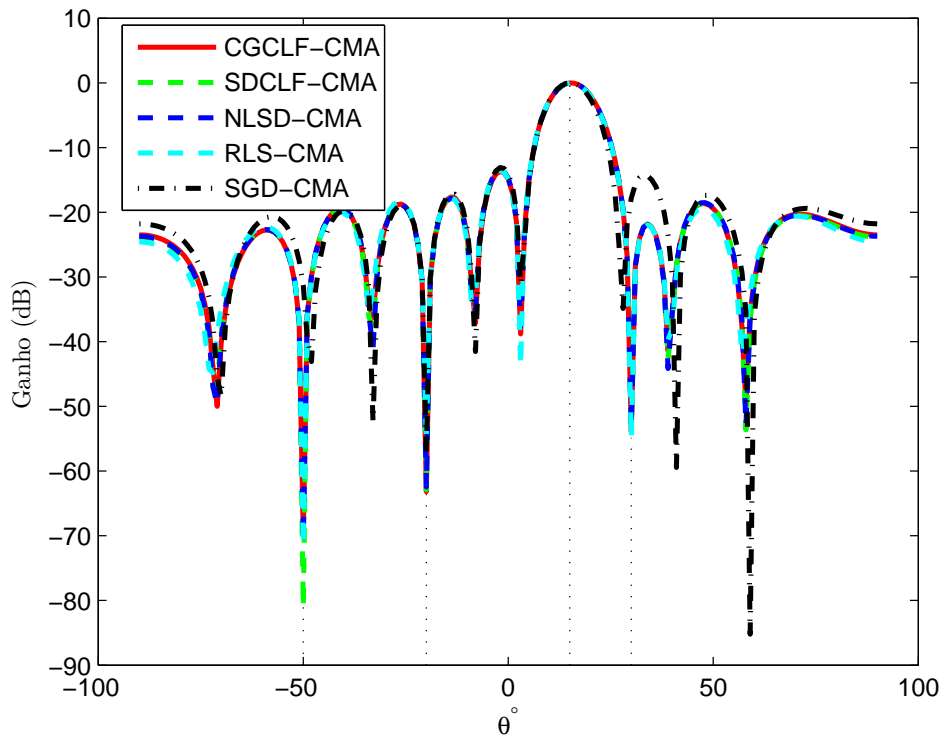


Figura 3.13: *Beampattern* normalizado dos métodos CGCLF-CMA, RLS-CMA, SGD-CMA e NLSD-CMA obtido na iteração 10000.

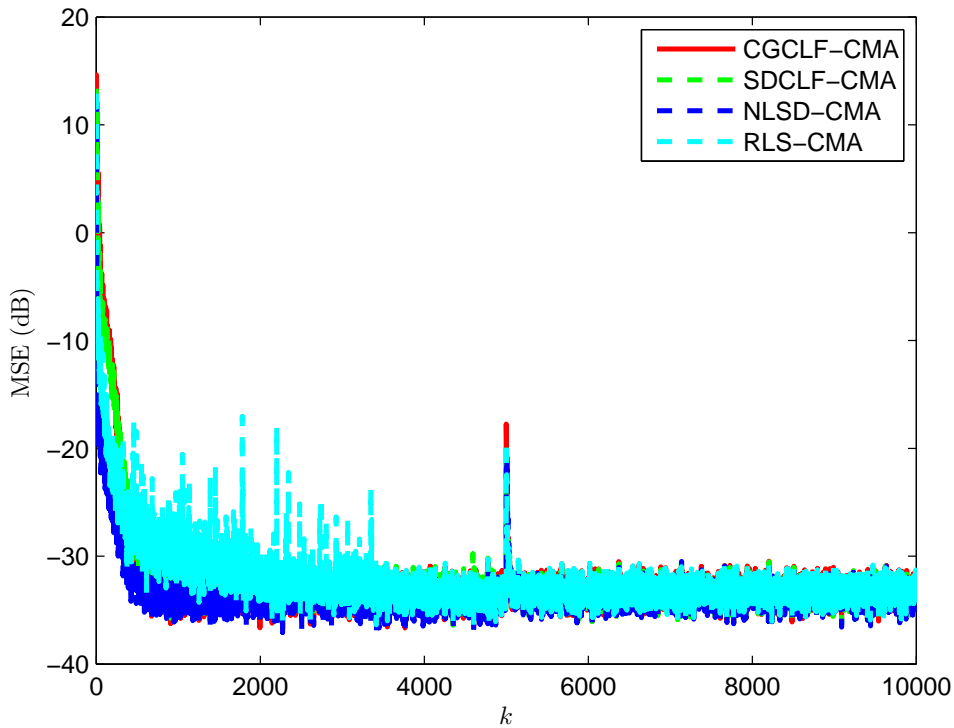


Figura 3.14: Erro médio quadrático dos métodos CGCLF-CMA, RLS-CMA, SDCLF-CMA e NLSD-CMA.

que o método SGD-CMA atenua menos as interferências do que os outros métodos.

Para validar a solução dada pela função custo aproximada (3.16), comparam-se os resultados dos métodos NLSD-CMA e SDCLF-CMA com os dos métodos CGCLF-CMA e RLS-CMA. Nota-se que os métodos NLSD-CMA e SDCLF-CMA atingem em regime um valor de SINR levemente maior que o CGCLF-CMA e o RLS-CMA, conforme os resultados mostrados na figura 3.11. Além disso, a figura 3.14 mostra que todos os quatro algoritmos convergem para o mesmo valor de erro médio quadrático (2.3) em regime, sugerindo que todos os algoritmos convergem para a mesma solução de (3.1). Isso valida a função custo aproximada (3.12). Por outro lado, nota-se novamente que o método RLS-CMA apresenta um transiente inicial mais lento que os outros três métodos, conforme mostrado na figura 3.15 que apresenta uma versão filtrada do erro médio quadrático da figura 3.14.

A convergência do NLSD-CMA é muito sensível à escolha do passo fixo μ de (3.29), porém o NLSD-CMA apresenta $8M$ operações de ponto flutuante (FLOP) a menos que o CGCLF-CMA e $M^2 - M + 2$ FLOP a menos que o RLS-CMA de Chen et al. [70].

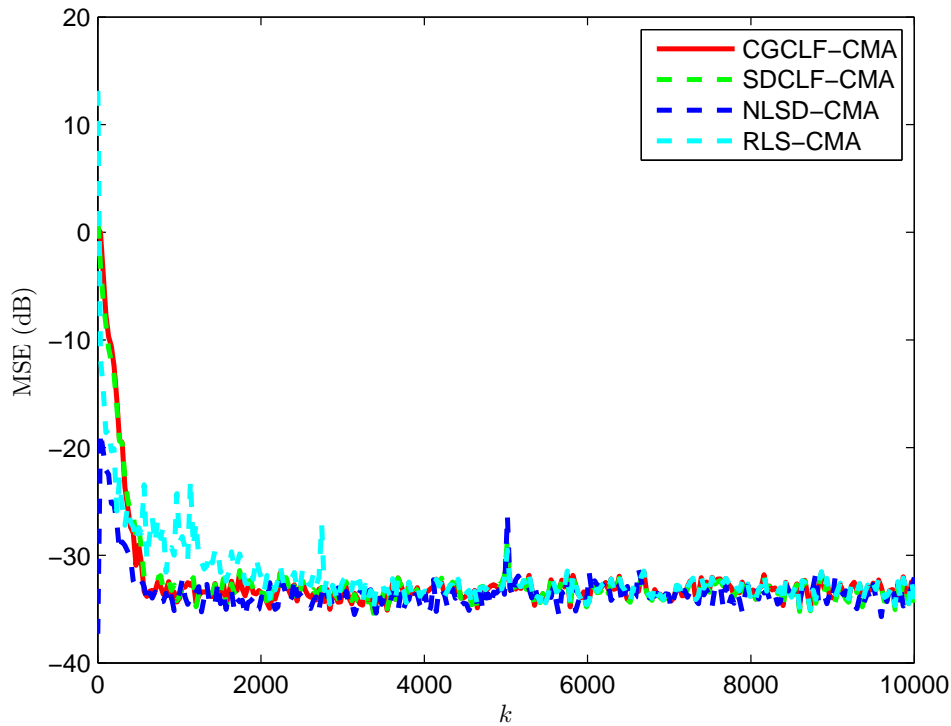


Figura 3.15: Versão filtrada do erro médio quadrático dos métodos CGCLF-CMA, RLS-CMA, SDCLF-CMA e NLSD-CMA.

3.6.2 Sistema de comunicação móvel DS-CDMA

Para testar as propriedades de convergência dos algoritmos de filtragem adaptativa cega com restrições lineares considera-se novamente o sistema de comunicação móvel DS-CDMA da seção 2.6.4. O experimento da seção 2.6.4 é realizado novamente sem fornecer o sinal desejado aos métodos. Para simular as condições de sinais com módulo constante normaliza-se o vetor de entrada $\mathbf{x}_k = \mathbf{S}\mathbf{A}\mathbf{c}_k + \mathbf{n}_k$ de forma que tenha módulo unitário. As figuras 3.16 e 3.17 apresentam as curvas de aprendizado dos métodos CCGCLF-CMA, CSDCLF-CMA, CRLS-CMA e CSGD-CMA para os casos de sistemas com 3 e 5 usuários respectivamente. Novamente a escolha do parâmetro fixo μ do método CSGD-CMA é sensível e pode-se notar que o erro médio quadrático da curva de aprendizado do método CSGD-CMA é quase estacionário em relação ao erro médio quadrático dos métodos CCGCLF-CMA, CSDCLF-CMA e CRLS-CMA. Os novos métodos propostos nesta tese apresentam uma taxa de convergência ligeiramente maior que o CRLS-CMA e também convergem para um erro médio quadrático menor em regime.

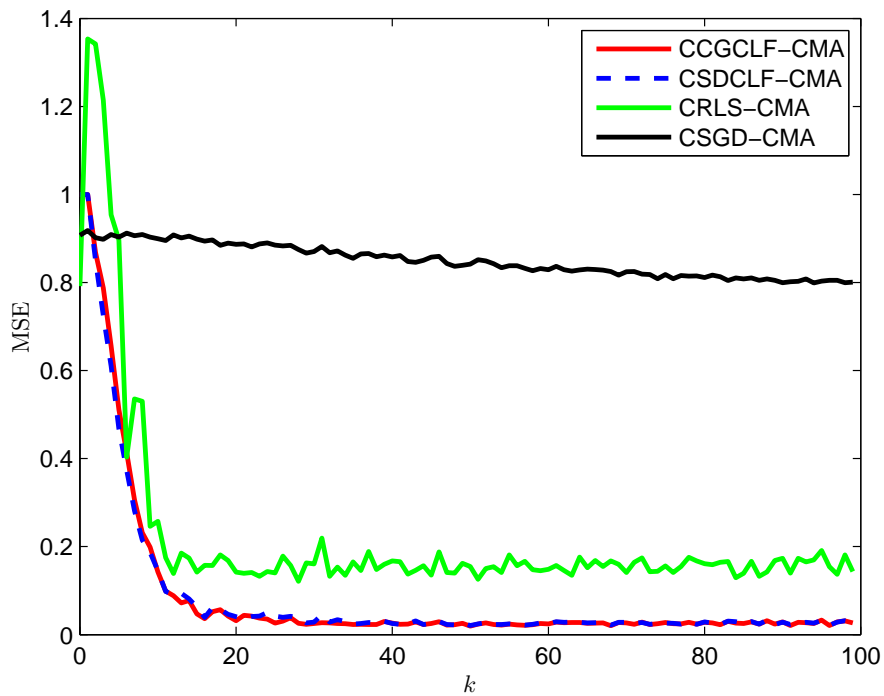


Figura 3.16: Curvas de aprendizado dos filtros adaptativos cegos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 3 usuários.

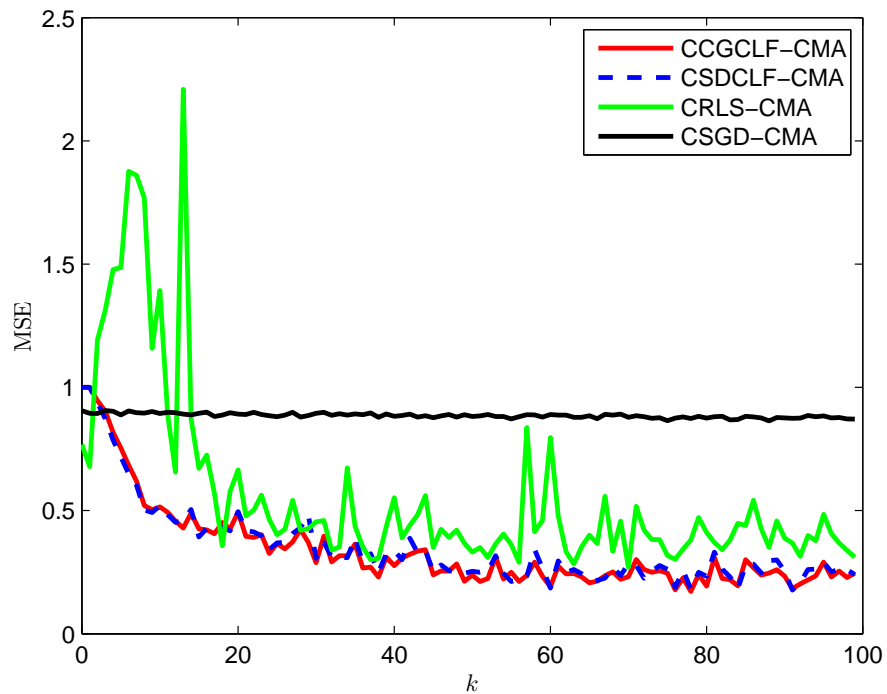


Figura 3.17: Curvas de aprendizado dos filtros adaptativos cegos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 5 usuários.

Método	FLOP
SGD-CMA	$4M + 3$
RLS-CMA	$7M^2 + 9M + 1$
SDCLF-CMA	$6M^2 + 14M - 2$
CGCLF-CMA	$6M^2 + 18M - 1$
CSGD-CMA	$2M^2 + 4M + 3$
CRLS-CMA	$4M^3 + 27M^2 - 2M$
CSDCLF-CMA	$10M^2 + 13M - 2$
CCG-CLF	$10M^2 + 17M - 1$

Tabela 3.2: Comparação do custo computacional dos métodos de filtragem adaptativa cega com ou sem restrições lineares.

3.7 Contribuições e Publicações

Neste capítulo, em analogia ao caso dos problemas de filtragem adaptativa com ou sem restrições lineares, estenderam-se os métodos SD e CG para ambos os casos de sistemas lineares variantes no tempo com ou sem restrições lineares decorrentes dos problemas de filtragem adaptativa cega com ou sem restrições lineares. Os métodos de resolução dos problemas de filtragem adaptativa cega com ou sem restrições lineares foram analisados sob a perspectiva de sistemas dinâmicos realimentados. Novamente, por analogia aos casos da filtragem adaptativa com ou sem restrições lineares, foram propostas duas novas famílias de métodos iterativos para cada um dos casos de filtragem cega. Formulando os problemas em questão como sistemas dinâmicos, provou-se a convergência e a otimalidade dos parâmetros dos novos métodos propostos nos corolários 2.2, 1.2, 2.3 e 1.3, utilizando a estrutura linear do filtro transversal variante no tempo da figura 3.1 e a estrutura bilinear dos sistemas dinâmicos (3.26), (3.34), (3.59) e (3.66) respectivamente. Os métodos propostos CGCLF-CMA e CCGCLF-CMA foram desenvolvidos a partir das funções custo aproximadas (3.12) [70] e (3.42) [81]; as simulações numéricas validam essas funções aproximadas, mostrando que o erro médio quadrático dos métodos supracitados converge para o mesmo valor de regime dos outros métodos desenvolvidos a partir do critério do módulo constante (3.1) e sua versão restrita (3.36). Os métodos propostos apresentam um desempenho superior aos métodos (C)SGD-CMA e (C)RLS-CMA (mais comumente utilizados na literatura), além disso, o custo computacional dos mesmos é maior que do (C)SGD-CMA e menor que do (C)RLS-CMA tido como melhor método na literatura, conforme mostrado na tabela 3.2.

Dos resultados e contribuições apresentados neste capítulo, resultaram os seguintes ar-

tigos submetidos para publicação em revista internacional e congresso nacional respectivamente:

1. DIENE, O., BHAYA, A., “Liapunov analysis and design of conjugate gradient constant modulus algorithm applied to blind adaptive array”. *IEEE Transactions on Signal Processing*. Submetido para publicação, 2008.
2. DIENE, O., BHAYA, A., “Algoritmo de Gradiente Conjugado aplicado ao Critério do Módulo Constante: Projeto e Análise através de funções de controle de Liapunov, e aplicações ao processamento não-supervisionada de antenas adaptativas”. *17th Congresso Brasileiro de Automática*, Juiz de Fora, MG, Brasil, Aceito para publicação, 2008.

Capítulo 4

Análise e projeto de algoritmos de treinamento de perceptron

4.1 Introdução

O problema de separação/classificação de dados, baseado nos atributos, ocorre com uma certa frequência em áreas tais como a mineração de dados e identificação de dados biométricos. Neste contexto, as redes neurais artificiais têm sido bastante utilizadas. O perceptron é o primeiro e o mais básico modelo de rede neural artificial, portanto é a base de muitas aplicações de redes neurais. Embora sejam limitados nas suas aplicações, a utilização dos perceptrons continua sendo importante na classificação de problemas linearmente separáveis devido à sua simplicidade inerente [36]. Treinar um perceptron consiste em resolver um conjunto de desigualdades lineares. Vários algoritmos de treinamento de perceptrons são propostos na literatura [84–88]. O algoritmo LMS (também conhecido como a regra delta ou regra de Widrow-Hoff) [50], a regra do perceptron e suas variações e extensões [89, 90], e o algoritmo Ho-Kashyap e suas versões adaptativas [31, 35] são alguns dos algoritmos mais usados na literatura. Neste capítulo dois dos principais métodos de treinamento de perceptrons são revisados e apresentados de um ponto de vista de controle. Novos métodos de treinamento de perceptrons são propostos e analisados através de funções de Liapunov com controle. Mostra-se através de simulações numéricas que os novos métodos propostos apresentam um desempenho comparável ao dos métodos mais comumente utilizados.

4.2 Formulação do problema

Considere o conjunto de treinamento composto de duas classes $\{w_1, w_2\}$, formadas por m pares $\{\tilde{\mathbf{a}}_k, y_k\}$, $k = 1, 2, \dots, m$, onde $\tilde{\mathbf{a}}_k \in \mathbb{R}^{n+1}$ (sendo que o último componente de $\tilde{\mathbf{a}}_k$ é um valor constante igual a 1) e

$$y_k = \begin{cases} +1 & \text{se } \tilde{\mathbf{a}}_k \in \text{classe } w_1 \\ -1 & \text{se } \tilde{\mathbf{a}}_k \in \text{classe } w_2 \end{cases} \quad (4.1)$$

A classificação correta do conjunto de treinamento descrito anteriormente pode ser feita através do treinamento de um perceptron, determinando um vetor \mathbf{x} de dimensão $n + 1$ que satisfaz as desigualdades lineares abaixo [89, 91, 92]:

$$\tilde{\mathbf{a}}_k^T \mathbf{x} \begin{cases} > 0 & \text{se } y_k = +1 \\ < 0 & \text{se } y_k = -1 \end{cases} \quad \text{para } k = 1, 2, \dots, m. \quad (4.2)$$

Defina o vetor

$$\mathbf{a}_k = \begin{cases} +\tilde{\mathbf{a}}_k & \text{se } y_k = +1 \\ -\tilde{\mathbf{a}}_k & \text{se } y_k = -1 \end{cases} \quad (4.3)$$

e a matriz

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1^T \\ \vdots \\ \mathbf{a}_m^T \end{bmatrix}. \quad (4.4)$$

Portanto, se o conjunto $\{w_1, w_2\}$ é linearmente separável, existe um vetor $\mathbf{b} > \mathbf{0}$, de dimensão m , denominado vetor de margem, tal que [31, 33]

$$\mathbf{A}\mathbf{x} = \mathbf{b} > \mathbf{0}. \quad (4.5)$$

O problema do treinamento de perceptrons pode ser formulado da seguinte forma

$$\text{Encontrar um elemento } (\mathbf{x}, \mathbf{b}) \in \mathcal{S}^+ := \{\mathbf{x} \in \mathbb{R}^n, \mathbf{b} \in \mathbb{R}^m : \mathbf{A}\mathbf{x} = \mathbf{b} > \mathbf{0}\}. \quad (4.6)$$

Na literatura, essa formulação do problema é referido como treinamento em modo batelada devido à necessidade da presença de todos os vetores de treinamento \mathbf{a}_k embutidos na matriz \mathbf{A} .

4.3 Algoritmos de treinamento de perceptrons

4.3.1 Método Ho-Kashyap

Definindo a função custo

$$J(\mathbf{x}, \mathbf{b}) = \|\mathbf{Ax} - \mathbf{b}\|^2, \quad (4.7)$$

o problema de treinamento de perceptrons (4.6) pode ser formulado como um problema equivalente de otimização restrita:

$$\min_{\mathbf{x}, \mathbf{b}} J(\mathbf{x}, \mathbf{b}) \quad \text{sujeito a} \quad \mathbf{b} > \mathbf{0}. \quad (4.8)$$

O problema (4.8) é um problema de otimização convexa com uma restrição de desigualdade na variável \mathbf{b} , e pode ser resolvido através de sistemas do tipo gradiente descendente [57]. Para tanto note que, de (4.7), tem-se

$$\nabla_{\mathbf{x}} J(\mathbf{x}, \mathbf{b}) = \frac{\partial J(\mathbf{x}, \mathbf{b})}{\partial \mathbf{x}} = 2\mathbf{A}^T(\mathbf{Ax} - \mathbf{b}) \quad (4.9)$$

$$\nabla_{\mathbf{b}} J(\mathbf{x}, \mathbf{b}) = \frac{\partial J(\mathbf{x}, \mathbf{b})}{\partial \mathbf{b}} = 2(\mathbf{b} - \mathbf{Ax}). \quad (4.10)$$

A solução $(\mathbf{x}_o, \mathbf{b}_o)$ de (4.8) é tal que

$$\left. \frac{\partial J(\mathbf{x}, \mathbf{b})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_o, \mathbf{b}=\mathbf{b}_o} = \mathbf{0}, \quad \text{e} \quad (4.11)$$

$$\left. \frac{\partial J(\mathbf{x}, \mathbf{b})}{\partial \mathbf{b}} \right|_{\mathbf{x}=\mathbf{x}_o, \mathbf{b}=\mathbf{b}_o} = \mathbf{0}, \quad \text{com } \mathbf{b}_o > \mathbf{0}. \quad (4.12)$$

De (4.11) tem-se que

$$\mathbf{x}_o = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}_o = \mathbf{A}^\dagger \mathbf{b}_o. \quad (4.13)$$

sendo que $(\cdot)^\dagger$ denota a pseudo-inversa de Moore-Penrose [93]. A solução de (4.12) pode ser encontrada iterativamente escolhendo um ponto inicial $\mathbf{b}_0 = \epsilon$, sendo ϵ um vetor de pequenos valores positivos, e determinando uma nova estimativa da seguinte forma

$$\mathbf{b}_{k+1} = \mathbf{b}_k - \frac{\mu_1}{2}(\mathbf{r}_k - |\mathbf{r}_k|), \quad (4.14)$$

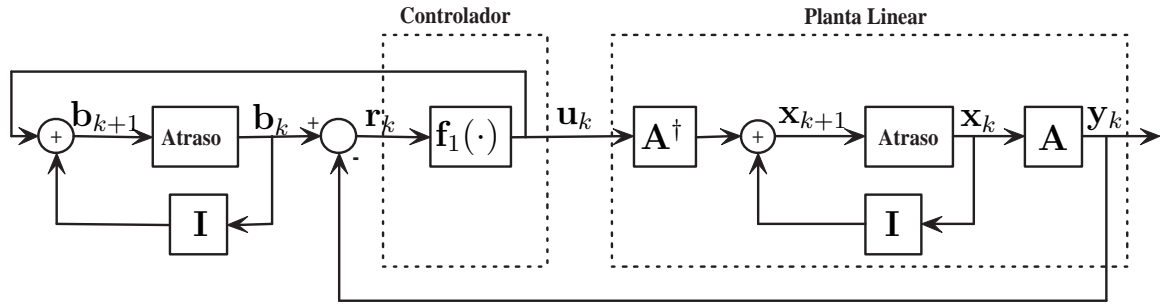


Figura 4.1: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.8) pelo sistema dinâmico (4.16) correspondente ao método Ho-Kashyap, sendo $f_1(\mathbf{r}_k) = \frac{\mu_1}{2}(|\mathbf{r}_k| - \mathbf{r}_k)$. O problema consiste em levar o resíduo \mathbf{r}_k para $\mathbf{0}$ de tal forma que $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}_k > \mathbf{0}$, quando $k \rightarrow \infty$.

sendo μ_1 um pequeno passo positivo e $\mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k$. Note que a nova estimativa \mathbf{b}_{k+1} é mantida sempre positiva pois apenas os componentes negativos do gradiente são utilizados na atualização. De (4.14) e (4.13) pode-se determinar uma estimativa da solução ótima \mathbf{x}_o da seguinte forma:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}^\dagger \mathbf{b}_{k+1} = \mathbf{A}^\dagger \mathbf{b}_k - \frac{\mu_1}{2} \mathbf{A}^\dagger (\mathbf{r}_k - |\mathbf{r}_k|) \\ &= \mathbf{x}_k - \frac{\mu_1}{2} \mathbf{A}^\dagger (\mathbf{r}_k - |\mathbf{r}_k|). \end{aligned} \quad (4.15)$$

As equações (4.14) e (4.15) descrevem o método de Ho e Kashyap [31]. Do ponto de vista de controle esse método pode ser representado por um sistema dinâmico representado em forma de diagrama de blocos na figura 4.1 e descrito pelo seguinte sistema de equações

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{A}^\dagger \mathbf{u}_k \\ \mathbf{b}_{k+1} = \mathbf{b}_k + \mathbf{u}_k \\ \mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k \\ \mathbf{u}_k = \frac{\mu_1}{2} (|\mathbf{r}_k| - \mathbf{r}_k) \end{cases} \quad (4.16)$$

O algoritmo 4.1 apresenta o método Ho-Kashyap na forma do pseudocódigo. A convergência desse método é garantido quando a solução de (4.8) existe (conjunto linearmente separável) [35], nesse caso as iterações do algoritmo 4.1 são encerradas quando todos os componentes do vetor de resíduo \mathbf{r}_k são nulos (ou pequenos o suficiente). No caso de um conjunto não linearmente separável ((4.8) não tem solução) o algoritmo é encerrado quando $\mathbf{r}_k > \mathbf{0}$ (vale lembrar que apenas os componentes negativos de \mathbf{r}_k são utilizados para determinar a direção

Algoritmo 4.1 Método Ho-Kashyap para treinamento de perceptrons em modo batelada

Escolher $\mathbf{b}_0 > \mathbf{0}$

Calcular $\mathbf{x}_0 = \mathbf{A}^\dagger \mathbf{b}_0$, $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{A}\mathbf{x}_0$

Para $k = 0, 1, \dots$, até convergência

$$\mathbf{u}_k = \frac{\mu_1}{2} (|\mathbf{r}_k| - \mathbf{r}_k)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{A}^\dagger \mathbf{u}_k$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \mathbf{u}_k$$

$$\mathbf{r}_{k+1} = \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1}$$

Fim

de descida).

4.3.2 Método Ho-Kashyap adaptativo

O método Ho-Kashyap é de implementação simples porém, o cálculo da pseudo-inversa de \mathbf{A} pode ser de alto custo computacional e/ou requerer um tratamento especial quando a matriz $\mathbf{A}^T \mathbf{A}$ é mal condicionada. Ao invés de determinar explicitamente a pseudo-inversa, pode-se utilizar a direção de descida do gradiente de $J(\mathbf{x}, \mathbf{b})$ em relação a \mathbf{x} na determinação da nova estimativa da solução ótima \mathbf{x}_0 dada por (4.15). Nesse caso a nova estimativa \mathbf{x}_{k+1} da solução de (4.11) é dada pela seguinte equação

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mu_2 \nabla_{\mathbf{x}} J(\mathbf{x}, \mathbf{b})|_{\mathbf{x}_k, \mathbf{b}_{k+1}}. \quad (4.17)$$

Inserindo (4.9) e (4.14) em (4.17) resulta em

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k - \mu_2 \mathbf{A}^T (\mathbf{A}\mathbf{x}_k - \mathbf{b}_{k+1}) \\ &= \mathbf{x}_k - \mu_2 \mathbf{A}^T \left(\mathbf{A}\mathbf{x}_k - \mathbf{b}_k + \frac{\mu_1}{2} (\mathbf{r}_k - |\mathbf{r}_k|) \right) \\ &= \mathbf{x}_k + \frac{\mu_1 \mu_2}{2} \mathbf{A}^T \left[|\mathbf{r}_k| - \mathbf{r}_k \left(1 - \frac{2}{\mu_1} \right) \right] \end{aligned} \quad (4.18)$$

As equações (4.14) e (4.18) descrevem o método Ho-Kashyap adaptativo, proposto em Hassoun e Song [35]. A figura 4.2 e o sistema (4.19) mostram o sistema dinâmico correspondente ao problema de resolução de (4.8) pelo método Ho-Kashyap adaptativo descrito em forma

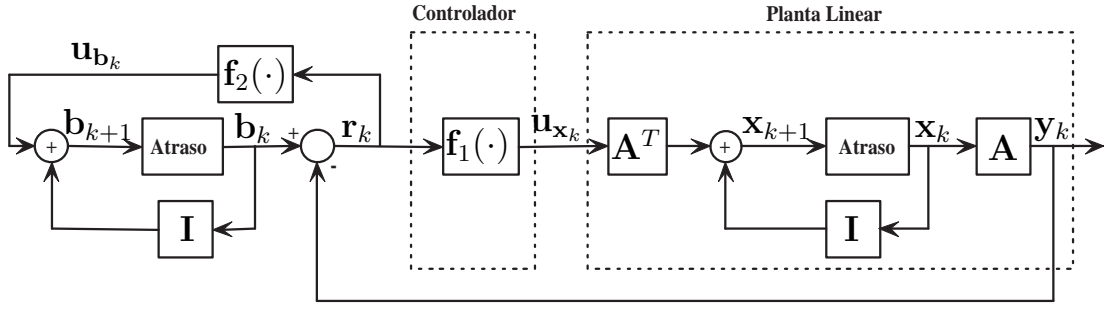


Figura 4.2: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.8) pelo sistema dinâmico (4.19) correspondente ao método Ho-Kashyap adaptativo, sendo $f_1(\mathbf{r}_k) = \frac{\mu_1 \mu_2}{2} \left[|\mathbf{r}_k| - \mathbf{r}_k \left(1 - \frac{2}{\mu_1} \right) \right]$ e $f_2(\mathbf{r}_k) = \frac{\mu_1}{2} (|\mathbf{r}_k| - \mathbf{r}_k)$. O problema consiste em levar o resíduo \mathbf{r}_k para $\mathbf{0}$ de tal forma que $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}_k > \mathbf{0}$, quando $k \rightarrow \infty$.

Algoritmo 4.2 Método Ho-Kashyap adaptativo para treinamento de perceptrons em modo batelada

Escolher $\mathbf{x}_0, \mathbf{b}_0 > \mathbf{0}$

Calcular $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{A}\mathbf{x}_0$

Para $k = 0, 1, \dots$, até convergência

$$\mathbf{u}_{\mathbf{b}_k} = \frac{\mu_1}{2} (|\mathbf{r}_k| - \mathbf{r}_k)$$

$$\mathbf{u}_{\mathbf{x}_k} = \frac{\mu_1 \mu_2}{2} \left[|\mathbf{r}_k| - \mathbf{r}_k \left(1 - \frac{2}{\mu_1} \right) \right]$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{A}^T \mathbf{u}_{\mathbf{x}_k}$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \mathbf{u}_{\mathbf{b}_k}$$

$$\mathbf{r}_{k+1} = \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1}$$

Fim

de pseudocódigo pelo algoritmo 4.2.

$$\left\{ \begin{array}{l} \mathbf{b}_{k+1} = \mathbf{b}_k + \mathbf{u}_{\mathbf{b}_k} \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{A}^T \mathbf{u}_{\mathbf{x}_k} \\ \mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k \\ \mathbf{u}_{\mathbf{b}_k} = \frac{\mu_1}{2} (|\mathbf{r}_k| - \mathbf{r}_k) \\ \mathbf{u}_{\mathbf{x}_k} = \frac{\mu_1 \mu_2}{2} \left[|\mathbf{r}_k| - \mathbf{r}_k \left(1 - \frac{2}{\mu_1} \right) \right] \end{array} \right. \quad (4.19)$$

Vale ressaltar que o método Ho-Kashyap adaptativo apresentado acima (algoritmos 4.2) garante uma convergência robusta para uma superfície separadora se o conjunto de treinamento for linearmente separável, porém não oferece nenhuma garantia de convergência para conjuntos não separáveis linearmente. Além disso, o número de passos necessários para a convergência, no caso linearmente separável, depende do ponto de inicialização [32]. Para determinar um método iterativo com convergência garantida para ambos os casos separável e

não separável, independente do ponto inicial, Nagaraja e Krishna [34], Nagaraja e Bose [36] usaram técnicas de gradiente conjugado para resolver o problema do treinamento de perceptrons (4.6), a partir da resolução da equação (4.5). Para o caso não separável linearmente, o método de gradiente conjugado proposto por Nagaraja e Krishna [34] converge para o par $(\mathbf{x}_k, \mathbf{b}_k > \mathbf{0})$ tal que a norma $\|\mathbf{A}\mathbf{x}_k - \mathbf{b}_k\|$ é mínima, isto corresponde geometricamente em encontrar a superfície separadora ($\mathbf{A}\mathbf{x}_k = \mathbf{b}_k$) tal que a soma das distâncias dos pontos mal classificados à superfície seja mínima. A técnica de gradiente conjugado consiste em substituir a direção de descida do gradiente, utilizada pelo método Ho-Kashyap adaptativo, pela direção conjugada do gradiente tal que:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mu_k \mathbf{P}_{\mathbf{x}_k} \quad (4.20)$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \mu_k \mathbf{P}_{\mathbf{b}_k} \quad (4.21)$$

$$\mathbf{p}_{\mathbf{x}_k} = \theta_k \mathbf{p}_{\mathbf{x}_{k-1}} + 2\mathbf{A}^T \mathbf{r}_k / \|\mathbf{g}_k\|^2 \quad (4.22)$$

$$\mathbf{p}_{\mathbf{b}_k} = \theta_k \mathbf{p}_{\mathbf{b}_{k-1}} + (|\mathbf{r}_k| - \mathbf{r}_k) / \|\mathbf{g}_k\|^2, \quad (4.23)$$

sendo

$$\mu_k = \frac{\mathbf{r}_k^T (\mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k})}{(\mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k})^T (\mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k})} \quad (4.24)$$

$$\theta_k \in \{0, 1\} \quad (4.25)$$

$$\mathbf{g}_k = \begin{bmatrix} 2\mathbf{A}^T \mathbf{r}_k \\ |\mathbf{r}_k| - \mathbf{r}_k \end{bmatrix}. \quad (4.26)$$

Note que a escolha do parâmetro θ_k é uma heurística baseada em técnicas de reinicialização do método de gradiente conjugado [94]. Nas duas próximas seções desenvolver-se-ão novos algoritmos para o treinamento de perceptrons em modo batelada sem o uso de nenhuma heurística e com prova sistemática de convergência. Os novos algoritmos são obtidos através da combinação de técnicas de gradiente conjugado e declividade máxima com o método Ho-Kashyap, e da extensão da mesma técnica de análise proposta em [9, 10], e utilizada nos capítulos 2 e 3, para o problema de treinamento de perceptron.

4.3.3 Método SDP-CLF (*Steepest Descent Perceptron*)

Combinando o método da declividade máxima com a técnica de Ho-Kashyap [35], as equações (4.14) e (4.18) podem ser reescritas da seguinte forma

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{r}_k \quad (4.27)$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k), \quad (4.28)$$

sendo α_k um parâmetro a ser determinado de forma a levar o resíduo \mathbf{r}_k a zero em um número finito de iterações. De (4.27) e (4.28) determina-se a lei de atualização do resíduo da seguinte forma

$$\begin{aligned} \mathbf{r}_{k+1} &= \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1} \\ &= \mathbf{b}_k + \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k) - \mathbf{A}(\mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{r}_k) \\ &= \mathbf{r}_k - \alpha_k ((\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|) \\ &= \mathbf{r}_k - \alpha_k \mathbf{r}'_k, \end{aligned} \quad (4.29)$$

sendo $\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|$. As equações (4.27-4.29) descrevem um sistema dinâmico (4.30), representado em forma de diagrama de blocos na figura 4.3, correspondente à resolução de (4.6) pelo método da declividade máxima

$$\left\{ \begin{array}{l} \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{A}^T \mathbf{u}_{\mathbf{x}_k} \\ \mathbf{b}_{k+1} = \mathbf{b}_k + \mathbf{u}_{\mathbf{b}_k} \\ \mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k \\ \mathbf{u}_{\mathbf{x}_k} = \alpha_k \mathbf{r}_k \\ \mathbf{u}_{\mathbf{b}_k} = \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k) \end{array} \right. \quad (4.30)$$

Para o sistema (4.30) outro resultado desta tese pode ser enunciado da seguinte forma [37]:

Teorema 3. *Suponha que*

$$\forall \mathbf{v} > \mathbf{0}, \mathbf{A}^T \mathbf{v} \neq \mathbf{0}. \quad (4.31)$$

Defina

$$\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|. \quad (4.32)$$

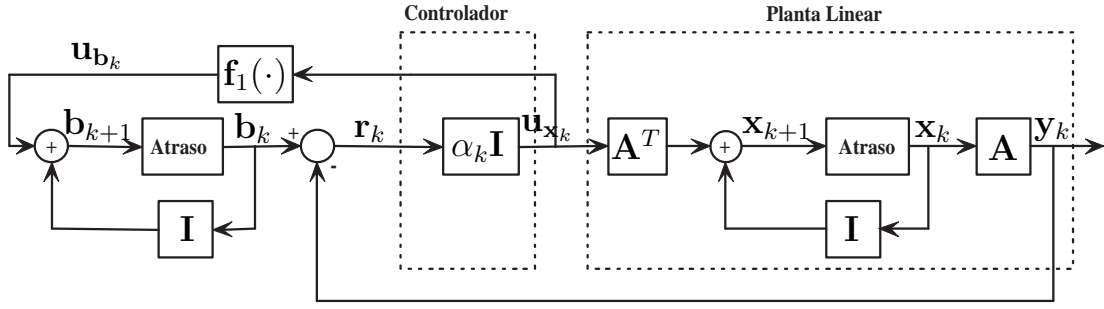


Figura 4.3: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.6) pelo sistema dinâmico (4.30) correspondente ao método SDP-CLF, sendo $\mathbf{f}_1(\mathbf{r}_k) = \alpha_k(|\mathbf{r}_k| - \mathbf{r}_k)$. O problema consiste determinar o parâmetro α_k tal que $\mathbf{r}_k \rightarrow \mathbf{0}$, i.e. $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}_k > \mathbf{0}$, quando $k \rightarrow \infty$.

A escolha

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} \quad (4.33)$$

é ótima para a função de Liapunov escolhida e garante a estabilidade assintótica do equilíbrio do sistema (4.30), i.e.: o equilíbrio de (4.30) tende para o conjunto $\mathcal{S}^+ := \{\mathbf{x} \in \mathbb{R}^{n+1}, \mathbf{b} \in \mathbb{R}^m : \mathbf{A}\mathbf{x} = \mathbf{b} > \mathbf{0}\}$.

Prova do Teorema 3. Escolha a função de Liapunov com controle

$$\mathbf{V}_r(k) = \langle \mathbf{r}_k, \mathbf{r}_k \rangle. \quad (4.34)$$

Então

$$\begin{aligned} \Delta \mathbf{V}_r &= \mathbf{V}_r(k+1) - \mathbf{V}_r(k) = \langle \mathbf{r}_{k+1}, \mathbf{r}_{k+1} \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k - \alpha_k \mathbf{r}'_k, \mathbf{r}_k - \alpha_k \mathbf{r}'_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k, \mathbf{r}_k \rangle - 2\alpha_k \langle \mathbf{r}_k, \mathbf{r}'_k \rangle + \alpha_k^2 \langle \mathbf{r}'_k, \mathbf{r}'_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= -2\alpha_k \langle \mathbf{r}_k, \mathbf{r}'_k \rangle + \alpha_k^2 \langle \mathbf{r}'_k, \mathbf{r}'_k \rangle. \end{aligned} \quad (4.35)$$

O valor ótimo α_k (que torna $\Delta \mathbf{V}_r$ o mais negativo possível), dado por (4.33), é determinado através da resolução da seguinte equação:

$$\frac{\partial \Delta \mathbf{V}_r}{\partial \alpha_k} = -2\langle \mathbf{r}_k, \mathbf{r}'_k \rangle + 2\alpha_k \langle \mathbf{r}'_k, \mathbf{r}'_k \rangle = 0.$$

Algoritmo 4.3 Método SDP-CLF para treinamento de perceptrons em modo batelada

Escolher $\mathbf{x}_0, \mathbf{b}_0 > \mathbf{0}$

Calcular $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{A}\mathbf{x}_0$

Para $k = 0, 1, \dots$, até convergência

$$\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|$$

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{r}_k$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k)$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{r}'_k$$

Fim

Substituindo (4.33) em (4.35) resulta

$$\Delta \mathbf{V}_r = - \frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle^2}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} \leq \mathbf{0}. \quad (4.36)$$

De (4.32) pode-se escrever

$$\langle \mathbf{r}_k, \mathbf{r}'_k \rangle = \mathbf{r}_k^T \mathbf{A} \mathbf{A}^T \mathbf{r}_k + \mathbf{r}_k^T \mathbf{r}_k - \mathbf{r}_k^T |\mathbf{r}_k| \quad (4.37)$$

No lado direito da equação (4.37), o primeiro termo é positivo, a menos que $\mathbf{A}^T \mathbf{r}_k = \mathbf{0}$ e a soma do segundo e do terceiro termo é sempre positiva, a menos que $\mathbf{r}_k > \mathbf{0}$. Desse fato pode-se concluir que

$$\langle \mathbf{r}_k, \mathbf{r}'_k \rangle \geq 0 \quad (4.38)$$

e é igual a zero se e somente se $\mathbf{r}_k > \mathbf{0}$ e $\mathbf{A}^T \mathbf{r}_k = \mathbf{0}$. Portanto, de (4.31)

$$\Delta \mathbf{V}_r = - \frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle^2}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} < \mathbf{0}, \quad (4.39)$$

o que implica que $\|\mathbf{r}_k\|$ é uma seqüência decrescente.

Para provar a positividade de \mathbf{b}_k , note que $|\mathbf{r}_k| - \mathbf{r}_k \geq \mathbf{0}$ e, de (4.33) e (4.38), $\alpha_k \geq 0$. Portanto se $\mathbf{b}_0 > \mathbf{0}$, então de (4.28) $\mathbf{b}_k > \mathbf{0}$ para todo k . ■

O algoritmo 4.3 apresenta o pseudocódigo da extensão do método SD para o treinamento de perceptrons em modo batelada, baseada no teorema 3 e referenciado nesta tese como SDP-CLF.

4.3.4 Método CGP-CLF (*Conjugate Gradient Perceptron*)

Outro novo método pode ser desenvolvido combinando a técnica de gradiente conjugado com a técnica de Ho-Kashyap [35], para aproveitar a convergência rápida de algoritmos do tipo CG para igualdades. Nesse caso as equações (4.14) e (4.18) são reescritas da seguinte forma

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} \quad (4.40)$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k}, \quad (4.41)$$

sendo

$$\mathbf{p}_{\mathbf{x}_k} = \beta_k \mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{r}_k, \quad (4.42)$$

$$\mathbf{p}_{\mathbf{b}_k} = \beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + |\mathbf{r}_k| - \mathbf{r}_k, \quad (4.43)$$

α_k e β_k os parâmetros a serem determinados de forma a levar o resíduo \mathbf{r}_k a zero, garantindo assim a convergência do método. Das equações (4.40-4.41) pode-se deduzir a lei de atualização do resíduo da seguinte forma

$$\begin{aligned} \mathbf{r}_{k+1} &= \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k} - \mathbf{A}(\mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{p}_{\mathbf{x}_k}) \\ &= \mathbf{r}_k - \alpha_k (\mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}) \\ &= \mathbf{r}_k - \alpha_k \mathbf{p}_k, \end{aligned} \quad (4.44)$$

sendo $\mathbf{p}_k = \mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}$. Dessa forma o problema de rastreamento de sinal (2.13), aplicado ao caso do problema do treinamento de perceptrons em modo batelada (4.6), pode ser reescrito como

$$\left\{ \begin{array}{l} \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} \\ \mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k} \\ \mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{p}_k \\ \mathbf{p}_{\mathbf{x}_k} = \beta_k \mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{r}_k \\ \mathbf{p}_{\mathbf{b}_k} = \beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + |\mathbf{r}_k| - \mathbf{r}_k \\ \mathbf{p}_k = \mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}. \end{array} \right. \quad (4.45)$$

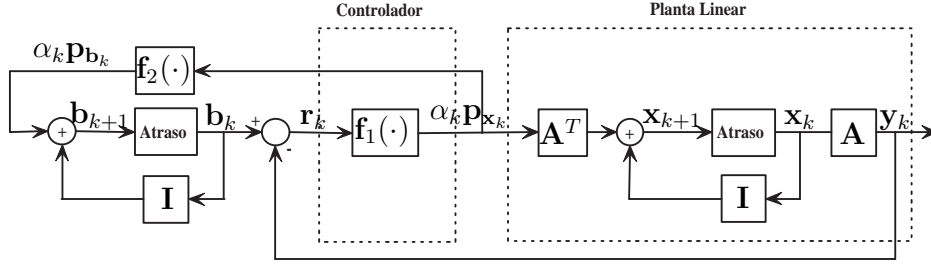


Figura 4.4: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.6) pelo sistema dinâmico (4.45) correspondente ao método CGP-CLF, sendo $\mathbf{f}_1(\mathbf{r}_k) = \alpha_k(\beta_k \mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{r}_k)$ e $\mathbf{f}_2(\mathbf{r}_k) = \alpha_k(\beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + |\mathbf{r}_k| - \mathbf{r}_k)$. O problema consiste determinar os parâmetros α_k e β_k tal que $\mathbf{r}_k \rightarrow \mathbf{0}$, i.e. $\mathbf{A}\mathbf{x}_k \rightarrow \mathbf{b}_k > \mathbf{0}$, quando $k \rightarrow \infty$.

A figura 4.4 apresenta o sistema (4.45) em forma de diagrama de blocos. Uma condição suficiente para a estabilidade assintótica do equilíbrio do sistema (4.45) é dada pelo seguinte teorema [37]:

Teorema 4. *Com a hipótese (4.31), defina*

$$\mathbf{p}_k = \mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}. \quad (4.46)$$

As escolhas

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \quad (4.47)$$

$$\beta_{k+1} = -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \quad (4.48)$$

são ótimas para as funções de Liapunov escolhidas e garantem que $\mathbf{r}_k \rightarrow \mathbf{0}$ e $\mathbf{p}_k \rightarrow \mathbf{0}$, i.e. o equilíbrio do sistema (4.45) tende para o conjunto $\mathcal{S} := \{\mathbf{x} \in \mathbb{R}^n, \mathbf{b} \in \mathbb{R}^m : \mathbf{A}\mathbf{x} = \mathbf{b}\}$.

Prova do Teorema 4. *Os parâmetros α_k e β_k são considerados como variáveis de controle e podem então ser determinados por uma escolha apropriada de CLF. Note que*

$$\mathbf{r}_{k+1} = \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{r}_k - \alpha_k(\mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}). \quad (4.49)$$

Defina

$$\begin{aligned}
\mathbf{p}_k &= \mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k} \\
&= \beta_k \mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{A}\mathbf{A}^T \mathbf{r}_k - \beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + \mathbf{r}_k - |\mathbf{r}_k| \\
&= \beta_k (\mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_{k-1}} - \mathbf{p}_{\mathbf{b}_{k-1}}) + (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k| \\
&= \beta_k \mathbf{p}_{k-1} + \mathbf{r}'_k,
\end{aligned}$$

sendo

$$\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|.$$

Então o problema de rastreamento de sinal pode ser reescrito nas coordenadas \mathbf{r}_k e \mathbf{p}_k conforme segue

$$\begin{cases} \mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{p}_k \\ \mathbf{p}_{k+1} = \beta_{k+1} \mathbf{p}_k + \mathbf{r}'_{k+1} \end{cases} \quad (4.50)$$

O objetivo agora é determinar os parâmetros α_k e β_k que tornam o equilíbrio $(\mathbf{0}, \mathbf{0})$ do sistema (4.50) assintoticamente estável. No equilíbrio de (4.50) tem-se que $\mathbf{A}\mathbf{x}_k = \mathbf{b}_k$ e $\mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} = \mathbf{p}_{\mathbf{b}_k}$, então $\mathbf{A}\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \alpha_k \mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k} = \mathbf{b}_{k+1}$. Isso significa que, no equilíbrio de (4.50), o equilíbrio de (4.45) pertence ao conjunto \mathcal{S} . Para determinar α_k , escolha novamente a CLF $V_{\mathbf{r}}(k)$ dada por (4.34), então

$$\begin{aligned}
\Delta V_{\mathbf{r}} &= V_{\mathbf{r}}(k+1) - V_{\mathbf{r}}(k) \\
&= \langle \mathbf{r}_{k+1}, \mathbf{r}_{k+1} \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\
&= \langle \mathbf{r}_k - \alpha_k \mathbf{p}_k, \mathbf{r}_k - \alpha_k \mathbf{p}_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\
&= \langle \mathbf{r}_k, \mathbf{r}_k \rangle - 2\alpha_k \langle \mathbf{r}_k, \mathbf{p}_k \rangle + \alpha_k^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\
&= -2\alpha_k \langle \mathbf{r}_k, \mathbf{p}_k \rangle + \alpha_k^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle.
\end{aligned} \quad (4.51)$$

O valor de α_k , dado por (4.47), é determinado através da resolução da equação $\frac{\partial \Delta V_{\mathbf{r}}}{\partial \alpha_k} = 0$.

Desde que $\mathbf{p}_k^T \mathbf{r}_k \neq \mathbf{0}$, substituindo (4.47) em (4.51) resulta

$$\Delta V_{\mathbf{r}} = -\frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} < \mathbf{0}. \quad (4.52)$$

Note que, de (4.50), (4.47), (4.31), (4.37) e (4.38)

$$\begin{aligned}\langle \mathbf{r}_k, \mathbf{p}_k \rangle &= \beta_k \langle \mathbf{r}_k, \mathbf{p}_{k-1} \rangle + \langle \mathbf{r}_k, \mathbf{r}'_k \rangle = \langle \mathbf{r}_k, \mathbf{r}'_k \rangle \\ &= \mathbf{r}_k^T \mathbf{A} \mathbf{A}^T \mathbf{r}_k + \mathbf{r}_k^T \mathbf{r}_k - \mathbf{r}_k^T |\mathbf{r}_k| > \mathbf{0}.\end{aligned}\quad (4.53)$$

De (4.52), pode-se concluir que a norma-2 do resíduo \mathbf{r}_k decresce monotonicamente. \square

O próximo passo consiste em determinar o parâmetro β_k , e para tanto defina

$$\begin{aligned}\mathbf{V}_{\mathbf{p}_{k+1}} &= \langle \mathbf{p}_{k+1}, \mathbf{p}_{k+1} \rangle \\ &= \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle + 2\beta_{k+1} \langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle + \beta_{k+1}^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle.\end{aligned}\quad (4.54)$$

Então pode-se escrever que

$$\frac{\partial \Delta \mathbf{V}_{\mathbf{p}_{k+1}}}{\partial \beta_{k+1}} = \mathbf{0} \implies \beta_{k+1} = -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle}\quad (4.55)$$

Substituindo (4.55) em (4.54) resulta

$$\mathbf{V}_{\mathbf{p}_{k+1}} = \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle - \frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \leq \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle\quad (4.56)$$

De (4.52) e (4.32) pode-se concluir que \mathbf{r}'_{k+1} decresce na norma-2, porém não necessariamente monotonicamente. Portanto (4.56) implica que \mathbf{p}_{k+1} decresce na norma-2, embora de forma não necessariamente monotônica, concluindo a prova. \blacksquare

O algoritmo 4.4 abaixo apresenta o pseudo-código do método CG para treinamento de perceptron em modo batelada, determinado através da técnica de CLF [10] e referenciado nesta tese como CGP-CLF. Vale ressaltar que o algoritmo 4.4 não utiliza nenhuma heurística.

Observação 4. Note que o teorema 4 garante que o algoritmo 4.4 convergirá para uma solução de $\mathbf{A} \mathbf{x}_k = \mathbf{b}_k$, mas não garante que $\mathbf{b}_k > \mathbf{0}$. Por outro lado, os resultados experimentais mostrados na seção 4.4 mostram que o algoritmo 4.4 converge para uma solução \mathbf{x}_k tal que $\mathbf{A} \mathbf{x}_k = \mathbf{b}_k > \mathbf{0}$, e a prova rigorosa da ocorrência da convergência para o conjunto \mathcal{S}^+ , ao invés de \mathcal{S} , é um dos tópicos considerados como trabalhos futuros.

Algoritmo 4.4 Método CGP-CLF para treinamento de Perceptron em modo batelada

Escolher $\mathbf{x}_0, \mathbf{b}_0 > \mathbf{0}$

Calcular $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{A}\mathbf{x}_0, \mathbf{p}_{\mathbf{x}_0} = \mathbf{r}_0, \mathbf{p}_{\mathbf{b}_0} = |\mathbf{r}_0| - \mathbf{r}_0, \mathbf{p}_0 = \mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_0} - \mathbf{p}_{\mathbf{b}_0}$

Para $k = 0, 1, \dots$, até convergência

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{p}_{\mathbf{x}_k}$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k}$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{p}_k$$

$$\mathbf{r}'_{k+1} = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_{k+1} - |\mathbf{r}_{k+1}|$$

$$\beta_{k+1} = -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle}$$

$$\mathbf{p}_{\mathbf{x}_{k+1}} = \beta_{k+1} \mathbf{p}_{\mathbf{x}_k} + \mathbf{r}_{k+1}$$

$$\mathbf{p}_{\mathbf{b}_{k+1}} = \beta_{k+1} \mathbf{p}_{\mathbf{b}_k} + |\mathbf{r}_{k+1}| - \mathbf{r}_{k+1}$$

$$\mathbf{p}_{k+1} = \mathbf{A}\mathbf{A}^T \mathbf{p}_{\mathbf{x}_{k+1}} - \mathbf{p}_{\mathbf{b}_{k+1}}$$

Fim

4.4 Simulações Numéricas: Reconhecimento de padrão

Para verificar o desempenho dos novos métodos propostos SDP-CLF e CGP-CLF (algoritmos 4.3 e 4.4), comparam-se esses com os métodos Ho-Kashyap (HK – algoritmo 4.1), Ho-Kashyap adaptativo (AHK – algoritmo 4.2) e o método de gradiente conjugado (CGA) proposto em Nagaraja e Krishna [34]. A experiência é feita para vários conjuntos linearmente separáveis e não separáveis. Os conjuntos escolhidos são da base de dados da planta iris da Universidade de California Irvine UCI [95]. Essa base de dados é a mais conhecida e utilizada na literatura de reconhecimento de padrão. O conjunto de dados contém três classes de cinquenta instâncias cada, sendo que cada classe refere-se a um tipo de planta de iris. A base de dados consiste em quatro atributos e a classe. Os atributos são comprimento de sepal, largura de sepal, comprimento de pétala e largura de pétala. As três classes correspondentes à planta iris são *iris setosa*, *iris versicolor* e *iris virginica*. A figura 4.5 apresenta o gráfico da largura de petal em função do comprimento de petal. Sabe-se que a *iris setosa* é linearmente separável das outras duas classes, porém a *iris versicolor* e a *iris virginica* são linearmente não separáveis em relação aos quatro atributos. Como esse caso consiste em um problema de três classes, divide-se esse em três problemas de duas classes. No primeiro caso, amostras da *iris setosa* são denominadas de classe 1 enquanto amostras da *iris versicolor* são consideradas como sendo da classe -1. No segundo caso amostras da *iris virginica* são considerados da classe -1 enquanto amostras da *iris setosa* são mantidos como classe 1. No terceiro e último caso, amostras da *iris versicolor* são considerados da classe 1

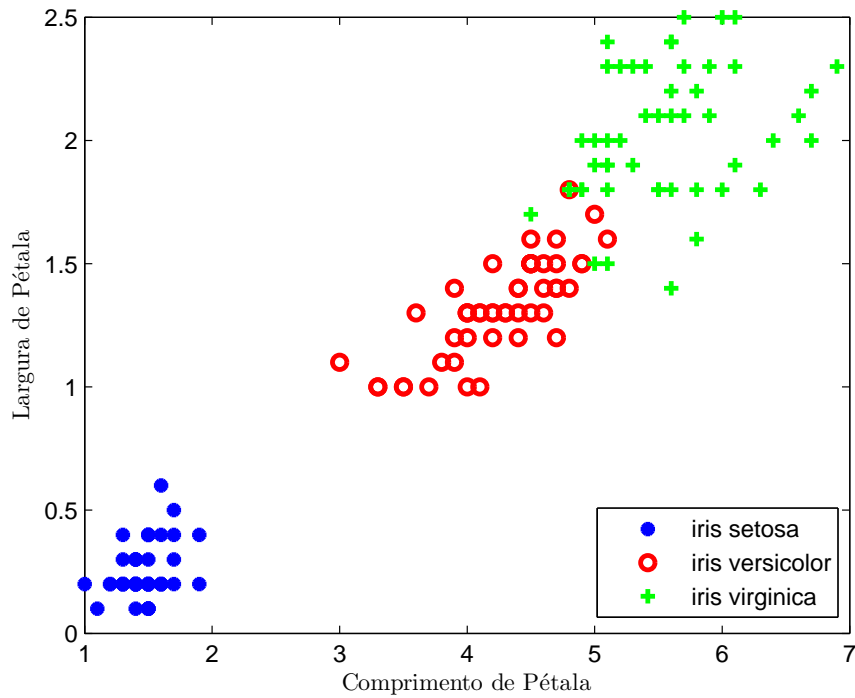


Figura 4.5: Curva dos dados da iris: Largura de pétala em função do Comprimento de pétala

e amostras da *iris virginica* da classe -1.

Os vetores de entrada são aumentados acrescentando a constante +1 que representa o “*bias*”. As amostras das classes negativas, incluindo o “*bias*”, são multiplicados por -1 para que o objetivo seja: encontrar o par de vetores $(\mathbf{x}_k, \mathbf{b}_k)$ tal que $\mathbf{A}\mathbf{x}_k = \mathbf{b}_k > 0$.

As figuras 4.6, 4.7 e 4.8 apresentam os hiperplanos separadores obtido pelos métodos CGP-CLF, SDP-CLF, HK, AHK e CGA [34]. Como o problema de treinamento de perceptron pode também ser resolvido pela teoria de “máquinas de vetor de suporte” SVM, apresentam-se, a título de comparação, os resultados obtidos pela LS-SVM Suykens em Ferreira et al. [96]. Nos dois casos linearmente separáveis, todos os algoritmos classificaram corretamente todos os pontos, porém nem todos alcançaram o critério de parada escolhido tal que $\|\mathbf{r}_k\| \leq 0$. No caso do algoritmo Ho-Kashyap adaptativo, a escolha dos parâmetros μ_1 e μ_2 é muito crítico em relação à convergência. Por isso a escolha $\mu_1 = 1,5$ e $\mu_2 = 1/\lambda_{max}$ (λ_{max} é o maior autovalor de $\mathbf{A}^T \mathbf{A}$) conforme Nagaraja e Bose [36] resulta em um resíduo com norma consideravelmente alta, conforme mostra a tabela 4.1 em qual

$$\%CC = \frac{\text{Número de pontos Classificados Corretamente}}{\text{Número total de pontos}} \times 100. \quad (4.57)$$

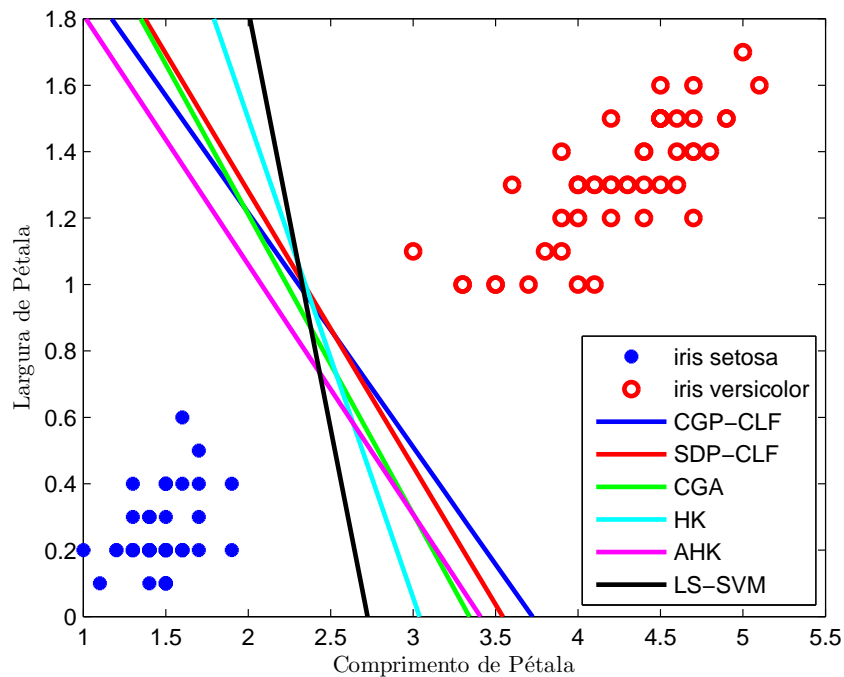


Figura 4.6: Superfícies separadoras das classes *iris setosa* and *iris versicolor*. Todos os algoritmos considerados classificam corretamente todas as amostras. A margem de distância M obtida após a convergência é mostrado na tabela 4.1. O critério de parada é $\|\mathbf{r}_k\| \leq 0.001$ e o número máximo de iterações é 1000.

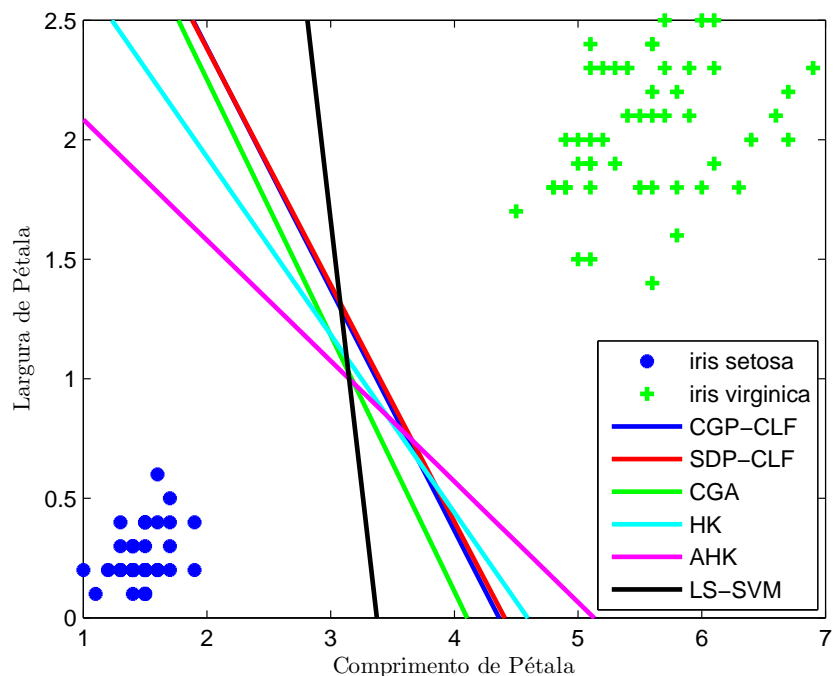


Figura 4.7: Superfícies separadoras das classes *iris setosa* and *iris virginica*. Todos os algoritmos considerados classificam corretamente todas as amostras. A margem de distância M obtida após a convergência é mostrado na tabela 4.1. O critério de parada é $\|\mathbf{r}_k\| \leq 0.001$ e o número máximo de iterações é 1000.

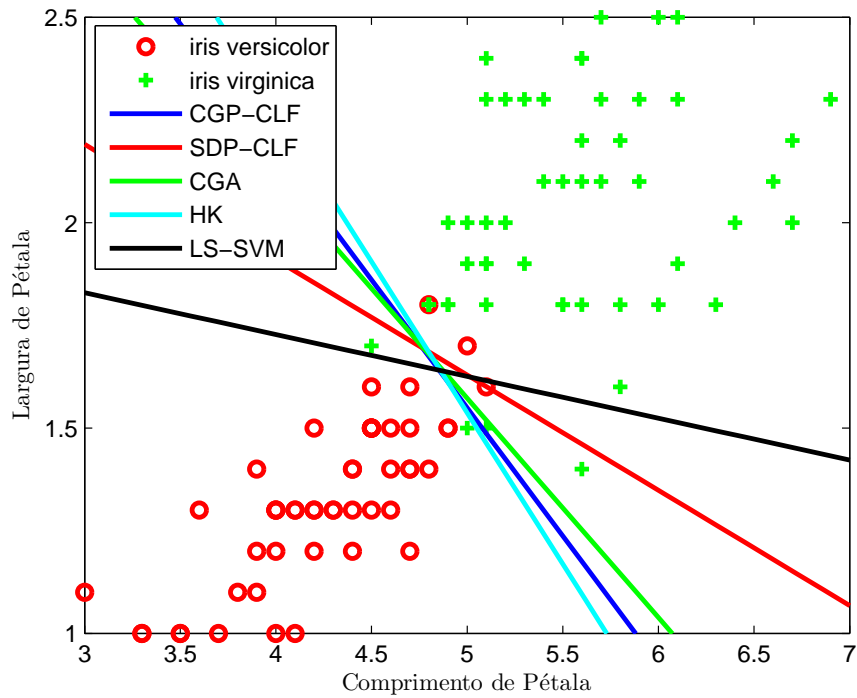


Figura 4.8: Superfícies separadoras das classes *iris versicolor* and *iris virginica*. A tabela 4.1 mostra a estatística das classificações e a margem M obtidas após alcançar o número máximo de iterações. O critério de parada é $\|\mathbf{r}_k\| \leq 0.001$ e o número máximo de iterações é 1000.

Os resultados da tabela 4.1 mostram que o algoritmo 4.4 converge em menos iterações e apresenta o resíduo de menor norma nos casos linearmente separáveis. No caso linearmente não separável todos os métodos apresentam um desempenho comparável exceto o Ho-Kashyap adaptativo que falha, classificando todos os dados como sendo da mesma classe.

Note que o custo computacional de uma iteração do algoritmo 4.3 é menor que uma do algoritmo 4.4, mas o algoritmo 4.3 realiza mais iterações, portanto o seu custo computacional total pode ser maior que o custo computacional total do algoritmo 4.4. Para ilustrar esse fato, realiza-se um experimento no âmbito de determinar o tempo necessário para a convergência dos algoritmos. O experimento consiste em executar 100 vezes cada algoritmo com as mesmas condições iniciais e determinar o tempo médio t (em segundos) e o número de iterações. O critério de parada é novamente escolhida tal que $\|\mathbf{r}_k\| \leq 0.001$. Os resultados deste experimento estão apresentados na tabela 4.4. Nota-se que, embora o algoritmo 4.3 tenha menos etapas que o algoritmo 4.4, esse último converge mais rapidamente e então necessita de menos tempo que os outros algoritmos.

<i>Iris</i>	Algoritmo	%CC	M	$\ \mathbf{r}_k\ $
<i>Setosa</i> × <i>Versicolor</i>	HK	100	1,3050	$9,96 \times 10^{-4}$
	AHK	100	1,3118	$3,41 \times 10^{-1}$
	CGA	100	1,3105	$7,30 \times 10^{-3}$
	CGP-CLF	100	1,2989	$8,01 \times 10^{-4}$
	SDP-CLF	100	1,2893	$9,51 \times 10^{-1}$
	LS-SVM	100	1,3133	×
<i>Setosa</i> × <i>Virginica</i>	HK	100	1,2561	$9,98 \times 10^{-4}$
	AHK	100	1,2625	$1,74 \times 10^{-1}$
	CGA	100	1,2554	$2,70 \times 10^{-3}$
	CGP-CLF	100	1,2504	$8,42 \times 10^{-4}$
	SDP-CLF	100	1,2463	$6,96 \times 10^{-1}$
	LS-SVM	100	1,2664	×
<i>Versicolor</i> × <i>Virginica</i>	HK	95	3,3292	2,01
	AHK	50	0,8636	$2,75 \times 10^{77}$
	CGA	94	3,3297	3,59
	CGP-CLF	95	3,3295	2,08
	SDP-CLF	94	3,3318	3,52
	LS-SVM	95	2,9931	×

Tabela 4.1: Desempenho dos algoritmos na separação das classes *iris setosa*, *iris versicolor* e *iris virginica*. A porcentagem %CC é dado por (4.57) e a margem M é o mínimo das distâncias dos pontos corretamente classificados ao hiperplano de separação. O número máximo de iterações é igual a 1000 e o critério de parada é $\|\mathbf{r}_k\| \leq 0.001$.

Outros testes numéricos são realizados com os conjuntos “*Ionosphere*”, “*BUPA Liver Disorders*” e “*Pima Indian Diabetes*”, para confirmar a observação 4. Os resultados de classificação estão apresentados na tabela 4.4. Mais detalhes sobre esses três conjuntos podem ser encontrados em Newman et al. [95].

Os resultados numéricos obtidos mostram que os métodos CGP-CLF e SDP-CLF propostos apresentam um desempenho equivalente ou melhor que os métodos HK, AHK e CGA sem utilizar nenhum parâmetro empírico de aprendizagem e nenhuma técnica heurística de reinicialização do vetor de direção. Nota-se que para os casos de conjuntos não separáveis linearmente (i.e. $\mathcal{S}^+ = \emptyset$), os algoritmos propostos convergem para um vetor de margem $\mathbf{b}_k > \mathbf{0}$ e uma solução \mathbf{x}_k que minimiza $\|\mathbf{b}_k - \mathbf{A}\mathbf{x}_k\|$, que corresponde a classificar corretamente o máximo número de padrões.

<i>Iris</i>	Algoritmo	t	iter	<i>Iris</i>	Algoritmo	t	iter
<i>Setosa</i> × <i>Versicolor</i>	HK	0.0234	462	<i>Setosa</i> × <i>Virginica</i>	HK	0.0172	295
	AHK	0.0562	1000		AHK	0.0563	1000
	CGA	0.1094	1000		CGA	0.1109	1000
	CG-CLF	0.0327	79		CG-CLF	0.0421	107
	SD-CLF	0.3656	1000		SD-CLF	0.3609	1000
<i>Versicolor</i> × <i>Virginica</i>	HK	0.0407	1000				
	AHK	0.0579	1000				
	CGA	0.1125	1000				
	CG-CLF	0.3547	1000				
	SD-CLF	0.3548	1000				

Tabela 4.2: Comparação de tempo médio e numero de iterações.

<i>Set</i>	Algoritmo	%CC	$\ r_k\ $	<i>Set</i>	Algoritmo	%CC	$\ r_k\ $
Ionosphere	HK	×	×	BUPA	HK	66.2	15.54
	AHK	89.7	8.07		AHK	×	×
	CGA	91.3	7.83		CGA	66.9	15.61
	CG-CLF	91.3	7.89		CG-CLF	66.9	15.54
	SD-CLF	88.5	9.05		SD-CLF	67.2	15.73
Diabetes	HK	73.7	19.05				
	AHK	×	×				
	CGA	74.3	19.79				
	CG-CLF	73.9	19.02				
	SD-CLF	66.4	21.85				

Tabela 4.3: Comparação do desempenho dos métodos. O método HK falha na determinação da pseudo-inversa da matriz de dados do conjunto “*Ionosphere*”, e o método AHK diverge no caso dos conjuntos “*BUPA*” e “*Diabetes*”.

4.5 Contribuições e Publicações

Neste capítulo, baseado nas provas de estabilidade assintótica apresentadas nos teoremas 3 e 4, propõem-se os novos métodos SDP-CLF e CGP-CLF, extensões dos métodos SD e CG para o treinamento de perceptrons em modo batelada. Em termos de classificação de padrões, o desempenho dos novos métodos propostos é comparável ao dos métodos CGA [36] e LS-SVM [96], e superior ao dos métodos HK [31, 32] e AHK [35]. A convergência dos métodos HK e AHK não é garantida em todos os casos. O primeiro (HK) utiliza a pseudo-inversa da matriz de dados, para determinar um vetor solução, cuja determinação, além de ser computacionalmente custoso para problemas de grande escala (i.e. separação de classes com um grande numero de atributos), torna-se instável a medida que a matriz $A^T A$

se aproxima da singularidade. A convergência do segundo (AHK) depende criticamente da escolha empírica dos parâmetros de aprendizagem ρ_1 e ρ_2 , uma escolha inapropriada dos parâmetros de aprendizagem ρ_1 e ρ_2 resulta em um baixo desempenho do método AHK. Devido às perdas de ortogonalidade dos resíduos, causadas pelos erros de precisão finita, o método CGA de Nagaraja e Bose [36] é baseado em uma técnica heurística que consiste na reinicialização da direção do gradiente [94] para garantir a convergência do método. As vantagens dos métodos propostos, neste capítulo, são a não-utilização de nenhuma técnica heurística e de nenhum parâmetro empírico, e as provas de convergência obtidas a partir da formulação, adotada ao longo desta tese, dos mesmos como sistemas dinâmicos. Dois tópicos considerados como trabalhos futuros relacionados a este capítulo são a prova rigorosa da positividade do vetor de margem \mathbf{b}_k encontrado pelo algoritmo 4.4 e a utilização da técnica de CLF para desenvolver novos métodos de treinamento em-linha de perceptrons.

Os resultados e as contribuições apresentados neste capítulo, foram publicados através dos seguintes artigos de revista e congresso internacionais respectivamente:

1. DIENE, O., BHAYA, A., “Perceptron training algorithms designed using discrete-time control Liapunov functions”. *NEUROCOMPUTING*. Aceito para publicação, 2008.
2. DIENE, O., BHAYA, A., “Perceptron training algorithms designed using discrete-time control Liapunov functions”. In: *IEEE Multi-conference on Systems and Control (MSC) 2007*, pp. 608-613, Singapore, October 2007.

Capítulo 5

Estudo da robustez de métodos iterativos lineares

5.1 Introdução

Os métodos numéricos são implementados em computadores digitais utilizando a aritmética de precisão finita. Nesse caso os números reais/complexos são representados com palavras de tamanho finito. Essa representação resulta no truncamento/arredondamento dos números, o que acarreta erros nos algoritmos. Esses erros podem acarretar, por sua vez, uma perda de algumas das propriedades dos métodos numéricos (ortogonalidade dos resíduos do método do gradiente conjugado por exemplo), causando instabilidade numérica. A estabilidade e o condicionamento são conceitos utilizados na análise numérica para relacionar a resposta de um algoritmo computacional às perturbações decorrentes dos dados ou da utilização da aritmética de precisão finita [97–99]. Em Björck et al. [38], Paige et al. [39] e Giraud et al. [40] a estabilidade reversa de métodos do tipo Krylov foi estudada. Uma modelagem e um estudo dos erros de precisão finita foram feitas em Strakos e Tichý [41] e Meurant e Strakos [42] no contexto da estabilidade reversa dos métodos do gradiente conjugado e de Lanczos. Na linguagem de controle, os conceitos de estabilidade numérica seriam denominados, genericamente, de robustez do algoritmo numérico. Entretanto, há poucos estudos de robustez de algoritmos numéricos do ponto de vista de controle robusto.

Neste capítulo, propõe-se uma nova modelagem de perturbações oriundas da utilização de aritmética de precisão finita, baseando-se na combinação do modelo proposto em [41, 42]

com a modelagem usual de perturbações multiplicativas na entrada da planta, introduzida no trabalho de Doyle e Stein [46]. Esta perspectiva de controle, aplicada ao problema clássico de perturbações numéricas (de precisão finita), permitirá uma aplicação inédita do teorema de pequeno ganho para determinar medidas de robustez ou estabilidade numérica dos algoritmos propostos nesta tese. Através da utilização de CLFs, derivar-se-á outro resultado inédito sobre a margem de estabilidade dos métodos numéricos sujeitos a perturbações variantes no tempo. Enfatiza-se que o objetivo desta nova proposta não é substituir a análise de estabilidade reversa (backward stability) ou os outros tipos de estabilidade, consistência etc. propostos em Bunch et al. [43], e sim prover novas maneiras e medidas quantitativas, familiares a usuários da comunidade de engenheiros, da robustez da estabilidade de métodos iterativos estudados nesta tese. As principais vantagens desta nova proposta sobre os trabalhos anteriores [43, 45] são a simplicidade de aplicação a uma gama ampla de algoritmos e a obtenção de medidas quantitativas da robustez dos métodos numéricos. O foco dos autores em [43] é a aplicação de técnicas de álgebra linear numérica à análise de algoritmos de processamento de sinais, no entanto, a complexidade destas técnicas, aliada a não-familiaridade das mesmas para a maioria dos engenheiros têm limitado a sua aplicação a outros algoritmos. O estudo de sistemas dinâmicos perturbados feito em Grüne [45] apenas considera as perturbações decrescentes no tempo são considerados e é necessário o conhecimento prévio das trajetórias dos estados do sistema, uma vez que o ganho de robustez é obtido a partir de limitantes superiores de uma função das trajetórias dos estados. Por fim mostrar-se-á através de simulações numéricas que os métodos numéricos desenvolvidos a partir de funções de Liapunov com controle apresentam uma medida de robustez superior ou igual à dos outros métodos numéricos.

5.2 Formulação do problema

Em Meurant e Strakos [42] mostra-se que os erros numéricos devidos à aritmética de precisão finita resultam na perda de ortogonalidade dos vetores de resíduos e de direção conjugada dos algoritmos Lanczos e gradiente conjugado. Nesse trabalho o modelo padrão para operações em ponto flutuante é utilizado. Neste caso a representação de uma variável X é denominada $fl(X)$, e para as quatro operações aritméticas básicas (+, -, *, /), denominadas

por op , tem-se

$$fl(x \text{ op } y) = (x \text{ op } y)(1 + \beta), \quad |\beta| \leq \epsilon \quad (5.1)$$

sendo ϵ a unidade de precisão da máquina utilizada (diz-se que β é de ordem $O(\epsilon)$) [100]. Para estudar os efeitos dos erros numéricos devidos à aritmética de precisão finita, utilizando a formulação introduzida em Strakos e Tichý [41], reescreve-se o sistema dinâmico (2.13) correspondente ao problema da resolução de (2.9) por métodos iterativos em aritmética de ponto flutuante da seguinte forma:

$$\begin{cases} fl(\mathbf{x}_{k+1}) = \tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k \oplus \tilde{\mathbf{u}}_k = (\tilde{\mathbf{x}}_k + \tilde{\mathbf{u}}_k)(1 + \beta) \\ fl(\mathbf{y}_{k+1}) = \tilde{\mathbf{y}}_{k+1} = \tilde{\mathbf{A}} \otimes \tilde{\mathbf{x}}_{k+1} = (\tilde{\mathbf{A}}\tilde{\mathbf{x}}_{k+1})(1 + \beta) \\ fl(\mathbf{r}_{k+1}) = \tilde{\mathbf{r}}_{k+1} = \tilde{\mathbf{b}} \ominus \tilde{\mathbf{y}}_{k+1} = (\tilde{\mathbf{b}} - \tilde{\mathbf{y}}_{k+1})(1 + \beta) \\ fl(\mathbf{u}_{k+1}) = \tilde{\mathbf{u}}_{k+1} = \tilde{\mathbf{f}}(\tilde{\mathbf{r}}_{k+1}) \end{cases} \quad (5.2)$$

Os efeitos dos erros de aritmética de precisão finita na estabilidade interna do sistema dinâmico (2.9) podem ser estudados formulando o problema como um problema de sistema dinâmico com perturbações. Isso é feito determinando a propagação desses erros de precisão finita a partir da lei de atualização do estado da planta até a determinação da lei de controle (do estado interno até a entrada da planta). Para tanto, de (5.2), escreve-se

$$\begin{aligned} \tilde{\mathbf{x}}_{k+1} &= (\tilde{\mathbf{x}}_k + \tilde{\mathbf{u}}_k)(1 + \beta) = \tilde{\mathbf{x}}_k + \tilde{\mathbf{u}}_k + \beta(\tilde{\mathbf{x}}_k + \tilde{\mathbf{u}}_k) \\ &= \mathbf{x}_k + \beta^{\mathbf{x}_k} + \mathbf{u}_k + \beta^{\mathbf{u}_k} + \mathbf{x}_k\beta + \mathbf{x}_k\beta^{\mathbf{x}_k}\beta + \mathbf{u}_k\beta^{\mathbf{u}_k}\beta \\ &= \mathbf{x}_k + \mathbf{u}_k + (2\beta + \beta^2)\mathbf{x}_k + (2\beta + \beta^2)\mathbf{u}_k \\ &= \mathbf{x}_k + \mathbf{u}_k + \beta^{\mathbf{x}_k+1} = \mathbf{x}_{k+1} + \beta^{\mathbf{x}_k+1} \end{aligned} \quad (5.3)$$

sendo $\beta^{\mathbf{x}_k} = \mathbf{x}_k\beta$ e $\beta^{\mathbf{u}_k} = \mathbf{u}_k\beta$ os erros de precisão finita nos vetores \mathbf{x}_k e \mathbf{u}_k respectivamente e

$$\beta^{\mathbf{x}_k+1} = (2\beta + \beta^2)(\mathbf{x}_k + \mathbf{u}_k) = (2\beta + \beta^2)\mathbf{x}_{k+1} \quad (5.4)$$

o erro total cometido na determinação de \mathbf{x}_{k+1} em precisão finita. O próximo passo consiste em determinar o erro de precisão finita em \mathbf{y}_{k+1} a partir de (5.2) e (5.3) da seguinte forma:

$$\begin{aligned}
\tilde{\mathbf{y}}_{k+1} &= (\tilde{\mathbf{A}}\tilde{\mathbf{x}}_{k+1})(1 + \beta) = (\mathbf{A} + \beta^{\mathbf{A}})(\mathbf{x}_{k+1} + \beta^{\mathbf{x}_{k+1}})(1 + \beta) \\
&= \mathbf{A}\mathbf{x}_{k+1} + \mathbf{A}\mathbf{x}_{k+1}\beta + \mathbf{A}\beta^{\mathbf{x}_{k+1}} + \mathbf{A}\beta^{\mathbf{x}_{k+1}}\beta + \beta^{\mathbf{A}}\mathbf{x}_{k+1} + \beta^{\mathbf{A}}\mathbf{x}_{k+1}\beta \\
&\quad + \beta^{\mathbf{A}}\beta^{\mathbf{x}_{k+1}} + \beta^{\mathbf{A}}\beta^{\mathbf{x}_{k+1}}\beta \\
&= \mathbf{A}\mathbf{x}_{k+1} + (4\beta + 6\beta^2 + 4\beta^3 + \beta^4)\mathbf{A}\mathbf{x}_{k+1} \\
&= \mathbf{A}\mathbf{x}_{k+1} + \beta^{\mathbf{y}_{k+1}} = \mathbf{y}_{k+1} + \beta^{\mathbf{y}_{k+1}}
\end{aligned} \tag{5.5}$$

sendo $\beta^{\mathbf{A}} = \mathbf{A}\beta$ o erro de precisão finita na matriz \mathbf{A} e

$$\beta^{\mathbf{y}_{k+1}} = (4\beta + 6\beta^2 + 4\beta^3 + \beta^4)\mathbf{A}\mathbf{x}_{k+1} = (4\beta + 6\beta^2 + 4\beta^3 + \beta^4)\mathbf{y}_{k+1} \tag{5.6}$$

o erro total cometido na determinação de \mathbf{y}_{k+1} em precisão finita. Para o vetor de resíduo \mathbf{r}_{k+1} o erro de precisão finita é determinado da mesma forma a partir de (5.2) e (5.5)

$$\begin{aligned}
\tilde{\mathbf{r}}_{k+1} &= (\tilde{\mathbf{b}} - \tilde{\mathbf{y}}_{k+1})(1 + \beta) = \tilde{\mathbf{b}} - \tilde{\mathbf{y}}_{k+1} + \beta(\tilde{\mathbf{b}} - \tilde{\mathbf{y}}_{k+1}) \\
&= \mathbf{b} + \beta^{\mathbf{b}} - \mathbf{y}_{k+1} - \beta^{\mathbf{y}_{k+1}} + \mathbf{b}\beta + \beta^{\mathbf{b}}\beta - \mathbf{y}_{k+1}\beta - \beta^{\mathbf{y}_{k+1}}\beta \\
&= \mathbf{b} - \mathbf{y}_{k+1} + (2\beta + \beta^2)\mathbf{b} - (5\beta + 10\beta^2 + 10\beta^3 + 5\beta^4 + \beta^5)\mathbf{y}_{k+1} \\
&= \mathbf{b} - \mathbf{y}_{k+1} + \beta^{\mathbf{r}_{k+1}} = \mathbf{r}_{k+1} + \beta^{\mathbf{r}_{k+1}}
\end{aligned} \tag{5.7}$$

sendo $\beta^{\mathbf{b}} = \mathbf{b}\beta$ o erro de precisão finita no vetor \mathbf{b} e

$$\beta^{\mathbf{r}_{k+1}} = (2\beta + \beta^2)\mathbf{b} - (5\beta + 10\beta^2 + 10\beta^3 + 5\beta^4 + \beta^5)\mathbf{y}_{k+1} \tag{5.8}$$

o erro total cometido na determinação de \mathbf{r}_{k+1} em precisão finita. Finalmente o erro na lei de controle \mathbf{u}_{k+1} pode ser determinado da seguinte forma:

$$\tilde{\mathbf{u}}_{k+1} = \tilde{\mathbf{f}}(\tilde{\mathbf{r}}_{k+1}) = \tilde{\mathbf{f}}(\mathbf{r}_{k+1} + \beta^{\mathbf{r}_{k+1}}) = \mathbf{f}(\mathbf{r}_{k+1}) + \mathbf{f}_1(\beta^{\mathbf{r}_{k+1}}) = \mathbf{u}_{k+1} + \beta^{\mathbf{u}_{k+1}} \tag{5.9}$$

sendo

$$\beta^{\mathbf{u}_{k+1}} = \mathbf{f}_1(\beta^{\mathbf{r}_{k+1}}) \tag{5.10}$$

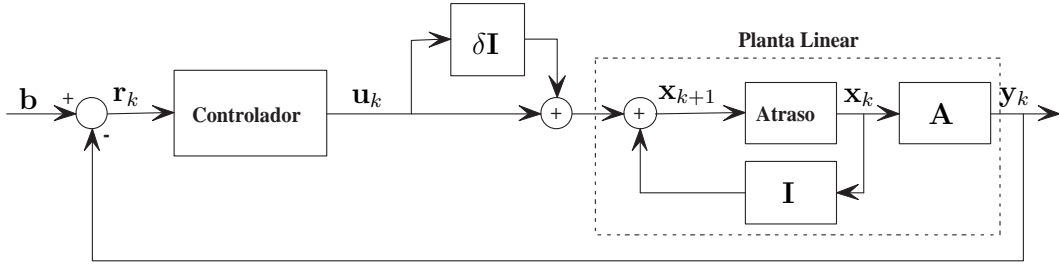


Figura 5.1: Diagrama de blocos ilustrativo do problema de regulação equivalente à resolução de (2.9) por métodos iterativos em aritmética de precisão finita. Os erros de precisão finita são representados como uma perturbação multiplicativa δ na entrada da planta.

o erro total cometido na determinação de \mathbf{u}_{k+1} em precisão finita. Portanto o sistema (5.2) resulta em

$$\begin{cases} \tilde{\mathbf{x}}_{k+1} = \mathbf{x}_k + \mathbf{u}_k + \beta^{\mathbf{x}_{k+1}} = \mathbf{x}_{k+1} + \beta^{\mathbf{x}_{k+1}} \\ \tilde{\mathbf{y}}_{k+1} = \mathbf{A}\mathbf{x}_{k+1} + \beta^{\mathbf{y}_{k+1}} = \mathbf{y}_{k+1} + \beta^{\mathbf{y}_{k+1}} \\ \tilde{\mathbf{r}}_{k+1} = \mathbf{b} - \mathbf{y}_{k+1} + \beta^{\mathbf{r}_{k+1}} = \mathbf{r}_{k+1} + \beta^{\mathbf{r}_{k+1}} \\ \tilde{\mathbf{u}}_{k+1} = \mathbf{f}(\mathbf{r}_{k+1}) + \mathbf{f}_1(\beta^{\mathbf{r}_{k+1}}) = \mathbf{u}_{k+1} + \mathbf{f}_1(\beta^{\mathbf{r}_{k+1}}) \end{cases} \quad (5.11)$$

Note que o sistema (5.11) é uma generalização dos sistemas considerados em [41, 42] no estudo dos algoritmos Lanczos e Gradiente Conjugado em aritmética de precisão finita. Desprezando os termos de ordem $O(\epsilon^2)$, $O(\epsilon^3)$, $O(\epsilon^4)$ e $O(\epsilon^5)$, as equações (5.4), (5.6), (5.8) e (5.10) podem ser reescritas como

$$\beta^{\mathbf{x}_{k+1}} = 2\beta\mathbf{x}_{k+1} \quad (5.12)$$

$$\beta^{\mathbf{y}_{k+1}} = 4\beta\mathbf{y}_{k+1} \quad (5.13)$$

$$\beta^{\mathbf{r}_{k+1}} = 2\beta\mathbf{b} - 5\beta\mathbf{y}_{k+1} \quad (5.14)$$

$$\beta^{\mathbf{u}_{k+1}} = \mathbf{f}_1(\beta^{\mathbf{r}_{k+1}}) = \mathbf{f}_1(\beta, \mathbf{x}_k, \mathbf{A}, \mathbf{b}, \mathbf{u}_k) \quad (5.15)$$

Note que todos os erros numéricos cometidos nos passos anteriores a determinação da lei de controle estão concentrados em $\beta^{\mathbf{u}_{k+1}} = \mathbf{f}_1(\beta, \mathbf{x}_k, \mathbf{A}, \mathbf{b}, \mathbf{u}_k)$. Defina δ tal que

$$\beta^{\mathbf{u}_{k+1}} = \mathbf{f}_1(\beta^{\mathbf{r}_{k+1}}) = \delta\mathbf{u}_{k+1}, \quad (5.16)$$

então $\tilde{\mathbf{u}}_{k+1} = (1 + \delta)\mathbf{u}_{k+1}$. O sistema (5.11), equivalente ao sistema dinâmico (2.13) em

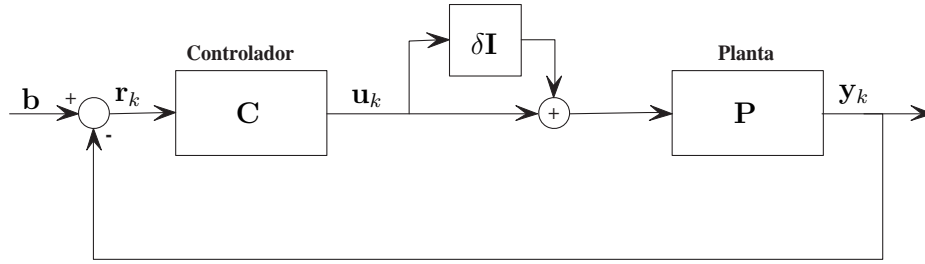


Figura 5.2: Forma geral do diagrama de blocos ilustrativo do problema de regulação equivalente à resolução de (2.9) por métodos iterativos em aritmética de precisão finita. Os erros de precisão finita são representados como uma perturbação multiplicativa δ na entrada da planta. Para os métodos iterativos estacionários a planta P e o controlador C podem ser representados tanto no domínio do tempo quanto no domínio da frequência.

aritmética de precisão finita, pode ser aproximado da seguinte forma:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + (1 + \delta)\mathbf{u}_k \\ \mathbf{y}_{k+1} = \mathbf{A}\mathbf{x}_{k+1} \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{y}_{k+1} \\ \mathbf{u}_{k+1} = \mathbf{f}(\mathbf{r}_{k+1}) \end{cases} \quad (5.17)$$

Neste caso a cada atualização do estado \mathbf{x}_{k+1} , os erros de precisão finita de todas as etapas anteriores estão representados pelo termo $\delta\mathbf{u}_k$. A figura 5.1 apresenta o sistema (5.17) na forma de diagrama de blocos.

Portanto o estudo dos efeitos dos erros numéricos, devidos à aritmética de precisão finita, na convergência dos métodos numéricos é equivalente ao estudo da estabilidade assintótica do equilíbrio do sistema (5.17) na presença da perturbação δ . Isso é equivalente a representar os erros numéricos de precisão finita como uma perturbação multiplicativa na entrada da planta do sistema dinâmico (2.13).

5.3 Métodos iterativos estacionários

Os métodos iterativos estacionários podem ser representados como sistemas dinâmicos constituídos por uma planta linear com um controlador estático fixo em realimentação negativa [11]. Portanto pode-se encontrar uma representação desses métodos no domínio da frequência através das funções de transferência da planta ($G(z)$) e do controlador ($K(z)$). A

figura 5.2 apresenta uma representação em diagrama de blocos da forma geral do sistema dinâmico (5.17), sendo que para os métodos iterativos estacionários a planta \mathbf{P} e o controlador \mathbf{C} podem ser representados tanto no domínio do tempo quanto no domínio da frequência. Para o sistema representado na figura 5.2 um resultado clássico é o teorema do pequeno ganho [101–103] que pode ser formulado no domínio da frequência da seguinte forma:

Lema 1. $\mathbf{K}(z)$ estabiliza $\mathbf{G}(z)$ na presença de δ se e somente se $\mathbf{K}(z)$ estabiliza $\mathbf{G}(z)$ e

$$\sup_{z \in Z_e} \bar{\sigma} \{ \mathbf{K}(z) \mathbf{G}(z) [\mathbf{I} + \mathbf{K}(z) \mathbf{G}(z)]^{-1} \} < \frac{1}{|\delta|}, \quad (5.18)$$

sendo $\bar{\sigma}\{\cdot\}$ o maior valor singular da matriz em questão e Z_e a região de estabilidade.

Prova do Lema 1. Ver [101–105]. ■

Pelo teorema do módulo máximo [103], a equação (5.18) pode ser reescrita como

$$\sup_{z \in \partial Z_e} \bar{\sigma} \{ \mathbf{K}(z) \mathbf{G}(z) [\mathbf{I} + \mathbf{K}(z) \mathbf{G}(z)]^{-1} \} < \frac{1}{|\delta|},$$

sendo ∂Z_e a fronteira da região de estabilidade, ou

$$\sup_{|z|=1} \bar{\sigma} \{ \mathbf{K}(z) \mathbf{G}(z) [\mathbf{I} + \mathbf{K}(z) \mathbf{G}(z)]^{-1} \} < \frac{1}{|\delta|}. \quad (5.19)$$

Esse último resultado pode ser utilizado para determinar uma medida da robustez dos métodos iterativos estacionários, isto é, determinar o máximo de perturbações multiplicativas na entrada da planta que o sistema dinâmico (5.17) tolera sem perder a estabilidade assintótica.

5.3.1 Método de Jacobi

O método de Jacobi pode ser representado através do seguinte sistema [11]

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k \\ \mathbf{y}_{k+1} = \mathbf{A} \mathbf{x}_{k+1} \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{y}_{k+1} \\ \mathbf{u}_{k+1} = \mathbf{D}^{-1} \mathbf{r}_{k+1}, \end{cases} \quad (5.20)$$

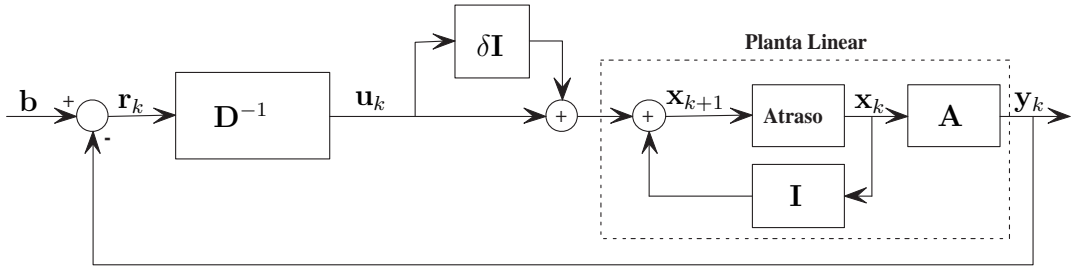


Figura 5.3: Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de Jacobi em aritmética de precisão finita no domínio do tempo. A perturbação δ representa os erros de precisão finita.

sendo \mathbf{D} a matriz diagonal contendo os elementos da diagonal da matriz \mathbf{A} . Do sistema (5.17), pode-se representar o método de Jacobi em precisão finita da seguinte forma:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + (1 + \delta)\mathbf{u}_k \\ \mathbf{y}_{k+1} = \mathbf{A}\mathbf{x}_{k+1} \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{y}_{k+1} \\ \mathbf{u}_{k+1} = \mathbf{D}^{-1}\mathbf{r}_{k+1} \end{cases} \quad (5.21)$$

Para aplicar o teorema do pequeno ganho ao método de Jacobi, representado em espaço de estado pelo sistema (5.21) e em diagrama de blocos na figura 5.3, determinam-se as funções de transferência discretas da planta e do controlador correspondente ao sistema (5.21). As representações em espaço de estado da planta e do controlador que formam o sistema dinâmico equivalente ao método de Jacobi são dados respectivamente por $\mathbf{P} = \{\mathbf{I}, \mathbf{I}, \mathbf{A}, \mathbf{0}\}$ e $\mathbf{C} = \{\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{D}^{-1}\}$. Portanto as funções de transferências correspondentes são:

$$\mathbf{G}(z) = \mathbf{A}(z\mathbf{I} - \mathbf{I})^{-1}\mathbf{I} + \mathbf{0} = \frac{1}{z - 1}\mathbf{A} \quad (5.22)$$

$$\mathbf{K}(z) = \mathbf{0}(z\mathbf{I} - \mathbf{0})^{-1}\mathbf{0} + \mathbf{D}^{-1} = \mathbf{D}^{-1} \quad (5.23)$$

A figura 5.4 apresenta o diagrama de blocos do sistema dinâmico equivalente ao método de Jacobi no domínio da frequência.

Uma condição suficiente para a convergência do método de Jacobi é a dominância dia-

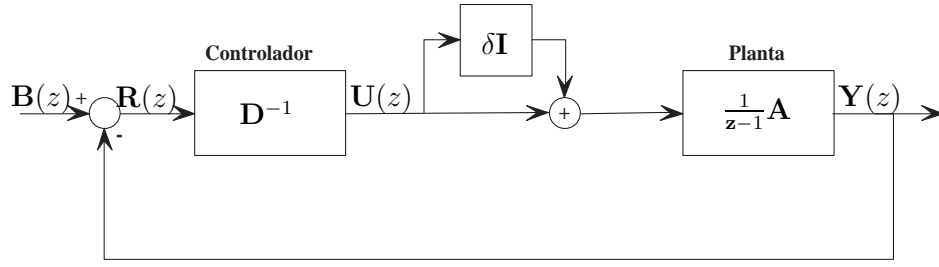


Figura 5.4: Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de Jacobi em aritmética de precisão finita no domínio da frequência. A perturbação δ representa os erros de precisão finita.

gonal da matriz \mathbf{A} . Para matrizes com dominância diagonal, isto é

$$|a_{jj}| > \sum_{\substack{i=1 \\ i \neq j}}^n |a_{ij}|, \quad 1 \leq j \leq n, \quad (5.24)$$

a seqüência produzida pela iteração de Jacobi (5.20) converge para a solução \mathbf{x}_o de (2.9) para qualquer ponto inicial \mathbf{x}_0 [11]. Para a classe de matrizes com dominância diagonal um resultado sobre a robustez do método de Jacobi pode ser enunciado da seguinte forma:

Teorema 5. *O equilíbrio do sistema representado na figura (5.4) é assintoticamente estável se*

$$|\delta| < \frac{\sigma\{\Delta - \mathbf{I}\}}{\bar{\sigma}\{\mathbf{I} + \Delta\}} \leq \frac{1}{\bar{\sigma}\{(\mathbf{I} + \Delta)(\Delta - \mathbf{I})^{-1}\}}, \quad (5.25)$$

sendo $\Delta = \{\delta_{ij}\}$, $\delta_{ii} = 0$ e $|\delta_{ij}| < 1, \forall i \neq j$ tal que $\mathbf{D}^{-1}\mathbf{A} = \mathbf{I} + \Delta$.

Prova do Teorema 5. *Note que das equações (5.22), (5.23) e (5.24) tem-se*

$$\mathbf{K}(z)\mathbf{G}(z) = \frac{1}{z-1}\mathbf{D}^{-1}\mathbf{A} = \frac{1}{z-1}(\mathbf{I} + \Delta), \quad (5.26)$$

sendo $\Delta = \{\delta_{ij}\}$, $\delta_{ii} = 0$ e $|\delta_{ij}| < 1, \forall i \neq j$. Portanto

$$\begin{aligned} \mathbf{K}(z)\mathbf{G}(z)[\mathbf{I} + \mathbf{K}(z)\mathbf{G}(z)]^{-1} &= \frac{1}{z-1}(\mathbf{I} + \Delta) \left[\mathbf{I} + \frac{1}{z-1}(\mathbf{I} + \Delta) \right]^{-1} \\ &= \frac{1}{z-1}(\mathbf{I} + \Delta) \left[\frac{z-1+1}{z-1}\mathbf{I} + \frac{1}{z-1}\Delta \right]^{-1} \\ &= \frac{1}{z-1}(\mathbf{I} + \Delta) \left[\frac{1}{z-1}(z\mathbf{I} + \Delta) \right]^{-1} \\ &= (\mathbf{I} + \Delta)(z\mathbf{I} + \Delta)^{-1}. \end{aligned} \quad (5.27)$$

Pelo lema 1 e o teorema do módulo máximo [103], $\mathbf{K}(z)$ estabiliza $\mathbf{G}(z)$ na presença de δ se e somente se $\mathbf{K}(z)$ estabiliza $\mathbf{G}(z)$ e

$$\sup_{|z|=1} \bar{\sigma} \{ \mathbf{K}(z)\mathbf{G}(z)[\mathbf{I} + \mathbf{K}(z)\mathbf{G}(z)]^{-1} \} < \frac{1}{|\delta|}. \quad (5.28)$$

Como sabe-se que $\mathbf{K}(z)$ estabiliza $\mathbf{G}(z)$ [11], o próximo passo consiste em determinar a condição (5.25) a partir de (5.28). Para tanto, note que substituindo (5.27) na equação (5.28) resulta em:

$$\sup_{|z|=1} \bar{\sigma} \{ (\mathbf{I} + \mathbf{\Delta})(z\mathbf{I} + \mathbf{\Delta})^{-1} \} < \frac{1}{|\delta|},$$

que pode ser reescrito como

$$\max_{0 \leq \theta \leq 2\pi} \bar{\sigma} \{ (\mathbf{I} + \mathbf{\Delta})(\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})^{-1} \} < \frac{1}{|\delta|}. \quad (5.29)$$

Pode-se determinar o valor de θ_o que maximiza (5.29) notando que

$$\begin{aligned} \theta_o &= \arg \max_{0 \leq \theta \leq 2\pi} \bar{\sigma} \{ (\mathbf{I} + \mathbf{\Delta})(\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})^{-1} \} \\ &= \arg \max_{0 \leq \theta \leq 2\pi} \bar{\sigma} \{ (\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})^{-1} \} \\ &= \arg \min_{0 \leq \theta \leq 2\pi} \underline{\sigma} \{ \exp^{j\theta} \mathbf{I} + \mathbf{\Delta} \} \\ &= \arg \min_{0 \leq \theta \leq 2\pi} \left\{ \min_{\mathbf{x}} \frac{\|(\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})\mathbf{x}\|}{\|\mathbf{x}\|} \right\} \\ &= \arg \min_{0 \leq \theta \leq 2\pi} \|(\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})\mathbf{x}\| \\ &= \arg \min_{0 \leq \theta \leq 2\pi} \|(\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})\mathbf{x}\|^2. \end{aligned}$$

Por outro lado note também que

$$\begin{aligned} \|(\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})\mathbf{x}\|^2 &= (\exp^{j\theta} \mathbf{x} + \mathbf{\Delta}\mathbf{x})^H (\exp^{j\theta} \mathbf{x} + \mathbf{\Delta}\mathbf{x}) \\ &= \exp^{j2\theta} \mathbf{x}^H \mathbf{x} + \exp^{j\theta} \mathbf{x}^H (\mathbf{\Delta} + \mathbf{\Delta}^H)\mathbf{x} + \mathbf{x}^H \mathbf{\Delta}^H \mathbf{\Delta} \mathbf{x}. \end{aligned}$$

O valor de θ_o pode ser encontrada resolvendo a seguinte equação

$$\begin{aligned}
\frac{\partial \|(\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})\mathbf{x}\|^2}{\partial \theta} &= 2j \exp^{j2\theta} \mathbf{x}^H \mathbf{x} + j \exp^{j\theta} \mathbf{x}^H (\mathbf{\Delta} + \mathbf{\Delta}^H) \mathbf{x} = 0 \\
&\Leftrightarrow 2 \exp^{j\theta} \mathbf{x}^H \mathbf{x} + \mathbf{x}^H (\mathbf{\Delta} + \mathbf{\Delta}^H) \mathbf{x} = 0 \\
&\Leftrightarrow \exp^{j\theta} = -\frac{\mathbf{x}^H (\mathbf{\Delta} + \mathbf{\Delta}^H) \mathbf{x}}{2\mathbf{x}^H \mathbf{x}}.
\end{aligned} \tag{5.30}$$

Como o termo do lado direito de 5.30 é sempre real e negativo, a única solução possível é $\theta_o = \pi$ e $\mathbf{x}^H (\mathbf{\Delta} + \mathbf{\Delta}^H) \mathbf{x} = 2\mathbf{x}^H \mathbf{x}$. Portanto tem-se

$$\begin{aligned}
\theta_o &= \arg \max_{0 \leq \theta \leq 2\pi} \bar{\sigma} \{(\mathbf{I} + \mathbf{\Delta})(\exp^{j\theta} \mathbf{I} + \mathbf{\Delta})^{-1}\} \\
&= \pi.
\end{aligned} \tag{5.31}$$

Substituindo (5.31) em (5.29) resulta em

$$\bar{\sigma} \{(\mathbf{I} + \mathbf{\Delta})(\mathbf{\Delta} - \mathbf{I})^{-1}\} < \frac{1}{|\delta|}. \tag{5.32}$$

Portanto

$$|\delta| < \frac{1}{\bar{\sigma} \{(\mathbf{I} + \mathbf{\Delta})(\mathbf{\Delta} - \mathbf{I})^{-1}\}} = \frac{1}{\bar{\sigma} \{\mathbf{D}^{-1} \mathbf{A} (\mathbf{D}^{-1} \mathbf{A} - 2\mathbf{I})^{-1}\}}. \tag{5.33}$$

Note que

$$\bar{\sigma} \{(\mathbf{I} + \mathbf{\Delta})(\mathbf{\Delta} - \mathbf{I})^{-1}\} \leq \bar{\sigma} \{\mathbf{I} + \mathbf{\Delta}\} \bar{\sigma} \{(\mathbf{\Delta} - \mathbf{I})^{-1}\} = \frac{\bar{\sigma} \{\mathbf{I} + \mathbf{\Delta}\}}{\underline{\sigma} \{\mathbf{\Delta} - \mathbf{I}\}},$$

o que implica que

$$\frac{\underline{\sigma} \{\mathbf{\Delta} - \mathbf{I}\}}{\bar{\sigma} \{\mathbf{I} + \mathbf{\Delta}\}} \leq \frac{1}{\bar{\sigma} \{\mathbf{D}^{-1} \mathbf{A} (\mathbf{D}^{-1} \mathbf{A} - 2\mathbf{I})^{-1}\}}. \tag{5.34}$$

Das equações (5.33) e (5.34) pode-se concluir que

$$|\delta| < \frac{\underline{\sigma} \{\mathbf{\Delta} - \mathbf{I}\}}{\bar{\sigma} \{\mathbf{I} + \mathbf{\Delta}\}} \leq \frac{1}{\bar{\sigma} \{(\mathbf{I} + \mathbf{\Delta})(\mathbf{\Delta} - \mathbf{I})^{-1}\}}.$$

■

Observação 5. Se a matriz \mathbf{A} for fortemente diagonal dominante, isto é:

$$|a_{jj}| \gg \sum_{\substack{i=1 \\ i \neq j}}^n |a_{ij}|, \quad 1 \leq j \leq n,$$

então $\Delta \approx 0$ e condição de estabilidade (5.25) resulta em

$$|\delta| < \frac{\sigma\{-\mathbf{I}\}}{\bar{\sigma}\{\mathbf{I}\}} = 1. \quad (5.35)$$

5.3.2 Método de Gauss-Seidel

O método de Gauss-Seidel pode ser representado através do seguinte sistema [11]

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k \\ \mathbf{y}_{k+1} = \mathbf{A}\mathbf{x}_{k+1} \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{y}_{k+1} \\ \mathbf{u}_{k+1} = \mathbf{M}^{-1}\mathbf{r}_{k+1}, \end{cases} \quad (5.36)$$

sendo \mathbf{M} é a parte triangular inferior de \mathbf{A} (incluindo a diagonal). Do sistema (5.17), pode-se representar o método de Gauss-Seidel em precisão finita da seguinte forma:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + (1 + \delta)\mathbf{u}_k \\ \mathbf{y}_{k+1} = \mathbf{A}\mathbf{x}_{k+1} \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{y}_{k+1} \\ \mathbf{u}_{k+1} = \mathbf{M}^{-1}\mathbf{r}_{k+1} \end{cases} \quad (5.37)$$

Para aplicar o teorema do pequeno ganho ao método de Gauss-Seidel, representado em espaço de estado pelo sistema (5.37) e em diagrama de blocos na figura 5.5, determinam-se as funções de transferência discretas da planta e do controlador correspondente ao sistema (5.37). As representações em espaço de estado da planta e do controlador que formam o sistema dinâmico equivalente ao método de Gauss-Seidel são dados respectivamente por $\mathbf{P} = \{\mathbf{I}, \mathbf{I}, \mathbf{A}, \mathbf{0}\}$ e $\mathbf{C} = \{\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{M}^{-1}\}$. Portanto as funções de transferências correspon-

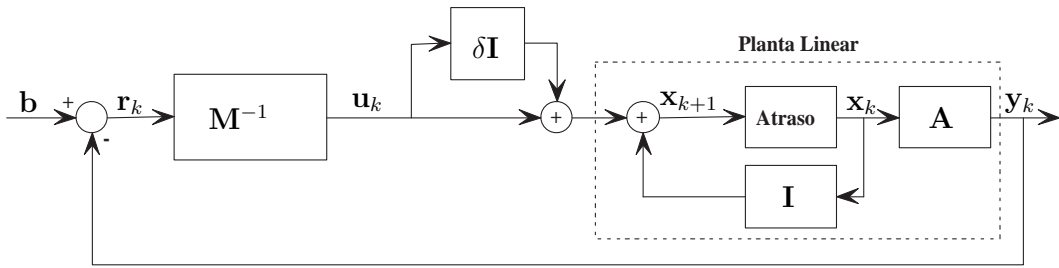


Figura 5.5: Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de Gauss-Seidel em aritmética de precisão finita no domínio do tempo. A perturbação δ representa os erros de precisão finita.

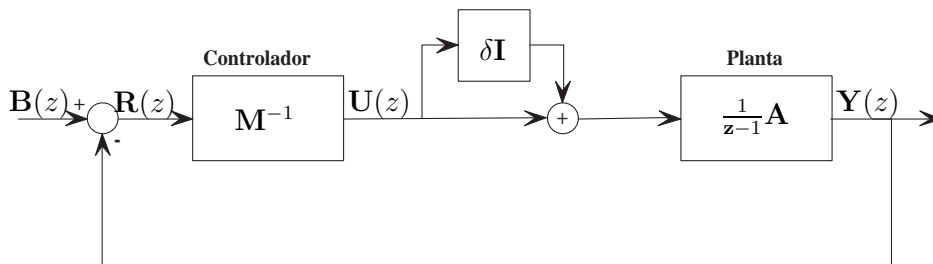


Figura 5.6: Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de Gauss-Seidel em aritmética de precisão finita no domínio da frequência. A perturbação δ representa os erros de precisão finita.

dentes são:

$$\mathbf{G}(z) = \mathbf{A}(z\mathbf{I} - \mathbf{I})^{-1}\mathbf{I} + \mathbf{0} = \frac{1}{z-1}\mathbf{A} \quad (5.38)$$

$$\mathbf{K}(z) = \mathbf{0}(z\mathbf{I} - \mathbf{0})^{-1}\mathbf{0} + \mathbf{M}^{-1} = \mathbf{M}^{-1} \quad (5.39)$$

A figura 5.6 apresenta o diagrama de blocos do sistema dinâmico equivalente ao método de Gauss-Seidel no domínio da frequência.

Analogamente ao método de Jacobi, uma condição suficiente para a convergência do método de Gauss-Seidel é a dominância diagonal da matriz \mathbf{A} , e novamente para matrizes com dominância diagonal a seqüência produzida pela iteração de Gauss-Seidel (5.36) converge para a solução \mathbf{x}_o de (2.9) para qualquer ponto inicial \mathbf{x}_0 [11]. Para a classe de matrizes com dominância diagonal um resultado sobre a robustez do método de Gauss-Seidel pode ser enunciado da seguinte forma:

Corolário 5.1. *O equilíbrio do sistema representado na figura (5.6) é assintoticamente es-*

tável se

$$|\delta| < \frac{\sigma\{\Delta - \mathbf{I}\}}{\bar{\sigma}\{\mathbf{I} + \Delta\}} \leq \frac{1}{\bar{\sigma}\{(\mathbf{I} + \Delta)(\Delta - \mathbf{I})^{-1}\}}, \quad (5.40)$$

sendo Δ tal que $\mathbf{M}^{-1}\mathbf{A} = \mathbf{I} + \Delta$.

Prova do Corolário 5.1. Substituindo \mathbf{D}^{-1} por \mathbf{M}^{-1} e escolhendo Δ tal que $\mathbf{M}^{-1}\mathbf{A} = \mathbf{I} + \Delta$, a prova segue da mesma forma que a prova do teorema 5. ■

5.4 Métodos iterativos dinâmicos

Em Diene [11], os métodos iterativos dinâmicos são representados como sistemas dinâmicos formados por uma planta linear com um controlador dinâmico com ganhos variantes no tempo. Nesse caso, devido aos ganhos variantes no tempo, o conceito de função de transferência não se aplica. O teorema do pequeno ganho utilizado para determinar uma medida de robustez dos métodos iterativos estacionários pode ser generalizado para todos os casos definindo operadores para a planta \mathbf{P} e o controlador \mathbf{C} . Nesse caso, o lema 1 é reescrito em função do ganho dos operadores de \mathbf{P} e \mathbf{C} [101]. É possível determinar operadores para \mathbf{P} e \mathbf{C} no caso dos métodos iterativos dinâmicos através de relações de entrada/saída, no entanto, devido aos parâmetros variantes no tempo, a determinação do ganho desses operadores é um problema difícil de resolver. Um conceito equivalente ao teorema do pequeno ganho para sistemas variantes no tempo é o conceito de estabilidade entrada-estado (ISS, do inglês *Input to State Stability*). O conceito ISS tem sido amplamente utilizado na literatura para o estudo de sistemas dinâmicos com perturbações [106–116]. O conceito ISS é útil para obter informações qualitativas sobre sistemas perturbados porém provê poucas informações quantitativas [45]. Em outras palavras, conhecendo a perturbação a propriedade ISS permite concluir sobre a estabilidade do sistema porém, a determinação de uma medida do máximo de perturbação que um sistema tolera sem perder sua estabilidade através do ISS é um problema difícil. Em Grüne [45] foi introduzido um conceito complementar, referenciado como estabilidade dinâmica entrada-estado (ISDS do inglês *Input to State Dynamical Stability*), na tentativa de obter informações quantitativas sobre os sistemas dinâmicos com perturbações decrescentes. Nesse caso um ganho de robustez é obtido a partir de limitantes superiores de uma função das trajetórias dos estados. No caso dos métodos iterativos dinâmicos as perturbações representando os erros de precisão finita não são decrescentes, portanto nesta tese

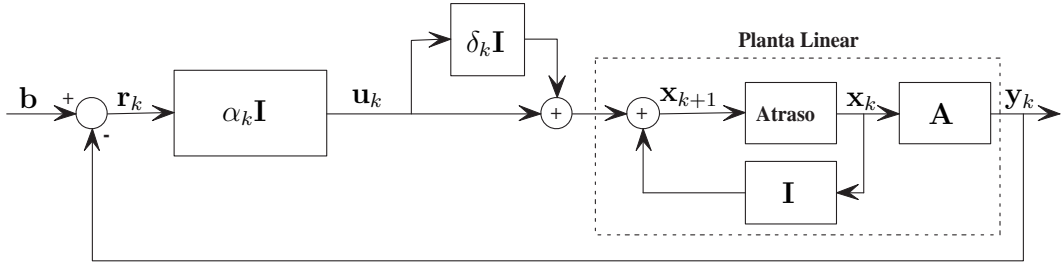


Figura 5.7: Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de SD em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

as mesmas funções de Liapunov com controle utilizadas para provar a convergências dos métodos iterativos serão novamente utilizadas para determinar uma medida da robustez dos mesmos.

5.4.1 Método SD-CLF para sistemas lineares

O método da declividade máxima (SD) pode ser representado pelo sistema de equações abaixo [11]

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{r}_k \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{r}_k, \end{cases} \quad (5.41)$$

sendo

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle}{\langle \mathbf{A}\mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle}. \quad (5.42)$$

Do sistema (5.17) representa-se o método SD em precisão finita da seguinte forma:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + (1 + \delta_k)\alpha_k \mathbf{r}_k \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{r}_k - (1 + \delta_k)\alpha_k \mathbf{A}\mathbf{r}_k, \end{cases} \quad (5.43)$$

sendo que neste caso a perturbação δ_k é considerada variante no tempo. Para o sistema (5.43), representado em forma de diagrama de blocos na figura 5.7, uma condição sobre a estabilidade assintótica do equilíbrio é dada pelo seguinte teorema

Teorema 6. Para α_k dado por (5.42), o equilíbrio do sistema (5.43) é assintoticamente estável se e somente se

$$|\delta_k| < 1. \quad (5.44)$$

Prova do Teorema 6. *Considere a função de Liapunov*

$$\mathbf{V}_r(k) = \langle \mathbf{r}_k, \mathbf{r}_k \rangle. \quad (5.45)$$

Tem-se que

$$\begin{aligned} \mathbf{V}_r(k+1) &= \langle \mathbf{r}_{k+1}, \mathbf{r}_{k+1} \rangle = \langle \mathbf{r}_k - (1 + \delta_k)\alpha_k \mathbf{A}\mathbf{r}_k, \mathbf{r}_k - (1 + \delta_k)\alpha_k \mathbf{A}\mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k, \mathbf{r}_k \rangle - 2(1 + \delta_k)\alpha_k \langle \mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle + (1 + \delta_k)^2 \alpha_k^2 \langle \mathbf{A}\mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle \end{aligned}$$

Portanto

$$\begin{aligned} \Delta \mathbf{V}_r &:= \mathbf{V}_r(k+1) - \mathbf{V}_r(k) \\ &= -2(1 + \delta_k)\alpha_k \langle \mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle + (1 + \delta_k)^2 \alpha_k^2 \langle \mathbf{A}\mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle \end{aligned} \quad (5.46)$$

Substituindo (5.42) em (5.46) resulta em

$$\begin{aligned} \Delta \mathbf{V}_r &= -2(1 + \delta_k) \frac{\langle \mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle^2}{\langle \mathbf{A}\mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle} + (1 + \delta_k)^2 \frac{\langle \mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle^2}{\langle \mathbf{A}\mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle} \\ &= (\delta_k^2 - 1) \frac{\langle \mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle^2}{\langle \mathbf{A}\mathbf{r}_k, \mathbf{A}\mathbf{r}_k \rangle}. \end{aligned} \quad (5.47)$$

Dessa forma, pode-se escrever

$$\Delta \mathbf{V}_r < 0 \Leftrightarrow \delta_k^2 - 1 < 0 \Leftrightarrow \delta_k^2 < 1 \Leftrightarrow |\delta_k| < 1. \quad (5.48)$$

Então o resíduo \mathbf{r}_k decresce na norma-2 se e somente se $|\delta_k| < 1$. ■

5.4.2 Método CG-CLF para sistemas lineares

O método do gradiente conjugado (CG) é dado pelo seguinte sistema dinâmico [11]

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k \\ \mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k, \end{cases} \quad (5.49)$$

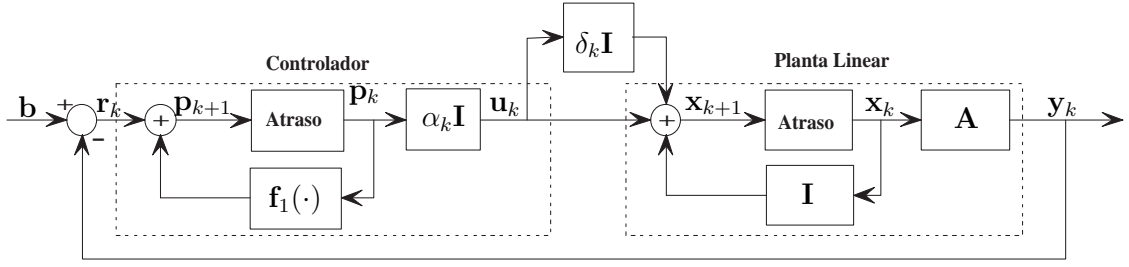


Figura 5.8: Diagrama de blocos do sistema dinâmico correspondente à resolução do sistema linear (2.9) pelo método de CG em aritmética de precisão finita, sendo $f_1(\mathbf{p}_k) = (\beta_k \mathbf{I} - \alpha_k \mathbf{A})\mathbf{p}_k$. A perturbação δ_k representa os erros de precisão finita.

sendo

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle} \quad (5.50)$$

$$\beta_k = -\frac{\langle \mathbf{r}_{k+1}, \mathbf{A}\mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle} \quad (5.51)$$

Do sistema (5.17), pode-se representar o método CG em aritmética de precisão finita da seguinte forma:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + (1 + \delta_k)\alpha_k \mathbf{p}_k \\ \mathbf{r}_{k+1} = \mathbf{b} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{r}_k - (1 + \delta_k)\alpha_k \mathbf{A}\mathbf{p}_k \\ \mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k, \end{cases} \quad (5.52)$$

O corolário 6.1 apresenta uma condição necessária e suficiente para a estabilidade assintótica do equilíbrio do sistema (5.52), representado em forma de diagrama de blocos na figura 5.8.

Corolário 6.1. Para α_k e β_k dados por (5.50) e (5.51), o equilíbrio do sistema (5.52) é assintoticamente estável se e somente se $|\delta_k| < 1$.

Prova do Corolário 6.1. Escolhendo a CLF

$$\mathbf{V}_r(k) = \langle \mathbf{r}_k, \mathbf{A}^{-1}\mathbf{r}_k \rangle,$$

a prova segue da mesma forma que a prova do teorema 6. ■

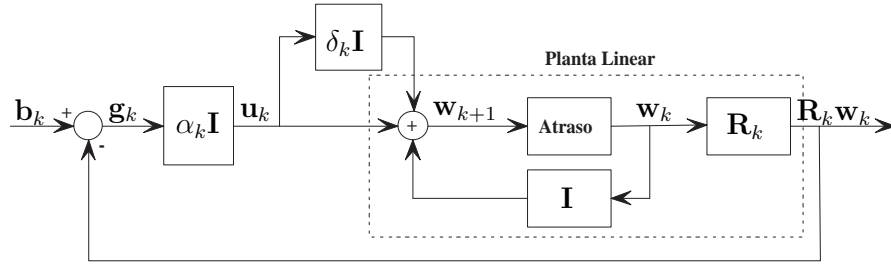


Figura 5.9: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa (2.7) pelo sistema dinâmico (5.53) correspondente ao método SD-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

5.4.3 Método SD-CLF para filtros adaptativos

Em aritmética de precisão finita, o método SD-CLF para filtros adaptativos dado pelo sistema (2.31) é representado pela figura 5.9 e pelo sistema seguinte

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + (1 + \delta_k)\alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - (1 + \delta_k)\alpha_k \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}^* \end{cases} \quad (5.53)$$

A condição necessária e suficiente para a estabilidade assintótica do equilíbrio do sistema (5.53) é dada pelo corolário 6.2 enunciado a seguir.

Corolário 6.2. *Sob a hipótese do teorema 1 e para α_k dado por (2.35), o equilíbrio do sistema (5.53) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.2. *A partir da prova do teorema 1 pode-se mostrar que o sistema (5.53) pode ser reescrito como*

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + (1 + \delta_k)\alpha_k \mathbf{g}_k \\ \tilde{\mathbf{g}}_{k+1} = \lambda_f \tilde{\mathbf{g}}_k - (1 + \delta_k)\alpha_k \lambda_f \mathbf{R}_{k+1} \mathbf{g}_k, \end{cases} \quad (5.54)$$

sendo $\tilde{\mathbf{g}}_k$ dado por (2.38). Escolha a CLF $V_{\tilde{\mathbf{g}}_k}(k) = \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle$, então

$$\begin{aligned} \Delta V_{\tilde{\mathbf{g}}} &= \langle \tilde{\mathbf{g}}_{k+1}, \tilde{\mathbf{g}}_{k+1} \rangle - \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle \\ &= (\lambda_f^2 - 1) \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle - 2\lambda_f^2 (1 + \delta_k) \alpha_k \langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k \rangle + \lambda_f^2 (1 + \delta_k)^2 \alpha_k^2 \langle \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k \rangle \end{aligned}$$

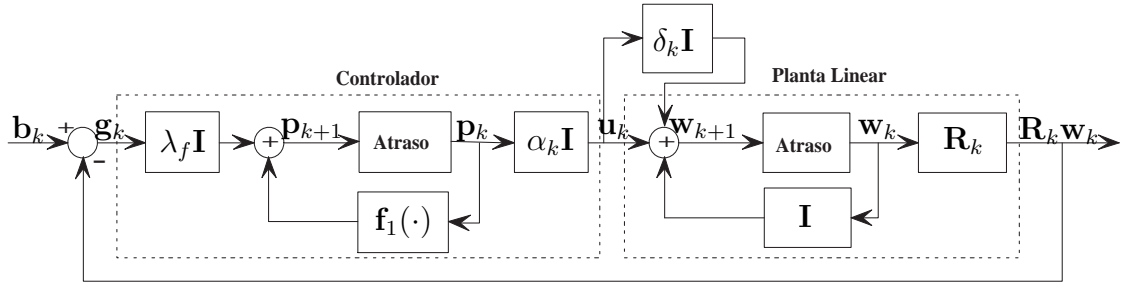


Figura 5.10: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa (2.7) pelo sistema dinâmico (5.56) correspondente ao método CG-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

Inserindo α_k dado por (2.35) na fórmula de $\Delta V_{\tilde{\mathbf{g}}}$ resulta em

$$\begin{aligned} \Delta V_{\tilde{\mathbf{g}}} &= (\lambda_f^2 - 1) \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle - (2\lambda_f^2(1 + \delta_k) - \lambda_f^2(1 + \delta_k)^2) \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k \rangle^2}{\langle \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k \rangle} \\ &= (\lambda_f^2 - 1) \langle \tilde{\mathbf{g}}_k, \tilde{\mathbf{g}}_k \rangle + \lambda_f^2 (\delta_k^2 - 1) \frac{\langle \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k \rangle^2}{\langle \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k, \mathbf{R}_{k+1} \tilde{\mathbf{g}}_k \rangle} \end{aligned}$$

Portanto, para $\lambda_f < 1$ tem-se que

$$\Delta V_{\tilde{\mathbf{g}}} < 0 \Leftrightarrow \delta_k^2 - 1 < 0 \Leftrightarrow \delta_k^2 < 1 \Leftrightarrow |\delta_k| < 1. \quad (5.55)$$

Então o resíduo $\tilde{\mathbf{g}}_k$ decresce na norma-2 se e somente se $|\delta_k| < 1$. ■

5.4.4 Método CG-CLF para filtros adaptativos

Em aritmética de precisão finita, o método CG-CLF para filtros adaptativos dado pelo sistema (2.50) é representado pela figura 5.10 e pelo sistema seguinte

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + (1 + \delta_k) \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - (1 + \delta_k) \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} e_{k+1}^* \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k \end{cases} \quad (5.56)$$

O corolário 6.3 provê uma condição necessária e suficiente para a estabilidade assintótica do equilíbrio do sistema (5.56).

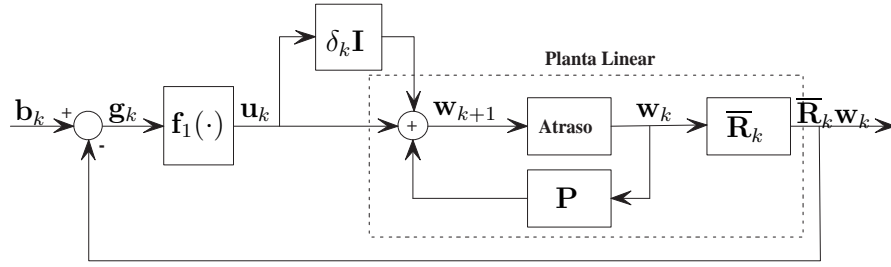


Figura 5.11: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (5.57) correspondente ao método CSD-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

Corolário 6.3. *Sob as hipóteses do teorema 2 e para α_k e β_k dados por (2.51) e (2.52), o equilíbrio do sistema (5.56) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.3. *Escolhendo a CLF $V_{\tilde{\mathbf{g}}_k}(k) = \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle$, a prova segue da mesma forma que a prova do corolário 6.2. ■*

5.4.5 Método CSD-CLF para filtros adaptativos com restrições lineares

O estudo dos efeitos dos erros devidos à aritmética de precisão finita no método CSD-CLF para filtros adaptativos com restrições lineares pode ser feito reescrevendo o sistema dinâmico (2.102) da seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P}\mathbf{w}_k + (1 + \delta_k)\bar{\mathbf{f}} - (1 + \delta_k)\alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - (1 + \delta_k)\alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{e}}_{k+1}^* \end{cases} \quad (5.57)$$

O efeito dos erros de precisão finita sobre a estabilidade assintótica do equilíbrio do sistema (5.57), representado em forma de diagrama de blocos na figura 5.11, é expresso através do seguinte corolário:

Corolário 6.4. *Sob as hipóteses do corolário 1.1 e para α_k dado por (2.103), o equilíbrio do sistema (5.57) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.4. *Substituindo \mathbf{R}_{k+1} por $\bar{\mathbf{R}}_{k+1}$, a prova segue da mesma forma que a prova do corolário 6.2. ■*

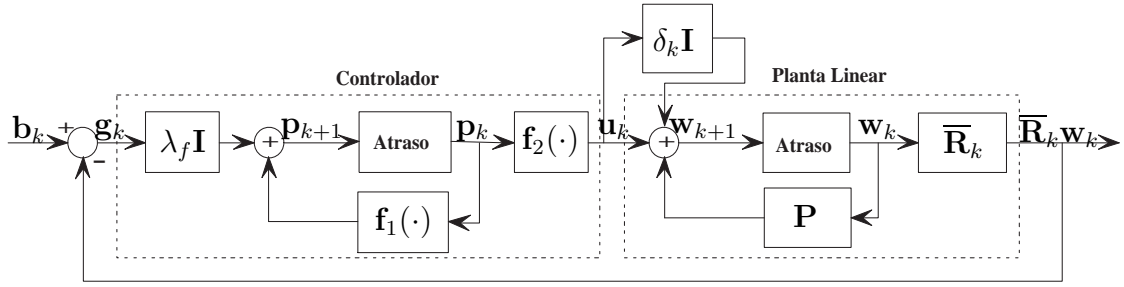


Figura 5.12: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (2.8) pelo sistema dinâmico (5.58) correspondente ao método CCG-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

5.4.6 Método CCG-CLF para filtros adaptativos com restrições lineares

Para estudar os efeitos dos erros devidos à aritmética de precisão finita sobre o método CCG-CLF para filtros adaptativos com restrições lineares, reescreve-se o sistema (2.110) da seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P}\mathbf{w}_k + (1 + \delta_k)\bar{\mathbf{f}} - (1 + \delta_k)\alpha_k\mathbf{P}\mathbf{p}_k \\ \mathbf{g}_{k+1} = \lambda_f\mathbf{g}_k - (1 + \delta_k)\alpha_k\bar{\mathbf{R}}_{k+1}\mathbf{p}_k + \bar{\mathbf{x}}_{k+1}\bar{\mathbf{e}}_{k+1}^* \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k\mathbf{p}_k, \end{cases} \quad (5.58)$$

A condição para a estabilidade assintótica do sistema (5.58), representado em forma de diagrama de blocos na figura 5.12, na presença da perturbação δ_k é dado pelo seguinte corolário:

Corolário 6.5. *Sob as hipóteses do corolário 2.1 e para α_k e β_k dados por (2.111) e (2.112), o equilíbrio do sistema (5.58) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.5. *Substituindo \mathbf{R}_{k+1} por $\bar{\mathbf{R}}_{k+1}$ e escolhendo a CLF $V_{\tilde{\mathbf{g}}_k}(k) = \langle \tilde{\mathbf{g}}_k, \bar{\mathbf{R}}_k^{-1}\tilde{\mathbf{g}}_k \rangle$, a prova segue da mesma forma que a prova do corolário 6.2. ■*

5.4.7 Método SDCLF-CMA para filtros adaptativos cegos

O estudo dos efeitos dos erros devidos à aritmética de precisão finita no método SDCLF-CMA para filtros adaptativos cegos pode ser feito reescrevendo o sistema dinâmico (3.34) da

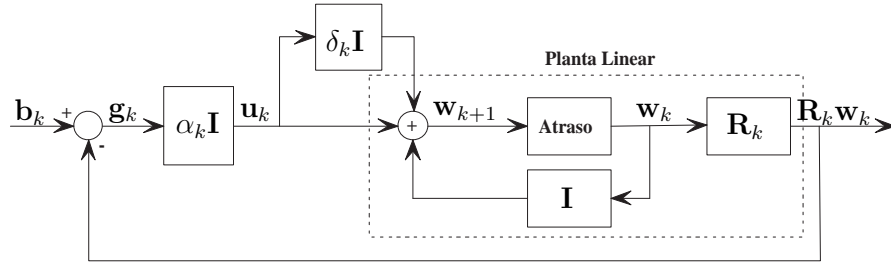


Figura 5.13: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (5.59) correspondente ao método SDCLF-CMA em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + (1 + \delta_k)\alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - (1 + \delta_k)\alpha_k \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{z}_{k+1} e_{k+1}^* \end{cases} \quad (5.59)$$

O efeito dos erros de precisão finita sobre a estabilidade assintótica do equilíbrio do sistema (5.59), representado em forma de diagrama de blocos na figura 5.13, é expressado através do corolário seguinte:

Corolário 6.6. *Sob a hipótese do corolário 1.2 e para α_k dado por (3.35), o equilíbrio do sistema (5.59) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.6. *Determinando \mathbf{R}_{k+1} e \mathbf{z}_{k+1} através de (3.14) e (3.31), a prova segue da mesma forma que prova do corolário 6.2. ■*

5.4.8 Método CGCLF-CMA para filtros adaptativos cegos

Para estudar os efeitos dos erros devidos à aritmética de precisão finita sobre o método CGCLF-CMA para filtros adaptativos cegos, reescreve-se o sistema (3.26) da seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + (1 + \delta_k)\alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \lambda \mathbf{g}_k - (1 + \delta_k)\alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{z}_{k+1} e_{k+1}^* \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k, \end{cases} \quad (5.60)$$

A condição para a estabilidade assintótica do sistema (5.60), representado em forma de diagrama de blocos na figura 5.14, na presença da perturbação δ_k é dado pelo corolário seguinte

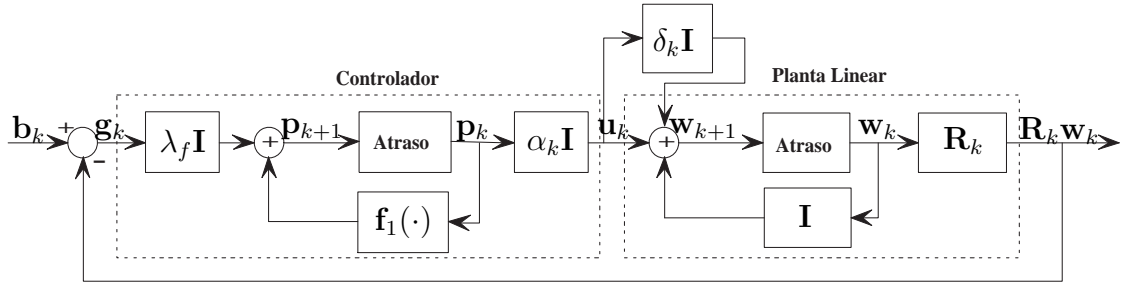


Figura 5.14: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa cega (3.17) pelo sistema dinâmico (5.60) correspondente ao método CGCLF-CMA em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

Corolário 6.7. *Sob as hipóteses do corolário 2.2 e para α_k e β_k dados por (3.27) e (3.28), o equilíbrio do sistema (5.60) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.7. *Determinando \mathbf{R}_{k+1} e \mathbf{z}_{k+1} através de (3.14) e (3.13), e escolhendo a CLF $V_{\tilde{\mathbf{g}}_k}(k) = \langle \tilde{\mathbf{g}}_k, \mathbf{R}_k^{-1} \tilde{\mathbf{g}}_k \rangle$, a prova segue da mesma forma que prova do corolário 6.2. ■*

5.4.9 Método CSDCLF-CMA para filtros adaptativos cegos com restrições lineares

Os efeitos dos erros devidos à aritmética de precisão finita no método SDCLF-CMA para filtros adaptativos cegos são estudados reescrevendo o sistema dinâmico (3.66) da seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P}\mathbf{w}_k + (1 + \delta_k)\bar{\mathbf{f}} - (1 + \delta_k)\alpha_k \mathbf{g}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - (1 + \delta_k)\alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k + \bar{\mathbf{z}}_{k+1} \bar{\mathbf{e}}_{k+1}^* \end{cases} \quad (5.61)$$

O efeito dos erros de precisão finita sobre a estabilidade assintótica do equilíbrio do sistema (5.61), representado em forma de diagrama de blocos na figura 5.15, é expressado através do corolário seguinte:

Corolário 6.8. *Sob as hipóteses do corolário 1.3 e para α_k dado por (3.67), o equilíbrio do sistema (5.61) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.8. *Substituindo \mathbf{R}_{k+1} e \mathbf{x}_{k+1} por $\bar{\mathbf{R}}_{k+1}$ e $\bar{\mathbf{z}}_{k+1}$, a prova segue da mesma forma que a prova do corolário 6.2. ■*

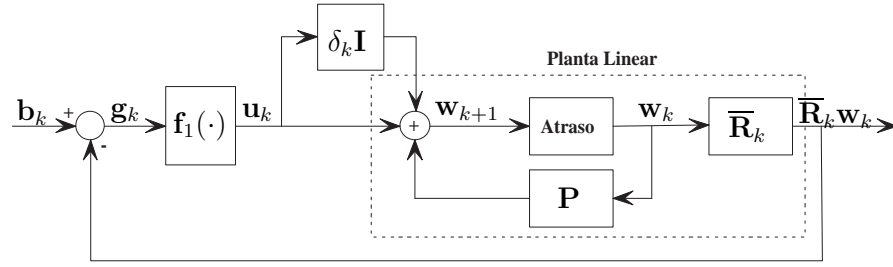


Figura 5.15: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.5) pelo sistema dinâmico (5.61) correspondente ao método CSDCLF-CMA em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

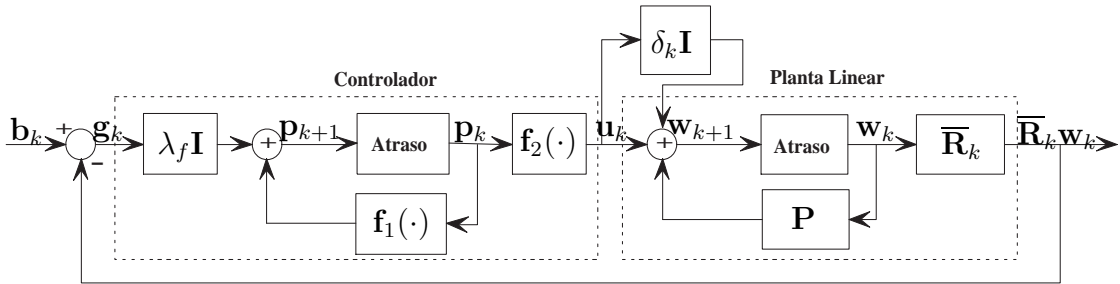


Figura 5.16: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução do problema de filtragem adaptativa com restrições lineares (3.5) pelo sistema dinâmico (5.62) correspondente ao método CCGCLF-CMA em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

5.4.10 Método CCGCLF-CMA para filtros adaptativos cegos com restrições lineares

Para estudar os efeitos dos erros devidos à aritmética de precisão finita sobre o método CCGCLF-CMA para filtros adaptativos cegos com restrições lineares, reescreve-se o sistema (3.59) da seguinte forma:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P}\mathbf{w}_k + (1 + \delta_k)\bar{\mathbf{f}} - (1 + \delta_k)\alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - (1 + \delta_k)\alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k + \bar{\mathbf{z}}_{k+1} \bar{\mathbf{e}}_{k+1}^* \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k. \end{cases} \quad (5.62)$$

A condição para a estabilidade assintótica do sistema (5.62), representado em forma de diagrama de blocos na figura 5.16, na presença da perturbação δ_k é dado pelo corolário seguinte

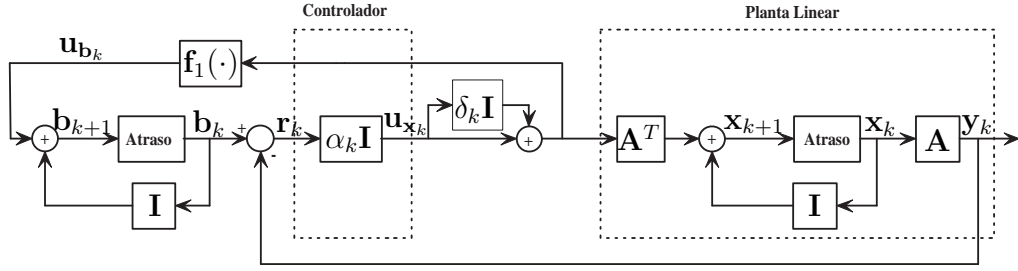


Figura 5.17: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.6) pelo sistema dinâmico (5.63) correspondente ao método SDP-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

Corolário 6.9. *Sob a hipótese do corolário 2.3 e para α_k e β_k dados por (3.60) e (3.61), o equilíbrio do sistema (5.62) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.9. *Substituindo \mathbf{R}_{k+1} e \mathbf{x}_{K+1} por $\bar{\mathbf{R}}_{k+1}$ e $\bar{\mathbf{z}}_{k+1}$, e escolhendo a CLF $V_{\tilde{\mathbf{g}}_k}(k) = \langle \tilde{\mathbf{g}}_k, \bar{\mathbf{R}}_k^{-1} \tilde{\mathbf{g}}_k \rangle$, a prova segue da mesma forma que a prova do corolário 6.2. ■*

5.4.11 Método SDP-CLF para o treinamento de perceptrons

O estudo dos efeitos dos erros devidos à aritmética de precisão finita no método SDP-CLF para o treinamento de perceptrons pode ser feito reescrevendo o sistema dinâmico (4.30) da seguinte forma:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + (1 + \delta_k) \alpha_k \mathbf{A}^T \mathbf{r}_k \\ \mathbf{b}_{k+1} = \mathbf{b}_k + (1 + \delta_k) \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k) \\ \mathbf{r}_{k+1} = \mathbf{r}_k - (1 + \delta_k) \alpha_k \mathbf{r}'_k \end{cases} \quad (5.63)$$

O efeito dos erros de precisão finita sobre a estabilidade assintótica do equilíbrio do sistema (5.63), representado em forma de diagrama de blocos na figura 5.17, é expressado através do corolário seguinte:

Corolário 6.10. *Sob a hipótese do teorema 3 e para α_k dado por (4.33), o equilíbrio do sistema (5.63) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.10. *A prova segue da mesma forma que a prova do teorema 6. ■*

5.4.12 Método CGP-CLF para o treinamento de perceptrons

Os efeitos dos erros devidos à aritmética de precisão finita no método CGP-CLF para o treinamento de perceptrons são estudados reescrevendo o sistema dinâmico (4.45) da

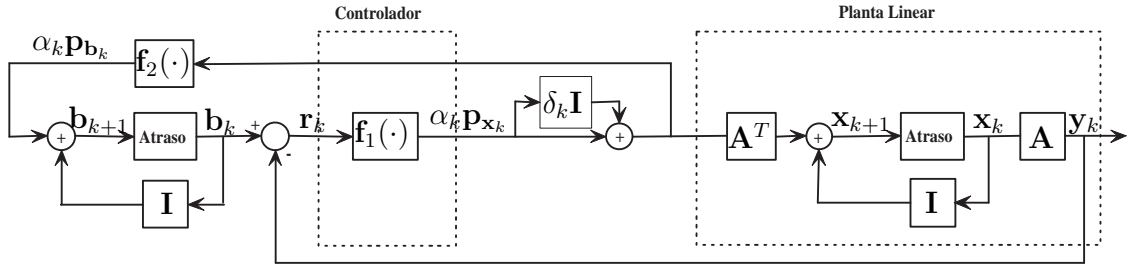


Figura 5.18: Diagrama de blocos ilustrativo do problema de rastreamento equivalente à resolução de (4.6) pelo sistema dinâmico (5.64) correspondente ao método CGP-CLF em aritmética de precisão finita. A perturbação δ_k representa os erros de precisão finita.

seguinte forma:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + (1 + \delta_k)\alpha_k \mathbf{A}^T \mathbf{p}_{\mathbf{x}_k} \\ \mathbf{b}_{k+1} = \mathbf{b}_k + (1 + \delta_k)\alpha_k \mathbf{p}_{\mathbf{b}_k} \\ \mathbf{r}_{k+1} = \mathbf{r}_k - (1 + \delta_k)\alpha_k \mathbf{p}_{\mathbf{r}_k} \\ \mathbf{p}_{\mathbf{x}_{k+1}} = \beta_k \mathbf{p}_{\mathbf{x}_k} + \mathbf{A}^T \mathbf{r}_{k+1} \\ \mathbf{p}_{\mathbf{b}_{k+1}} = \beta_k \mathbf{p}_{\mathbf{b}_k} + |\mathbf{r}_{k+1}| - \mathbf{r}_{k+1} \\ \mathbf{p}_{\mathbf{r}_{k+1}} = \beta_k \mathbf{p}_{\mathbf{r}_k} + \mathbf{r}'_{k+1}. \end{cases} \quad (5.64)$$

O efeito dos erros de precisão finita sobre a estabilidade assintótica do equilíbrio do sistema (5.64), representado em forma de diagrama de blocos na figura 5.18, é expressado através do corolário seguinte:

Corolário 6.11. *Sob a hipótese do teorema 4 e para α_k e β_k dados por (4.47) e (4.48), o equilíbrio do sistema (5.64) é assintoticamente estável se e somente se $|\delta_k| < 1$.*

Prova do Corolário 6.11. *A prova segue da mesma forma que a prova do teorema 6. ■*

5.5 Simulações Numéricas

Para ilustrar a robustez dos métodos as simulações numéricas dos capítulos 2, 3 e 4 serão refeitas considerando uma perturbação δ_k . Além disso, para a resolução do sistema linear (2.9), as simulações numéricas feitas em [11] também serão novamente realizadas considerando a perturbação δ conforme mostrado nas figuras 5.3-5.8.

5.5.1 Sistemas lineares

Para ilustrar a robustez dos métodos de resolução do sistema linear (2.9), consideram-se as matrizes de teste utilizadas em Concus e Saylor [117]. Estas matrizes são Toeplitz simétricas positivas definidas. Os elementos das matrizes são dados por

$$\mathbf{A}_1 : \quad a_k = \frac{1}{k+1}, \quad k = 0, 1, \dots, n-1 \quad (5.65)$$

$$\mathbf{A}_2 : \quad a_k = \left(\frac{1}{2}\right)^k, \quad k = 0, 1, \dots, n-1 \quad (5.66)$$

$$\mathbf{A}_3 : \quad a_k = \frac{\cos k}{k+1}, \quad k = 0, 1, \dots, n-1 \quad (5.67)$$

$$\mathbf{A}_4 : \quad a_0 = 2 + \frac{1}{n^2}, a_1 = -1, a_2 = a_3 = \dots = a_{n-1} = 0. \quad (5.68)$$

Os elementos da diagonal são multiplicados por 20 para tornar as matrizes com dominância diagonal e fazer uma comparação do desempenho dos métodos Jacobi, Gauss-Seidel, SD-CLF e CG-CLF.

Para os métodos de Jacobi e Gauss-Seidel, determina-se a margem δ a partir das equações (5.25) e (5.40) respectivamente para cada matriz \mathbf{A} . Para os métodos SD-CLF e CG-CLF a margem δ é dada pela equação (5.44). Em seguida aplica-se uma perturbação $\delta_a = \delta - 0.001$ e determina-se o número de iterações necessário para alcançar o critério de parada escolhido tal que $\|\mathbf{r}_k\| \leq 0.001$ ou o número máximo de iterações igual 10000. Determina-se δ_M , o menor valor para o qual cada algoritmo diverge e a porcentagem de conservadorismo definido como

$$\%C = 100 \frac{\delta_M - \delta}{\delta_M}.$$

A tabela 5.1 mostra o desempenho do método iterativo apresentado no algoritmo 5.1. Nota-se que os métodos dinâmicos apresentam uma medida de robustez maior que os métodos estacionários porém, os últimos apresentam uma taxa de convergência maior na presença da perturbação máxima tolerável, conforme pode ser visto no caso da matriz \mathbf{A}_2 na figura 5.19. Nota-se também que no caso dos métodos estacionários (Jacobi, Gauss-Seidel) a porcentagem de conservadorismo da margem de estabilidade fornecida pelo teorema do pequeno ganho é relativamente pequena para as matrizes \mathbf{A}_2 e \mathbf{A}_4 .

Matriz	Método	δ_a	Iterações	$\ \mathbf{r}_k\ $	δ_M	%C
A_1	Jacobi	0.284	6009	9.996×10^{-4}	0.286	0.35
	Gau-Sei	0.645	139	9.552×10^{-4}	0.713	9.40
	SD-CLF	0.999	10000	1.429×10^{-3}	1.001	0.10
	CG-CLF	0.999	10000	2.125×10^{-1}	1.001	0.10
A_2	Jacobi	0.817	7877	9.994×10^{-4}	0.819	0.12
	Gau-Sei	0.908	5543	9.945×10^{-4}	0.910	0.11
	SD-CLF	0.999	10000	1.429×10^{-3}	1.001	0.10
	CG-CLF	0.999	10000	2.125×10^{-1}	1.001	0.10
A_3	Jacobi	0.711	3837	9.997×10^{-4}	0.713	0.14
	Gau-Sei	0.867	320	9.732×10^{-4}	0.887	2.14
	SD-CLF	0.999	5000	1.429×10^{-3}	1.001	0.10
	CG-CLF	0.999	10000	2.125×10^{-1}	1.001	0.10
A_4	Jacobi	0.904	1749	9.999×10^{-4}	0.906	0.11
	Gau-Sei	0.951	4209	9.985×10^{-4}	0.953	0.11
	SD-CLF	0.999	10000	1.429×10^{-3}	1.001	0.10
	CG-CLF	0.999	10000	2.125×10^{-1}	1.001	0.10

Tabela 5.1: Comparação do desempenho dos métodos Jacobi, Gauss-Seidel, SD-CLF e CG-CLF. A perturbação δ representa os erros de precisão finita. A tolerância é escolhida como $\epsilon = 10^{-3}$, a dimensão das matrizes é $n = 1000$ e o número máximo de iterações é igual a 10000. Os métodos dinâmicos apresentam uma margem de estabilidade maior que os métodos estacionários porém, os últimos apresentam uma taxa de convergência maior na presença da perturbação máxima tolerável.

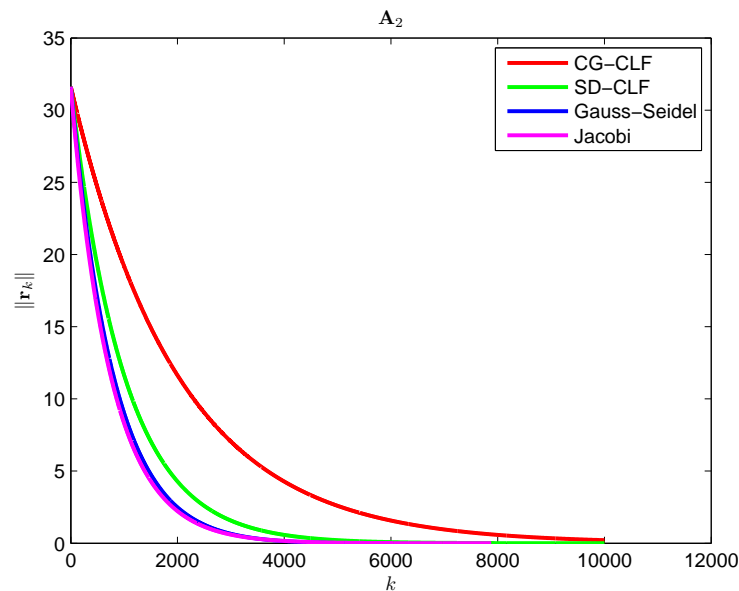


Figura 5.19: Comparação da taxa de convergência dos métodos iterativos lineares para a matriz A_2 na presença da perturbação máxima. Os métodos dinâmicos apresentam uma margem de estabilidade maior que os métodos estacionários porém, os últimos apresentam uma taxa de convergência maior na presença da perturbação máxima tolerável.

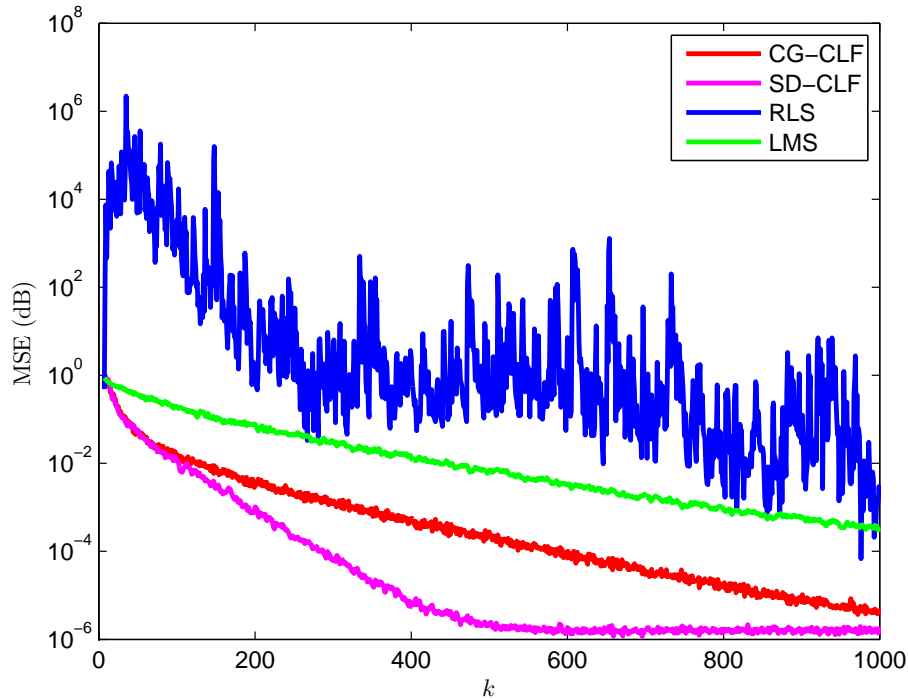


Figura 5.20: Curvas de aprendizado dos equalizadores adaptativos com $W = 2.9$. A tabela 5.2 apresenta os valores de δ_M , o maior valor da perturbação δ_k para o qual o erro médio quadrático de cada algoritmo continua convergindo mais para os valores de regime da tabela 2.1.

Método	δ_M
SD-CLF	0.99
CG-CLF	0.99
RLS	0.70
LMS	0.01

Tabela 5.2: Para os métodos SD-CLF, CG-CLF, LMS e RLS compara-se o valor de δ_M , o maior valor da perturbação δ_k para o qual o erro médio quadrático de cada algoritmo continua convergindo para os valores de regime da tabela 2.1.

5.5.2 Filtros adaptativos

Para ilustrar a robustez dos métodos de filtragem adaptativa, repete-se o experimento de equalização adaptativa da seção 2.6.1 para o caso $W = 2.9$. Uma perturbação δ_k é introduzida na entrada da planta e é propagado até a entrada do controlador correspondente a cada método. O desempenho dos métodos obtidos a partir de CLF's são comparados com os métodos RLS e LMS. A figura 5.20 apresenta as curvas de aprendizado enquanto a tabela 5.2 os valores de δ_M , o maior valor da perturbação δ_k para o qual o erro médio quadrático de

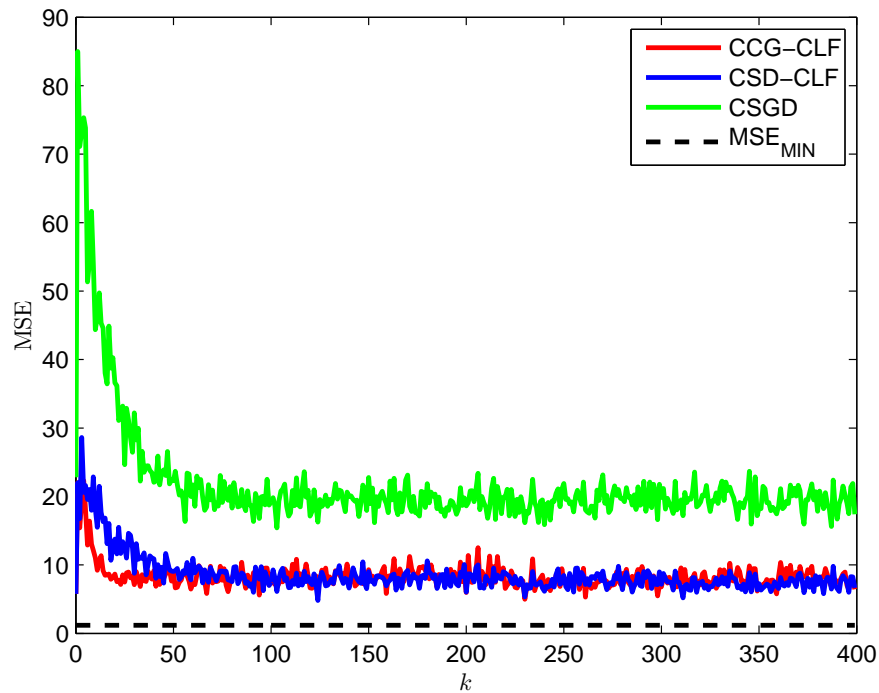


Figura 5.21: Curvas de aprendizado dos filtros adaptativos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 3 usuários, na presença da perturbação $\delta_k = 0.99$. Os métodos CCG-CLF e CSD-CLF apresentam um erro médio quadrático menor em regime enquanto o método CRLS diverge.

cada algoritmo continua convergindo para os valores de regime da tabela 2.1. Nota-se que os métodos desenvolvidos via CLF apresentam uma medida de robustez maior que os métodos LMS e RLS. A curva de aprendizado do método RLS mostra que, além de apresentar uma medida de robustez menor que os métodos CLF's, o método RLS apresenta um erro médio quadrático consideravelmente oscilatório e com uma taxa de convergência menor. O método LMS é consideravelmente sensível à perturbação, uma perturbação $\delta_k = 0.01$ é suficiente para alterar o valor de regime do erro médio quadrático. Embora os dois métodos SD-CLF e CG-CLF ainda convergem para os valores de regime da tabela 2.1 na presença da perturbação δ_k , nota-se que o método SD-CLF é menos afetado pela perturbação, e apresenta uma taxa de convergência maior.

Método	δ_M
SDCLF-CMA	0.99
CGCLF-CMA	0.99
RLS-CMA	0.89
NLSD-CMA	0.19

Tabela 5.3: Para os métodos SDCLF-CMA, CGCLF-CMA, RLS-CMA e NLSD-CMA compara-se o valor de δ_M , o maior valor da perturbação δ_k para o qual o erro médio quadrático de cada algoritmo converge para os valores de regime da figura 3.14.

5.5.3 Filtros adaptativos com restrições lineares

Para ilustrar a robustez dos métodos de filtragem adaptativa com restrições lineares, repete-se o experimento de simulação de um sistema de comunicação DS-CDMA da seção 2.6.4. Novamente uma perturbação δ_k é introduzida na entrada da planta e propagada até a entrada do controlador correspondente a cada método. Nesse caso note que a perturbação atua também nas restrições de tal forma que qualquer solução \mathbf{w}_k de (2.8) será tal que $\mathbf{C}^H \mathbf{w}_k = (1 + \delta_k)\mathbf{f}$. Por outro lado, a cada iteração o vetor de pesos \mathbf{w}_k é projetado no espaço nulo a direita da matriz das restrições \mathbf{C} pela matriz de projeção \mathbf{P} . Portanto, nesse caso, os efeitos da perturbação δ causarão um deslocamento da solução em relação à solução ótima, antes de resultar em uma divergência dos métodos. Então para $\delta_k = 0.99$ a figura 5.21 mostra as curvas de aprendizado dos métodos. Pode-se notar que os métodos CLF's convergem para um erro médio quadrático menor em regime. Para o método CRLS, devido às recorrências das matrizes $\mathbf{\Gamma}_{k+1}$, $\mathbf{\Psi}_{k+1}^{-1}$ e \mathbf{R}_{k+1}^{-1} , uma perturbação $\delta_k = 0.01$ é suficiente para causar a divergência do método.

5.5.4 Filtros adaptativos cegos

A robustez dos métodos de filtragem adaptativa cega é ilustrada repetindo o experimento do sistema de antenas adaptativas da seção 3.6.1 e introduzindo uma perturbação δ_k na entrada da planta. A tabela 5.3 mostra o valor δ_M , o maior valor da perturbação δ_k para o qual o erro médio quadrático de cada método continua convergindo para os valores de regime da figura 3.14. As figuras 5.22-5.25 apresentam o desempenho dos métodos na presença da perturbação δ_M . Nota-se que os métodos desenvolvidos apresentam uma robustez maior. Embora o método RLS-CMA apresenta uma medida de robustez maior que o método NLSD-CMA, o último mostra um comportamento menos oscilatório na presença da perturbação.

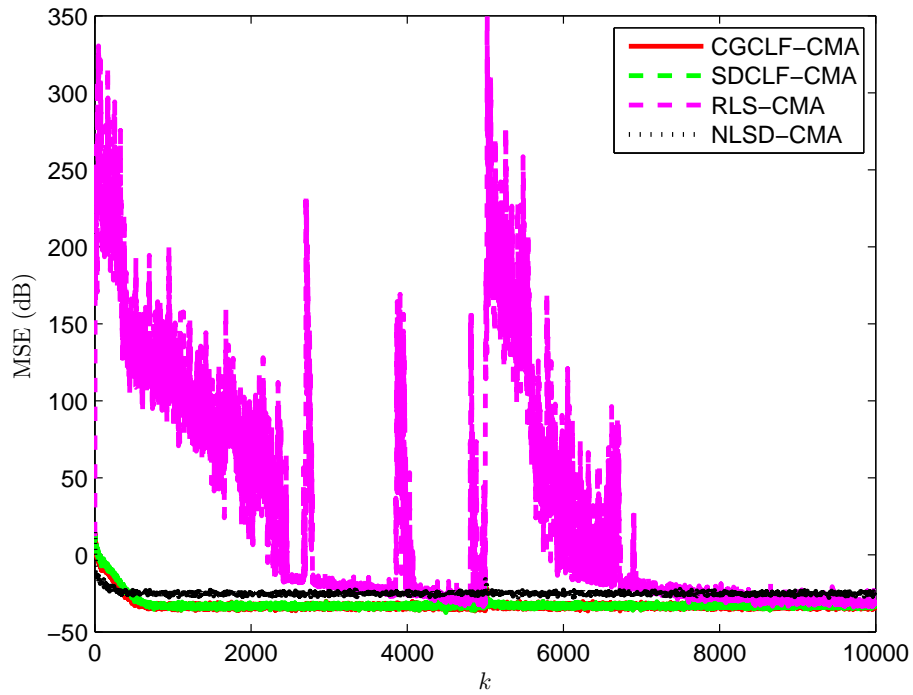


Figura 5.22: Erro médio quadrático dos métodos SDCLF-CMA, CGCLF-CMA, NLSD-CMA e RLS-CMA, na presença da perturbação δ_M

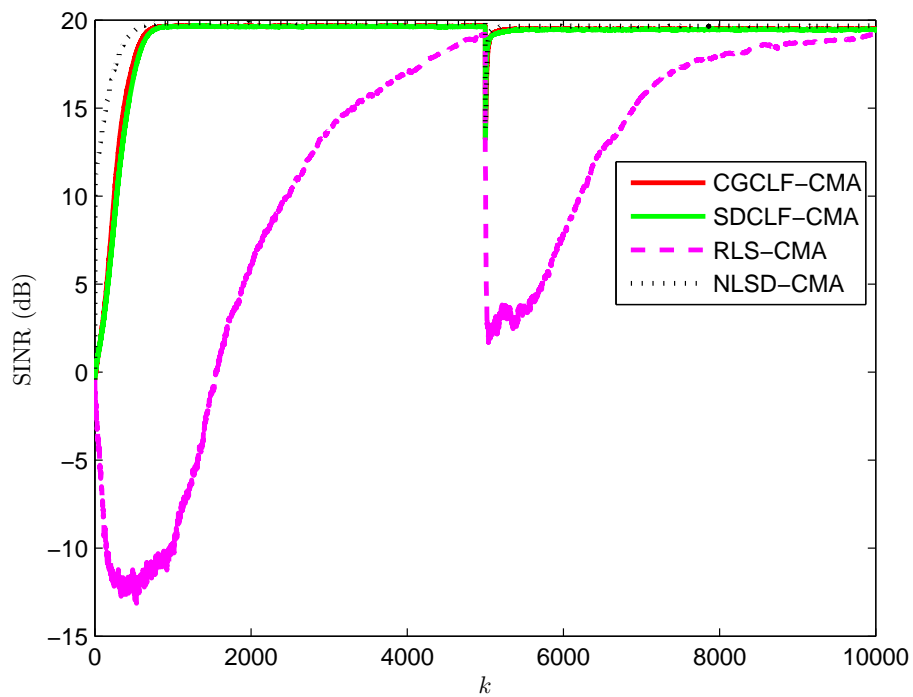


Figura 5.23: Quociente entre o sinal e a interferência mais o ruído dos métodos CGCLF-CMA, RLS-CMA, SGD-CMA e NLSD-CMA na presença da perturbação δ_M .

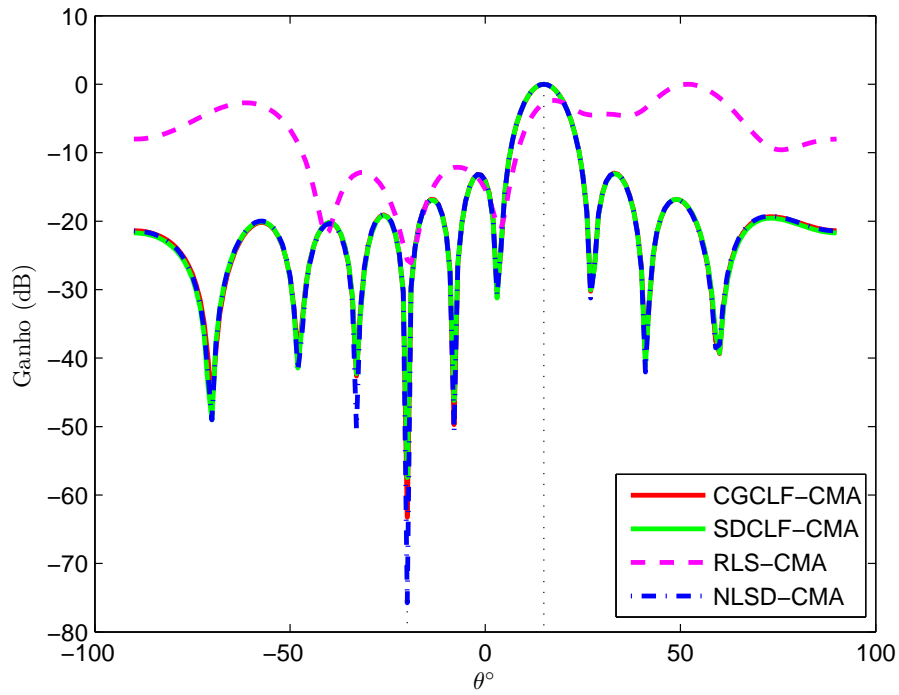


Figura 5.24: *Beampattern* normalizado dos métodos SDCLF-CMA, CGCLF-CMA, RLS-CMA e NLSD-CMA obtido na iteração 5000, na presença da perturbação δ_M .

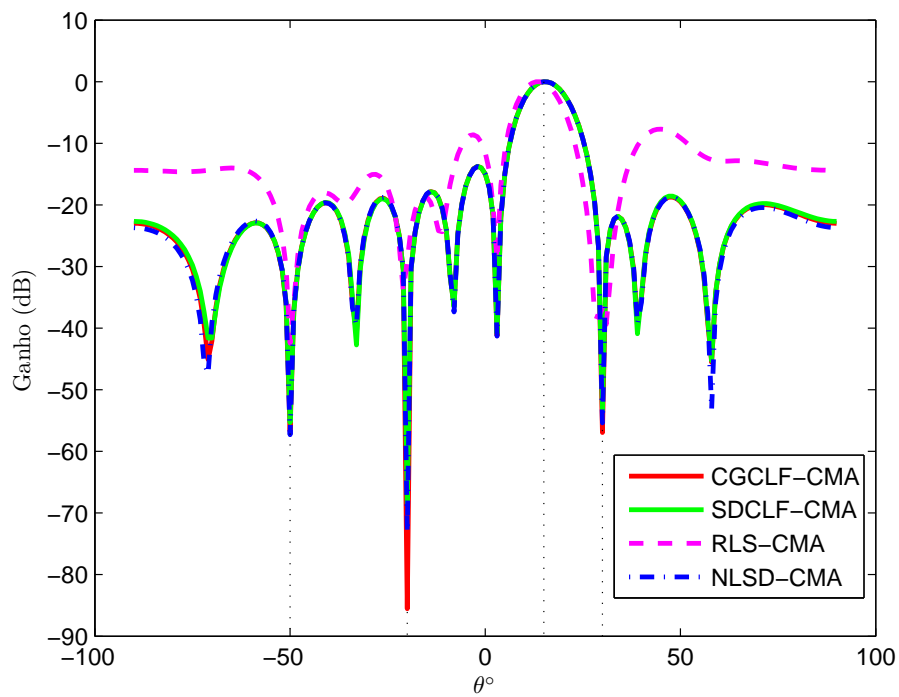


Figura 5.25: *Beampattern* normalizado dos métodos SDCLF-CMA, CGCLF-CMA, RLS-CMA e NLSD-CMA obtido na iteração 10000, na presença da perturbação δ_M .

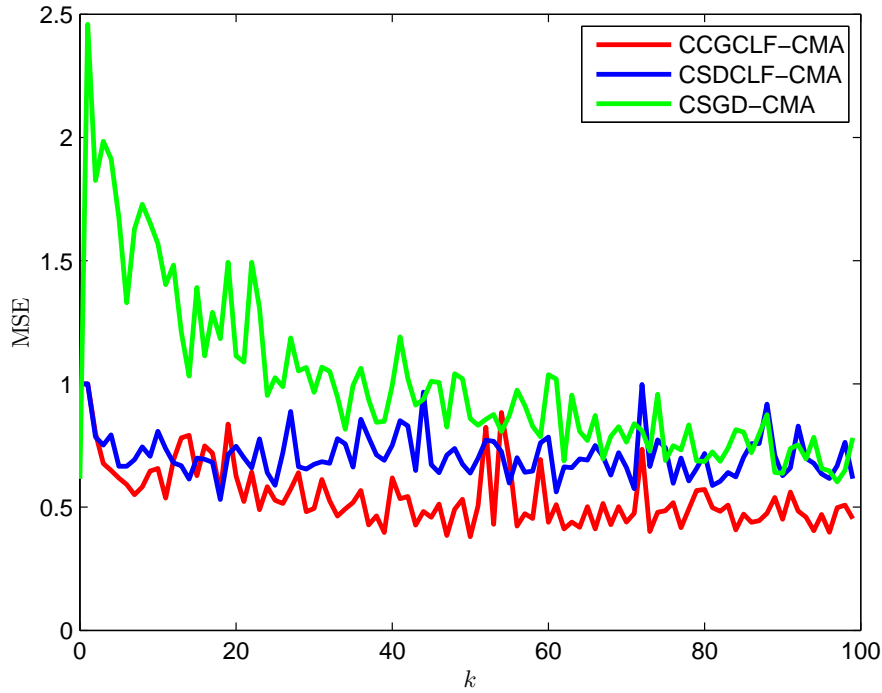


Figura 5.26: Curvas de aprendizado dos filtros adaptativos cegos com restrições lineares aplicados ao problema de detecção mono-usuário em um sistema de comunicação móvel DS-CDMA com 5 usuários, na presença da perturbação $\delta_k = 0.99$. Os métodos CCGCLF-CMA e CSDCLF-CMA apresentam um erro médio quadrático menor em regime enquanto o método CRLS-CMA diverge.

5.5.5 Filtros adaptativos cegos com restrições lineares

A robustez dos métodos de filtragem adaptativa cega com restrições lineares é ilustrada realizando novamente o experimento numérico da seção 3.6.2. Nesse caso introduz-se uma perturbação δ_k na entrada da planta. Como no caso dos filtros adaptativos com restrições lineares, note que a perturbação atua também nas restrições de tal forma que qualquer solução \mathbf{w}_k de (3.5) será tal que $\mathbf{C}^H \mathbf{w}_k = (1 + \delta_k) \mathbf{f}$. Por outro lado, a cada iteração o vetor de pesos \mathbf{w}_k é projetado no espaço nulo a direita da matriz das restrições \mathbf{C} pela matriz de projeção \mathbf{P} . Portanto, analogamente ao caso dos filtros adaptativos com restrições lineares, os efeitos da perturbação δ_k causarão um deslocamento da solução em relação à solução ótima, antes de resultar em uma divergência dos métodos. Então para $\delta_k = 0.99$, a figura 5.26 mostra as curvas de aprendizado dos métodos. Novamente, pode-se notar que os métodos CLF's convergem para um erro médio quadrático menor em regime. Para o método CRLS-CMA, devido às recorrências das matrizes $\mathbf{\Gamma}_{k+1}$, $\mathbf{\Psi}_{k+1}^{-1}$ e \mathbf{R}_{k+1}^{-1} , uma perturbação $\delta_k = 0.01$ é suficiente para causar a divergência do método.

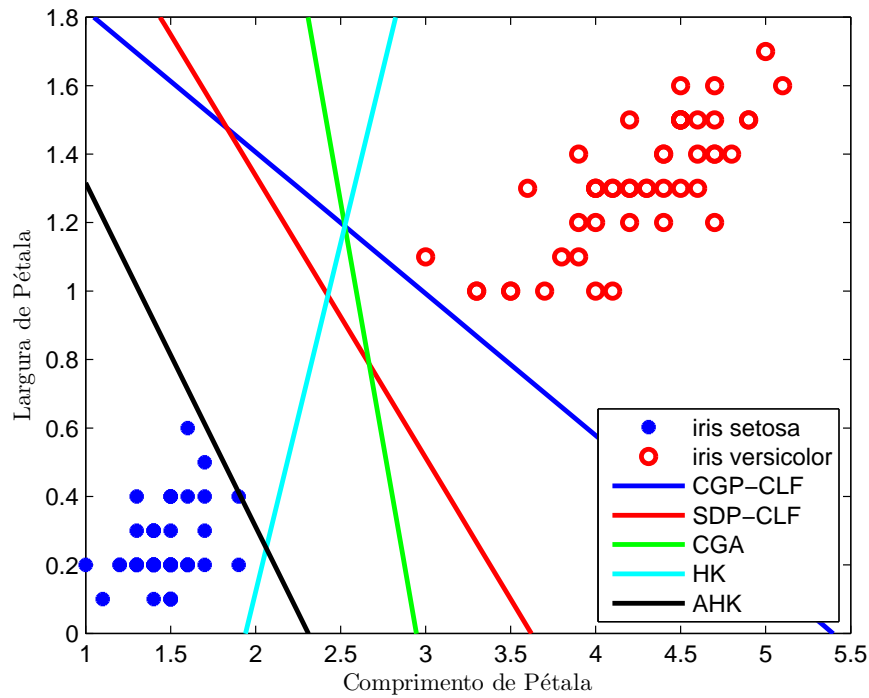


Figura 5.27: Superfícies separadoras das classes *iris setosa* and *iris versicolor* na presença da perturbação δ_k . O critério de parada é $\|\mathbf{r}_k\| \leq 0.001$ e o número máximo de iterações é 2000.

Método	C_1	C_2
SDP-CLF	0.99	0.99
CGP-CLF	0.99	0.99
CGA	0.99	0.99
HK	0.99	0.99
AHK	0.87	0.23

Tabela 5.4: Comparação dos valores de δ_M , para os casos C_1 (separação dos conjuntos *iris setosa* e *iris versicolor*) e C_2 (separação dos conjuntos *iris setosa* e *iris virginica*).

5.5.6 Treinamento de perceptrons

Para ilustrar a robustez dos métodos de treinamento de perceptrons, repetem-se as simulações numéricas da seção 4.4, introduzindo uma perturbação δ_k na entrada da planta. A tabela 5.4 mostra uma comparação de δ_M , o maior valor da perturbação δ_k para o qual os métodos separam os conjuntos *iris setosa* e *iris versicolor* (caso C_1), e os conjuntos *iris setosa* e *iris virginica* (caso C_2). As figuras 5.27 e 5.28 mostram as superfícies separadoras encontradas. Pode-se notar que o método SDP-CLF apresenta uma margem maior enquanto o método AHK apresenta uma medida de robustez menor.

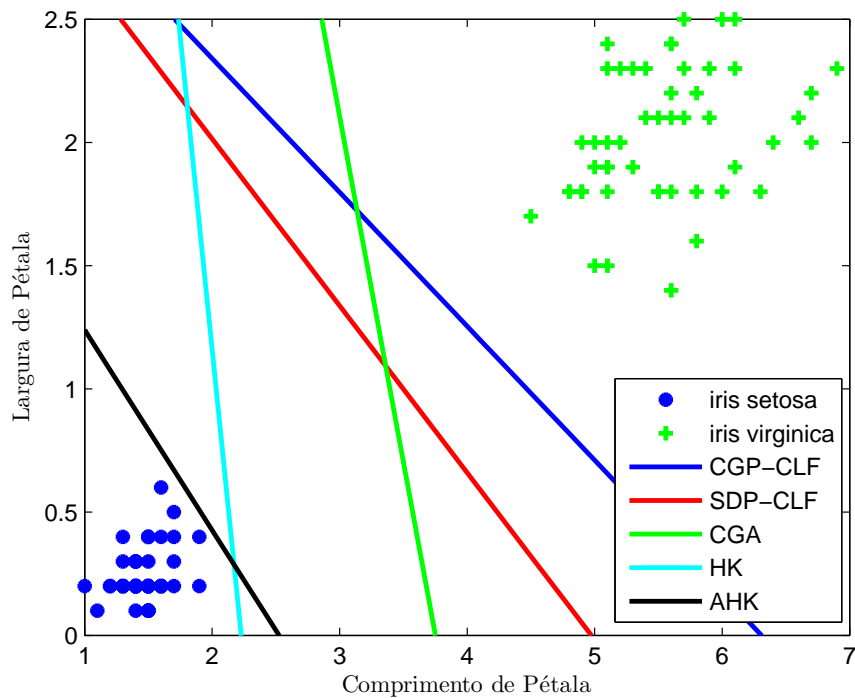


Figura 5.28: Superfícies separadoras das classes *iris setosa* e *iris virginica* na presença da perturbação δ_k . O critério de parada é $\|\mathbf{r}_k\| \leq 0.001$ e o número máximo de iterações é 2000.

5.6 Contribuições e Publicações

Neste capítulo, uma nova modelagem de perturbações oriundas da utilização de aritmética de precisão finita, foi proposta baseando-se na combinação do modelo proposto em [41, 42] com a modelagem usual de perturbações multiplicativas na entrada da planta, introduzida no trabalho de Doyle e Stein [46]. Dessa nova modelagem resultou uma aplicação inédita do teorema do pequeno ganho para determinar medidas de robustez dos métodos de Jacobi e Gauss-Seidel para sistemas lineares. Vale ressaltar que o mesmo pode ser feito para outros métodos de decomposição matricial. Através da utilização de CLFs, foi derivado outro resultado inédito sobre a estabilidade dos métodos numéricos de filtragem adaptativa com ou sem restrições lineares, de filtragem adaptativa cega com ou sem restrições lineares e de treinamento de perceptrons em modo batelada sujeitos a perturbações variantes no tempo. As principais vantagens da nova modelagem proposta são a simplicidade de aplicação a uma gama ampla de algoritmos e a obtenção de medidas quantitativas da robustez dos métodos numéricos. As simulações numéricas mostraram que de uma forma geral, os métodos desenvolvidos a partir de escolhas de funções de Liapunov com controle apresentam uma medida

de robustez maior ou igual que dos outros métodos, e portanto são menos afetados pelos erros de precisão finita.

As contribuições e os resultados apresentados neste capítulo estão sendo preparados para serem submetidos a publicação em periódico internacional.

Capítulo 6

Conclusões, Contribuições e Trabalhos futuros

6.1 Conclusões

Nesta tese foram apresentados métodos numéricos de resolução de problemas de processamento de sinais. Os problemas foram formulados utilizando uma abordagem de sistemas dinâmicos do tipo planta + controlador em malha fechada, introduzida em Bhaya e Kaszkurewicz [10]. Os problemas de processamento de sinais abordados nesta tese foram os de filtragem adaptativa com ou sem restrições lineares, filtragem adaptativa cega com ou sem restrições lineares e treinamento de perceptrons. Os métodos numéricos existentes na literatura para a resolução dos problemas supra-mencionados foram formulados e analisados do ponto de vista de sistemas dinâmicos. Essa formulação permitiu o desenvolvimento de novos métodos numéricos de resolução dos problemas citados. Através de escolhas adequadas de funções de Liapunov com controle foi provado a convergência e a otimalidade dos parâmetros dos métodos propostos.

No caso dos problemas de filtragem adaptativa com ou sem restrições lineares, os métodos SD e CG foram estendidos para ambos os casos através de representação dos métodos iterativos como sistemas dinâmicos, resultando na proposta de duas novas famílias de métodos numéricos para cada um dos dois problemas. Ao contrário dos trabalhos anteriores, os novos métodos propostos no capítulo 2 não utilizam nenhum parâmetro empírico e são reforçadas por provas formais de convergência e de otimalidade dos parâmetros. Verificou-se,

através de simulações numéricas de três cenários de filtragem adaptativa (equalização, identificação de sistema e predição linear) e de um cenário de filtragem adaptativa com restrições lineares (sistema de comunicação DS-CDMA), que os mesmos apresentam um desempenho superior ou comparável aos métodos RLS, CRLS, CG[19] e CCG [20] (a um custo computacional menor) e superior aos métodos LMS e CLMS.

Para os problemas de filtragem adaptativa cega com ou sem restrições lineares, a representação através de sistemas dinâmicos permitiu desenvolver duas novas famílias de métodos para cada um dos dois problemas, estendendo os métodos SD e CG em analogia aos casos de filtragem adaptativa. As principais vantagens dos novos métodos propostos no capítulo 3 em relação aos existentes na literatura são a não-utilização de nenhum parâmetro empírico, a prova formal da convergência e um desempenho superior ou igual, tudo a um custo computacional menor que dos métodos RLS-CMA [70] e CRLS-CMA [81].

No capítulo 4, duas novas famílias de métodos de treinamento de perceptrons em modo batelada foram desenvolvidos através da formulação do problema como sistemas dinâmicos. Verificou-se que em termos de classificação de padrões, o desempenho dos novos métodos propostos é comparável ao dos métodos CGA [36] e LS-SVM [96], e superior ao dos métodos HK [31, 32] e AHK [35]. A convergência dos métodos HK e AHK não é garantida em todos os casos. O primeiro (HK) utiliza a pseudo-inversa da matriz de dados, para determinar um vetor solução que além de ser computacionalmente custoso para problemas de grande escala (i.e. separação de classes com um grande número de atributos), torna-se instável a medida que a matriz $\mathbf{A}^T \mathbf{A}$ se aproxima da singularidade. A convergência do segundo (AHK) depende criticamente da escolha empírica dos parâmetros de aprendizagem ρ_1 e ρ_2 , uma escolha inapropriada dos parâmetros de aprendizagem ρ_1 e ρ_2 resulta em um baixo desempenho do método AHK. Devido às perdas de ortogonalidade dos resíduos, causadas pelos erros de precisão finita, o método CGA de Nagaraja e Bose [36] é baseado em uma técnica heurística que consiste na reinicialização da direção do gradiente [94] para garantir a convergência do método. As vantagens dos métodos propostos, no capítulo 4, são a não-utilização de nenhuma técnica heurística e de nenhum parâmetro empírico, e as provas de convergência obtidas a partir da formulação, adotada ao longo desta tese, dos mesmos como sistemas dinâmicos.

Para a análise da robustez dos métodos numéricos (ou estabilidade numérica), a nova modelagem de perturbações oriundas da utilização de aritmética de precisão finita, proposta

no capítulo 5, baseada na combinação do modelo proposto em [41, 42] com a modelagem usual de perturbações multiplicativas na entrada da planta, resultou na aplicação inédita do teorema do pequeno ganho para determinar medidas de robustez dos métodos de Jacobi e Gauss-Seidel para sistemas lineares. Enfatiza-se que a mesma análise pode ser facilmente estendida para outros métodos iterativos estacionários. Outro resultado inédito obtido no mesmo contexto é a derivação de uma margem de estabilidade dos métodos iterativos dinâmicos sujeitos a perturbações variantes no tempo através da utilização de CLFs. As principais vantagens da nova modelagem proposta são a simplicidade de aplicação a uma gama ampla de algoritmos e a obtenção de medidas quantitativas da robustez dos métodos numéricos. As simulações numéricas mostraram que de uma forma geral, os métodos desenvolvidos a partir de escolhas de funções de Liapunov com controle apresentam uma medida de robustez maior ou igual que dos outros métodos, e portanto são menos afetados pelos erros de precisão finita.

A conclusão geral desta tese, é que a formulação através de sistemas dinâmicos permite resolver vários problemas distintos com uma abordagem única. A consideração dos parâmetros dos algoritmos como parâmetros de controle a serem determinados permite fazer escolhas ótimas, sem o uso de heurísticas, que garantem a convergência através de escolhas apropriadas de funções de Liapunov com controle. A partir das mesmas CLFs é possível determinar uma medida da robustez, ou margem de estabilidade na presença de erros de precisão finita variantes no tempo, dos métodos em consideração.

6.2 Contribuições

As principais contribuições desta tese são enumeradas a seguir

- Contribuições em filtragem adaptativa
 - Formulação e análise dos métodos de filtragem adaptativa como sistemas dinâmicos;
 - Desenvolvimento de duas famílias de novos métodos de filtragem adaptativa;
 - Prova da convergência, sob as hipóteses padrão de filtragem adaptativa, dos métodos de filtragem adaptativa propostos com escolhas ótimas dos parâmetros do filtro de acordo com a escolha de uma função de Liapunov com controle;

- Contribuições em filtragem adaptativa com restrições lineares
 - Formulação e análise dos métodos de filtragem adaptativa com restrições lineares como sistemas dinâmicos;
 - Desenvolvimento de duas famílias de novos métodos de filtragem adaptativa com restrições lineares;
 - Prova da convergência, sob as hipóteses padrão de filtragem adaptativa com restrições lineares, dos métodos de filtragem adaptativa com restrições lineares propostos com escolhas ótimas dos parâmetros do filtro de acordo com a escolha de uma função de Liapunov com controle;

- Contribuições em filtragem adaptativa cega
 - Formulação e análise dos métodos de filtragem adaptativa cega como sistemas dinâmicos;
 - Desenvolvimento de duas famílias de novos métodos de filtragem adaptativa cega;
 - Prova da convergência, sob as hipóteses padrão de filtragem adaptativa cega dos métodos de filtragem adaptativa cega propostos com escolhas ótimas dos parâmetros do filtro de acordo com a escolha de uma função de Liapunov com controle;

- Contribuições em filtragem adaptativa cega com restrições lineares
 - Formulação e análise dos métodos de filtragem adaptativa cega com restrições lineares como sistemas dinâmicos;
 - Desenvolvimento de duas famílias de novos métodos de filtragem adaptativa cega com restrições lineares;
 - Prova da convergência, sob as hipóteses padrão de filtragem adaptativa cega com restrições lineares, dos métodos de filtragem adaptativa cega com restrições lineares propostos com escolhas ótimas dos parâmetros do filtro de acordo com a escolha de uma função de Liapunov com controle;

- Contribuições no treinamento de perceptrons

- Formulação e análise dos métodos de treinamento de perceptrons em modo batelada como sistemas dinâmicos;
 - Desenvolvimento de duas famílias de novos métodos de treinamento de perceptrons em modo batelada;
 - Prova sistemática da convergência dos métodos de treinamento de perceptrons em modo batelada propostos com escolhas ótimas dos parâmetros do algoritmo de acordo com a escolha de uma função de Liapunov com controle;
- Contribuições na análise numérica
 - Modelagem dos erros oriundas da utilização da aritmética de precisão finita como perturbações multiplicativas na entrada da planta;
 - Aplicação do teorema do pequeno ganho na determinação de medidas de robustez dos métodos iterativos estacionários.
 - Aplicação de CLFs na determinação de margens de estabilidade dos métodos iterativos dinâmicos sujeitos a perturbações variantes no tempo.
 - Sistematização de uma forma simples de obter informações quantitativas sobre a robustez de métodos numéricos estacionários e dinâmicos.

Os resultados e as contribuições alcançados nesta tese decorreram na publicação dos seguintes artigos

1. DIENE, O., BHAYA, A., “Adaptive filtering algorithms designed using control Liapunov functions”. *IEEE Signal Processing Letters* v. 4, pp. 224-227, 2006. Correction in vol. 14 (12) (2007) 1047.
2. DIENE, O., BHAYA, A., “Métodos iterativos lineares projetados através de ferramentas da teoria de controle e suas aplicações”. *Revista de Controle e Automação* v. 17, n. 3, pp. 265-277, 2006. URL <http://www.scielo.br/pdf/ca/v17n3/a03v17n3.pdf>.
3. DIENE, O., BHAYA, A., “Linearly constrained adaptive filtering algorithms designed using control Liapunov functions”. In: *Proc. 16th Congresso Brasileiro de Automação*, Salvador, BA, Brazil, October 2006.

4. DIENE, O., BHAYA, A., “Liapunov analysis and design of conjugate gradient constant modulus algorithm applied to blind adaptive array”. *IEEE Transactions on Signal Processing*. Submetido para publicação, 2008.
5. DIENE, O., BHAYA, A., “Algoritmo de Gradiente Conjugado aplicado ao Critério do Módulo Constante: Projeto e Análise através de funções de controle de Liapunov, e aplicações ao processamento não-supervisionada de antenas adaptativas”. *17th Congresso Brasileiro de Automática*, Juiz de Fora, MG, Brasil, Aceito para publicação, 2008.
6. DIENE, O., BHAYA, A., “Perceptron training algorithms designed using discrete-time control Liapunov functions”. *NEUROCOMPUTING*. Aceito para publicação, 2008.
7. DIENE, O., BHAYA, A., “Perceptron training algorithms designed using discrete-time control Liapunov functions”. In: *IEEE Multi-conference on Systems and Control (MSC) 2007*, pp. 608-613, Singapore, October 2007.

6.3 Trabalhos futuros

Nesta tese vários problemas de processamento de sinais foram formulados e analisados como sistemas dinâmicos. A mesma formulação e análise pode ser estendido a vários outros problemas relacionados aos tratados neste trabalho. Dentre estes destacam-se a seguir alguns itens como perspectivas de trabalhos futuros.

- Análise e desenvolvimento de métodos de filtragem adaptativa por blocos de dados no domínio da frequência [118, 119];
- Análise e desenvolvimento de métodos de separação cega de sinais acústicos [120];
- Aplicação dos métodos de filtragem adaptativa cega na supressão de interferências em redes de telefonia celular 4G desenvolvidos a partir da integração de redes 3G com redes WLAN com padrão IEEE 802.11 [121, 122];
- Análise e desenvolvimento de métodos de treinamento de redes neurais recorrentes [123];

- Prova da positividade do vetor de margem fornecido pelo método CGP-CLF para o treinamento de perceptrons;
- Análise e desenvolvimento de métodos de treinamento de perceptrons em tempo real;

Apêndice A

Artigos Publicados

Adaptive Filtering Algorithms Designed Using Control Liapunov Functions

Oumar Diene and Amit Bhaya

Abstract—The standard conjugate gradient (CG) method uses orthogonality of the residues to simplify the formulas for the parameters necessary for convergence. In adaptive filtering, the sample-by-sample update of the correlation matrix and the cross-correlation vector causes a loss of the residue orthogonality in a modified online algorithm, which, in turn, results in loss of convergence and an increase of the filter quadratic mean error. This letter extends a recently proposed control Liapunov function analysis of the CG method viewed as a dynamic system in the standard feedback configuration to the case of adaptive filtering.

Index Terms—Adaptive equalizer, adaptive filtering algorithms, control Liapunov function (CLF), iterative methods, linear prediction, system identification.

I. INTRODUCTION

THE conjugate gradient (CG) is considered to be one of the best iterative methods for linear systems of equations with symmetric positive definite (spd) coefficient matrices. The standard CG method uses orthogonality of the residues to simplify the formulas for the parameters necessary for convergence [1]. In applications such as adaptive filtering, the sample-by-sample update of the correlation matrix and the cross-correlation vector causes a loss of the residue orthogonality in a modified online algorithm. This loss of orthogonality results in loss of convergence and an increase of the filter quadratic mean error. This letter extends a recently proposed control Liapunov function (CLF) analysis of the CG method viewed as a dynamic system in the standard feedback configuration [8] (plant + controller) to the case of adaptive filtering. The main technique consists of an appropriate choice of a CLF and has been used in [3], [8] and to analyze and design Krylov methods.

II. PROBLEM FORMULATION

Consider the linear system

$$\mathbf{R}\mathbf{w} = \mathbf{b}. \quad (1)$$

Iterative methods for solving the system (1) can be described by the following equation:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \quad (2)$$

Manuscript received October 21, 2005; revised November 17, 2005. This work was supported in part by Projects 141566/2004-3 and 551863/2002-1 of CNPq and in part by the agencies CAPES and FAPERJ. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Richard J. Kozick.

The authors are with the Department of Electrical Engineering, NACAD-COPPE/Federal University of Rio de Janeiro, 21945-970 Rio de Janeiro, Brazil (e-mail: oumar@ufrj.br; amit@nacad.ufrj.br).

Digital Object Identifier 10.1109/LSP.2005.863659

TABLE I
HESTENES-STIEFEL CG ALGORITHM FOR CONSTANT LINEAR SYSTEMS

Calculate $\mathbf{g}_0 = \mathbf{b} - \mathbf{R}\mathbf{w}_0, \mathbf{p}_0 = \mathbf{g}_0$
For $k = 0, 1, \dots$, until convergence
$\alpha_k = \frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}\mathbf{p}_k \rangle}$
$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k$
$\mathbf{g}_{k+1} = \mathbf{g}_k - \alpha_k \mathbf{R}\mathbf{p}_k$
$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}\mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}\mathbf{p}_k \rangle}$
$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k$
End

where \mathbf{w}_k is an approximation of the solution, and \mathbf{u}_k is a correction calculated in order to increase the accuracy of the approximated solution \mathbf{w}_{k+1} at each iteration [1] and then drive the residue, defined as $\mathbf{g}_k = \mathbf{b} - \mathbf{R}\mathbf{w}_k$, to zero in a finite number of iterations. The problem of solving (1) by iterative methods can be represented by the signal regulation problem

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \\ \mathbf{y}_k = \mathbf{R}\mathbf{w}_k \\ \mathbf{g}_k = \mathbf{b} - \mathbf{y}_k \\ \mathbf{u}_k = f(\mathbf{g}_k) \end{cases} \quad (3)$$

i.e., designing control signal \mathbf{u}_k in order to zero the steady-state error by choice of the control law $f(\cdot)$. In mathematical terms, $\lim_{k \rightarrow \infty} \mathbf{g}_k = \mathbf{0} \Leftrightarrow \lim_{k \rightarrow \infty} \mathbf{R}\mathbf{w}_k = \mathbf{b}$.

System (3) models an iterative method as a feedback control system (plant $\{\mathbf{I}, \mathbf{I}, \mathbf{R}, \mathbf{0}\}$ with controller in unitary feedback configuration). In control theory, both classical and modern, there exist various techniques of solving the signal regulation problem (3) representing the solution of (1), which correspond to different choices of the controller $f(\cdot)$ in (3). However, as shown in [4] and [9], many of these techniques (pole allocation, state feedback) result in controllers that explicitly use the inverse matrix \mathbf{R}^{-1} . In [3], the authors show that, by using appropriate quadratic CLFs, choosing a proportional derivative controller for $f(\cdot)$ in (3) leads to the CG method, shown in Table I. The CLF method proposed in [3], [8] gives an idea of the robustness of the CG method, since it does not use the assumption $\mathbf{p}_0 = \mathbf{g}_0$ (initialization of conjugate directions), used to obtain residue orthogonality (see, e.g., [2]). This indicates that, in situations where the assumption does not hold, the determination of CG parameters by CLF still ensures the decrease of some norm of the residual vector. When the assumption $\mathbf{p}_0 = \mathbf{g}_0$ holds, it can be shown that the parameters determined by CLFs are the same as those commonly found in the literature [2, p. 411, Eq. 3:1b and 3:1e].

III. DESIGNING NEW ADAPTIVE FILTERING ALGORITHMS

As shown in [5] and [6], iterative methods such as the CG method can be used to solve the adaptive filter problem. In [5], matrices \mathbf{R} and \mathbf{b} are estimated using a sliding data window and are kept fixed in this window during CG iteration. In this case, the gradient of the objective function is estimated by a mean of n_w past values such as

$$\mathbf{g}_w(k) = \left(\frac{2}{n_w}\right) \sum_{j=k-n_w+1}^k \{\mathbf{w}^T(k)\mathbf{x}(j) - d(j)\} \mathbf{x}(j). \quad (4)$$

Notice that for each sample k , $\min(M, n_w)$ iterations would be necessary for the convergence of the CG method. Thus, the choice of n_w , the window in which the mean of the gradient of the objective function is evaluated, has a direct impact on convergence of the filter weights. An increase in the dimension of the *tap* and/or the input data directly causes an increase in the computational cost of the method, which can exceed the cost of inverting the matrix \mathbf{R} . A modification using an exponentially decaying window was made in [6]. This modification permits only one iteration of the CG method at each sample and leads to an algorithm with performance comparable to the RLS algorithm; however, in order to ensure the convergence, it is necessary to introduce a factor η , determined experimentally, in the calculation of α_k , and the Polak–Ribière formula for the calculation of β_k . In this letter, it is shown that the CLF method [3], [8], can be extended to the design of adaptive filters.

Using the exponentially decaying data window proposed in [6], the correlation and the cross-correlation functions are given by

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T \quad (5)$$

$$\mathbf{b}_{k+1} = \lambda_f \mathbf{b}_k + d_{k+1} \mathbf{x}_{k+1} \quad (6)$$

where λ_f is the forgetting factor. For sample-by-sample processing, a recursive formulation for the residual vector can be found by using (2), (5), and (6), resulting in

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} (d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k). \quad (7)$$

Then, for the time-varying case, the system (3) can be rewritten as

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} (d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k) \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k. \end{cases} \quad (8)$$

For the system (8), the main result of this letter is stated as follows.

Theorem 1: The following assumptions are made:

- 1) $\inf \lambda_{\min}(\mathbf{R}_k^{-1}) = \zeta_1 > 0$, $\inf \lambda_{\min}(\mathbf{R}_k) = \zeta_2 > 0$;
- 2) $\mathbf{x}_{k+1}^T \mathbf{w}_k \approx \mathbf{x}_{k+1}^T \mathbf{w}_{k+1} = y_{k+1}$.

With these assumptions, the choices

$$\alpha_k = \frac{\langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} \quad (9)$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} \quad (10)$$

are optimal and ensure that the equilibrium of the system (8) is asymptotically stable (i.e., $\mathbf{g}_k \rightarrow 0$ and $\mathbf{p}_k \rightarrow 0$).

Proof: It is desired that $e_{k+1} = d_{k+1} - y_{k+1} \rightarrow 0$, i.e., $y_{k+1} \rightarrow d_{k+1}$. By hypothesis (2), $\mathbf{x}_{k+1}^T \mathbf{w}_{k+1} \approx \mathbf{x}_{k+1}^T \mathbf{w}_k$, thus $d_{k+1} \approx \mathbf{x}_{k+1}^T \mathbf{w}_k$, and substituting in (7) gives

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k. \quad (11)$$

Choose the CLF $V_{\mathbf{g}_k}(k) = \langle \mathbf{g}_k, \mathbf{R}_k^{-1} \mathbf{g}_k \rangle$; then

$$\Delta V_{\mathbf{g}} = \langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1}^{-1} \mathbf{g}_{k+1} \rangle - \langle \mathbf{g}_k, \mathbf{R}_k^{-1} \mathbf{g}_k \rangle. \quad (12)$$

Substituting (11) into (12) results in

$$\Delta V_{\mathbf{g}} = \lambda_f^2 \langle \mathbf{g}_k, \mathbf{R}_{k+1}^{-1} \mathbf{g}_k \rangle - 2\alpha_k \langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle + \alpha^2 \langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle - \langle \mathbf{g}_k, \mathbf{R}_k^{-1} \mathbf{g}_k \rangle. \quad (13)$$

The optimal value of α_k (which makes $\Delta V_{\mathbf{g}}$ as negative as possible), given by (9), is determined calculating $(\partial \Delta V_{\mathbf{g}} / \partial \alpha_k)$ and setting it to zero. Substituting (9) into (13) yields

$$\Delta V_{\mathbf{g}} = -\frac{\langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} + \lambda_f^2 \langle \mathbf{g}_k, \mathbf{R}_{k+1}^{-1} \mathbf{g}_k \rangle - \langle \mathbf{g}_k, \mathbf{R}_k^{-1} \mathbf{g}_k \rangle. \quad (14)$$

Using the matrix inversion lemma [7], one can write

$$\mathbf{R}_{k+1}^{-1} = \lambda_f^{-1} \mathbf{R}_k^{-1} - c \lambda_f^{-2} \mathbf{R}_k^{-1} \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T \mathbf{R}_k^{-1} \quad (15)$$

where $c = (1 / (1 + \lambda_f^{-1} \mathbf{x}_{k+1}^T \mathbf{R}_k^{-1} \mathbf{x}_{k+1})) > 0$. Inserting this into (14) results in

$$\Delta V_{\mathbf{g}} = -\frac{\langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} - (1 - \lambda_f) \langle \mathbf{g}_k, \mathbf{R}_k^{-1} \mathbf{g}_k \rangle - c \langle \mathbf{g}_k, \mathbf{R}_k^{-1} \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T \mathbf{R}_k^{-1} \mathbf{g}_k \rangle. \quad (16)$$

Then, from hypothesis (1), it follows that $\lambda_f < 1$ implies that

$$\Delta V_{\mathbf{g}} < -\zeta_1 \|\mathbf{g}_k\|_2^2 < 0 \quad (17)$$

which implies that $\|\mathbf{g}_k\|_{\mathbf{R}_k^{-1}}$ is a decreasing sequence.

For the choice of β_k , choose the CLF $V_{\mathbf{p}_k}(k) = \langle \mathbf{p}_k, \mathbf{R}_k \mathbf{p}_k \rangle = \|\mathbf{p}_k\|_{\mathbf{R}_k}^2$. Then

$$\Delta V_{\mathbf{p}} = \langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{g}_{k+1} \rangle + 2\beta_k \langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle + \beta_k^2 \langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle - \langle \mathbf{p}_k, \mathbf{R}_k \mathbf{p}_k \rangle.$$

In this case, the optimal value of β_k , given by (10), is obtained by calculating $(\partial \Delta V_{\mathbf{p}} / \partial \beta_k)$ and setting it to zero, yielding

$$\begin{aligned} V_{\mathbf{p}_{k+1}}(k+1) &= \|\mathbf{g}_{k+1}\|_{\mathbf{R}_{k+1}}^2 - \frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle} \\ &< \|\mathbf{g}_{k+1}\|_{\mathbf{R}_{k+1}}^2. \end{aligned} \quad (18)$$

From (17), the equivalence of norms, and hypothesis (1), it can be concluded that $\|\mathbf{g}_{k+1}\|_{\mathbf{R}_{k+1}}$ is a decreasing sequence. Thus, (18) implies that \mathbf{p}_{k+1} decreases in \mathbf{R}_{k+1} -norm, not necessarily monotonically, concluding the proof. ■

Observation 1: Notice that (16) can be used to determine the value of λ_f , which makes $\Delta V_{\mathbf{g}}$ as negative as possible. Using the expression (16) and solving the unidimensional minimization problem $\min_{\lambda_f} \Delta V_{\mathbf{g}}$ allows, in theory, the determination of an optimal value of λ_f .

The pseudocode of the extension of the Hestenes–Stiefel CG method for adaptive filtering algorithms, based on theorem 1 and referenced in this letter as CG-CLF, is presented in Table II.

Note that the algorithm in Table II does not introduce any additional computation and requires fewer floating point operations with respect to the algorithm in [6]. As pointed out by

TABLE II
CG-CLF METHOD FOR ADAPTIVE FILTERS

Calculate $\mathbf{R}_0 = \mathbf{x}_0 \mathbf{x}_0^T$, $\mathbf{b}_0 = d_0 \mathbf{x}_0$, $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$, $\mathbf{p}_0 = \mathbf{g}_0$
 For $k = 0, 1, \dots, N-1$

$$\alpha_k = \frac{\langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T$$

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} (d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k)$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k$$

End

a reviewer, choosing the *ad hoc* parameter η in [6] equal to λ_f gives our formula for α_k , which is shown above to be an optimal choice in the sense of making $\Delta \mathbf{V}_g$ as negative as possible, although, in nonstationary environments, a smaller value ($\eta < \lambda_f$, which keeps $\Delta \mathbf{V}_g < 0$) may be advisable.

IV. SIGNAL PROCESSING APPLICATIONS

Several simulations are made using the three basic configurations: equalization, system identification, and linear prediction [7] and [10]. The standard CG method, used in [5] for the stationary case, is implemented below in a time-varying environment and its performance compared to the adaptive filters designed by the algorithms II and Chang–Willson [6], in order to emphasize the problem caused by loss of orthogonality.

A. Adaptive Equalizer

This example is taken from [7]. The input to the system is a random sequence $\{a(k)\}$ with values ± 1 , so that $\{a(k)\}$ is zero mean. The sequence is distorted by a channel with impulse response

$$h(k) = \frac{1}{2} \left[1 + \frac{\cos(2\pi(k-2))}{W} \right], \quad k = 1, 2, 3$$

$$= 0, \quad k > 3$$

where W controls the amplitude distortion in the channel. An increase in W increases the channel distortion and implies a higher value for the eigenvalue spread ρ of the correlation matrix \mathbf{R} of the inputs to the equalizer. The channel is corrupted by an additive white noise sequence that is zero mean with a variance $\sigma^2 = 0.001$. The equalizer has 11 taps. Since the channel impulse response is symmetric about $k = 2$, it follows that the optimum tap weights of the equalizer are symmetric about the sixth tap. Accordingly, the channel input $\{a(k)\}$ is delayed by seven samples to provide the desired response for the equalizer. The experiment was performed with two values of W equaling 2.9 and 3.5, which implied ρ equal to 6.07 and 46.821, respectively. In this case, the parameters were chosen such that $\lambda_f = \eta = 0.99$, and the ensemble averaging was carried out over 200 independent trials of the experiment. Fig. 1 shows the learning curves of the adaptive filters designed by the algorithms CG-CLF, Chang–Willson [6], and standard CG in the equalizer configuration, and Table III compares the mean-squared errors. It can be observed that the CG-CLF method proposed in this letter has convergence properties comparable with the

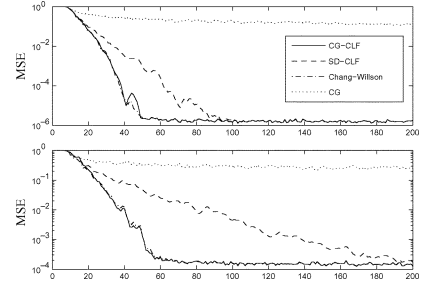


Fig. 1. Learning curves of the adaptive equalizer. The steady-state values are shown in Table III. (Top) $W = 2.9$. (Bottom) $W = 3.5$.

TABLE III
STEADY-STATE MEAN-SQUARED ERROR

Method	Adaptive Equalizer		S.I.
	$W = 2.9$	$W = 3.5$	
CG-CLF	1.74×10^{-6}	1.68×10^{-4}	1.35×10^{-2}
SD-CLF	1.82×10^{-6}	3.70×10^{-3}	1.35×10^{-2}
Chang-Willson	1.69×10^{-6}	1.63×10^{-4}	1.36×10^{-2}
CG	1.36×10^{-1}	2.62×10^{-1}	5.05×10^{-1}

Chang–Willson method [6], as well as with the RLS method [6]. For the standard CG method, the condition $\mathbf{p}_0 = \mathbf{g}_0$ is necessary to ensure the orthogonality of the residual vectors that guarantees the convergence [2]. In the time-varying case, this is equivalent to setting $\mathbf{p}_k = \mathbf{g}_k$ at each iteration. Thus, observing that the standard CG method actually uses $\mathbf{p}_k = \mathbf{g}_k + \beta_{k-1} \mathbf{p}_{k-1}$, this means that residual vectors will lose orthogonality, leading to bad performance. Given this intuition, another modification of the CG method is proposed in algorithm 1. This modification consists in substituting the last step of the standard CG method with $\mathbf{p}_{k+1} = \mathbf{g}_{k+1}$. This new modification leads to the well-known *steepest descent* (SD) method, which can also be derived by an appropriate choice of CLF [3], [8]. It can be noticed that the SD-CLF and the other CLF's method converge to the same steady-state mean-squared error; however, the convergence rate of the SD-CLF is lower and is inversely proportional to the eigenvalue spread ρ (the higher ρ is, the lower the rate of convergence). It is important to notice that, although it has a rate of convergence lower than that of the RLS algorithm in a noisy environment, the SD-CLF method can be used in cases where the LMS algorithm (which has a comparable rate of convergence but higher steady-state mean-squared error and higher computational cost) is applicable.

Also note that algorithm SD-CLF has fewer steps and a lower computational cost than the algorithms in [5] and [6]. From theorem 1, it is straightforward to prove the asymptotic stability of algorithm SD-CLF.

B. System Identification (SI)

The unknown plant is a finite time impulse response (FIR) filter of order 20, and the input is a white Gaussian signal of variance $\sigma^2 = 1$. The ensemble average is composed of 100 independent samples. Fig. 2 shows the learning curves of the adaptive filters designed by the algorithms CG-CLF, SD-CLF, Chang–Willson [6], and standard CG in the SI configuration. Table III shows that the methods proposed have convergence properties comparable with the Chang–Willson and the RLS

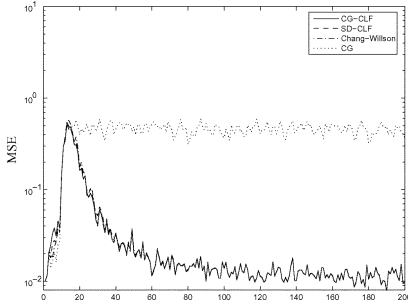


Fig. 2. Learning curves of the adaptive filters in system identification configuration. The steady-state values are shown in Table III.

TABLE IV
SD-CLF METHOD FOR ADAPTIVE FILTERS

Calculate $\mathbf{R}_0 = \mathbf{x}_0 \mathbf{x}_0^T$, $\mathbf{b}_0 = d_0 \mathbf{x}_0$, $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$
For $k = 0, 1, \dots, N-1$
$\alpha_k = \frac{\langle \lambda_f \mathbf{g}_k, \mathbf{g}_k \rangle}{\langle \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}$
$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{g}_k$
$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T$
$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{x}_{k+1} (d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k)$
End

method [6]. As in the case of the adaptive equalizer, the standard CG loses orthogonality of the residual vectors and, therefore, presents worse performance than the other methods.

C. Linear Prediction

The third application, from [5], is a two-tap adaptive filter in the linear prediction scenario. The tap inputs $u(k-1)$ and $u(k-2)$ were drawn from a real-valued autoregressive process described by

$$u(k) + a_1 u(k-1) + a_2 u(k-2) = v(k) \quad (19)$$

where $v(k)$ is a white noise process with zero mean and variance σ_v^2 . The values of a_1 and a_2 are determined by the desired eigenvalue spread ρ . For one trial of the experiment, 500 samples of white noise process $\{v(k)\}$ were obtained by a random-number generator of zero mean and adjustable variance. The ensemble average is composed of 200 independent trials, and the experiment was conducted for eigenspreads $\rho = 3$ and 10. The learning curves of the linear predictors designed by the algorithms CG-CLF, SD-CLF, Chang-Willson [6], and standard CG are not shown here for lack of space, but Table V compares the mean-squared errors and the misadjustment $M(\%)$ calculated by estimating the average mean-squared error $J(\infty)$ and using the theoretical value for J_{\min} [5]. Once again, the CLF methods have convergence properties comparable to the Chang-Willson method [6] and the RLS method [6]; however, the slow convergence of the SD-CLF method, which depends on the eigenvalue spread, observed in the case of the adaptive equalizer does not occur. For the standard CG method, the loss of orthogonality leads again to loss of convergence for small eigenspreads.

TABLE V
STEADY-STATE MEAN-SQUARED ERROR AND MISADJUSTMENT FOR THE LINEAR PREDICTORS $J_{\min} = 0.0322$ FOR $\rho = 10$ AND $J_{\min} = 0.0731$ FOR $\rho = 3$

Method	Linear Predictor		Misadjustment	
	$\rho = 10$	$\rho = 3$	$\rho = 10$	$\rho = 3$
CG-CLF	4.27×10^{-2}	1.02	0.33	12.95
SD-CLF	4.27×10^{-2}	1.02	0.33	12.95
Chang-Willson	4.28×10^{-2}	1.02	0.33	12.95
CG	5.55×10^{-2}	80.38	0.72	1098.59

V. CONCLUSION

The control viewpoint [3], [8] reveals that the formulas for the CG parameters α_k and β_k in [2, p. 411, Eq. 3:2a and 3:2b] actually enhance the robustness of the CG algorithm with respect to the standard equivalent choices ($\alpha_k = \langle \mathbf{g}_k, \mathbf{g}_k \rangle / \langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle$) and ($\beta_k = \langle \mathbf{g}_{k+1}, \mathbf{g}_{k+1} \rangle / \langle \mathbf{g}_k, \mathbf{g}_k \rangle$), since the latter makes use of the assumption $\mathbf{p}_0 = \mathbf{g}_0$, to obtain orthogonality of residues. Hestenes and Stiefel [2, p. 411] recognize that “*although these formulas are slightly more complicated, they have advantage that scale factors (introduced to increase accuracy) are more easily changed during the course of the computation.*” Our results, both theoretical and simulated, show that since the “more complicated formulas” correspond to optimal choices in the CLF method, they endow the system with robustness, since there is some leeway corresponding to neighboring nonoptimal choices of the parameters α_k and β_k . The authors in [6] treated a linear time-varying system; however, their convergence analysis was carried out for the “frozen” (i.e., time-invariant) case, whereas this letter proves asymptotic stability using the linear structure of the time-varying system (3) and the bilinear structure of the dynamical system solver (8).

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for helpful comments.

REFERENCES

- [1] Y. Saad, *Iterative Methods for Sparse Linear Systems*. Boston, MA: PWS-Kent, 1996.
- [2] M. R. Hestenes and E. Stiefel, “Methods of conjugate gradients for solving linear systems,” *J. Res. Natl. Bur. Stand.*, vol. 49, pp. 409–436, 1952.
- [3] A. Bhaya and E. Kaszkurewicz, “Iterative methods as dynamical systems with feedback control,” in *Proc. 42nd IEEE Conf. Decision Control*, Maui, HI, Dec. 2003, pp. 2374–2380.
- [4] O. Diene, “Linear iterative methods: Design and analysis using control theory tools,” Master’s thesis, COPPE/UFRJ, Apr. 2004. In Portuguese.
- [5] G. K. Boray and M. D. Srinath, “Conjugate gradient techniques for adaptive filtering,” *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 39, no. 1, pp. 1–10, Jan. 1992.
- [6] P. S. Chang and A. N. Willson, “Analysis of conjugate gradient algorithms for adaptive filtering,” *IEEE Trans. Signal Process.*, vol. 48, no. 2, pp. 409–418, Feb. 2000.
- [7] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1991.
- [8] A. Bhaya and E. Kaszkurewicz, *Control Perspectives on Numerical Algorithms and Matrix Problems*. Philadelphia, PA: SIAM, 2006.
- [9] O. Diene and A. Bhaya, “Linear iterative methods: Design and analysis using control theory tools,” in *Proc. 15th Brazilian Conf. Automatica*, Gramado, Brazil, Sep. 2004. In Portuguese.
- [10] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New York: Wiley, 2003.

Erratum

Correction to “Adaptive Filtering Algorithms Designed Using Control Liapunov Functions”

Oumar Diene and Amit Bhaya

After the publication of [1], the authors noticed an error in the application of Assumption 2 in Theorem 1 (the main result of the note) and would like to correct the statement of Theorem 1 as well as indicate the resulting changes in the proof and in Tables II and IV. The overall conclusions as well as the numerical results were not affected by this error and remain unchanged, as stated in the original letter.

Changes to Theorem 1: In formula (9) \mathbf{g}_k should be replaced by $\tilde{\mathbf{g}}_k$, and the text: “where $\tilde{\mathbf{g}}_k := \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1}$,” should be introduced between formula (10) and the text: “are optimal and ensure that . . .”

Changes to the proof (of Theorem 1): The first sentence: “It is desired that $e_{k+1} = d_{k+1} - y_{k+1} \rightarrow 0$, i.e., $y_{k+1} \rightarrow d_{k+1}$ ” should be removed, and the following sentences and equations:

“Define

$$\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} e_{k+1} \quad (11a)$$

then from (11)

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k. \quad (11b)$$

From (a), (b), and the fact that, for online processing, entry \mathbf{x}_{k+2} is not available at instant $k + 1$, it can be written that

$$\begin{aligned} \tilde{\mathbf{g}}_{k+1} &= \lambda_f \mathbf{g}_{k+1} + \mathbf{x}_{k+2} e_{k+2} \\ &\approx \lambda_f \mathbf{g}_{k+1} = \lambda_f \tilde{\mathbf{g}}_k - \alpha_k \lambda_f \mathbf{R}_{k+1} \mathbf{p}_k. \end{aligned} \quad (11c)$$

Manuscript received October 21, 2005; revised November 17, 2005. This work was supported in part by Projects 141566/2004-3 and 551863/2002-1 of CNPq and in part by the agencies CAPES and FAPERJ. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Richard J. Kozick.

The authors are with the Department of Electrical Engineering, NACAD-COPPE/Federal University of Rio de Janeiro, 21945-970 Rio de Janeiro, Brazil (e-mail: oumar@ufrj.br; amit@nacad.ufrj.br).

Digital Object Identifier 10.1109/LSP.2007.911603

Thus, system (8) can be rewritten as

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \tilde{\mathbf{g}}_{k+1} = \lambda_f \tilde{\mathbf{g}}_k - \alpha_k \lambda_f \mathbf{R}_{k+1} \mathbf{p}_k \\ \mathbf{p}_{k+1} = \frac{1}{\lambda_f} \tilde{\mathbf{g}}_{k+1} + \beta_k \mathbf{p}_k. \end{cases} \quad (11d)$$

” should be introduced between (11) and the text: “Choose the control Liapunov function. . .” In addition, after (11), \mathbf{g}_k and \mathbf{g}_{k+1} should be replaced, wherever they occur, by $\tilde{\mathbf{g}}_k$ and $(1/\lambda_f)\tilde{\mathbf{g}}_{k+1}$, respectively. With these changes, the asymptotic stability of system (8) goes through as before.

In Tables II and IV, inside the “for” loop, the new first line should read

$$\tilde{\mathbf{g}}_k = \lambda_f \mathbf{g}_k + \mathbf{x}_{k+1} \left(d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k \right)$$

and in both tables, $\lambda_f \mathbf{g}_k$ in the formula for α_k should be replaced by $\tilde{\mathbf{g}}_k$. Finally, in both Tables II and IV, after calculating the correlation matrix \mathbf{R}_{k+1} , the residue should be calculated using

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k,$$

and

$$\mathbf{g}_{k+1} = \tilde{\mathbf{g}}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k$$

respectively.

REFERENCES

- [1] O. Diene and A. Bhaya, “Adaptive filtering algorithms designed using control Liapunov functions,” *IEEE Signal Process. Lett.*, vol. 13, no. 4, pp. 225–228, Apr. 2006.

MÉTODOS ITERATIVOS LINEARES PROJETADOS ATRAVÉS DA TEORIA DE CONTROLE E SUAS APLICAÇÕES

Oumar Diene*
oumar@ufrj.br

Amit Bhaya*
amit@nacad.ufrj.br

*Programa de Engenharia Elétrica, COPPE,
Núcleo de Atendimento em Computação de Alto Desempenho, NACAD,
Universidade Federal do Rio de Janeiro
C.P. 68504, RJ 21945-970, Brasil

RESUMO

É mostrado como projetar uma família de novos métodos iterativos lineares através de uma abordagem pela teoria de controle. Escolhas apropriadas de funções de Liapunov resultam em controladores da família PI/PD com ganhos variantes no tempo. Os novos métodos iterativos são aplicados à determinação dos pesos ótimos de filtros adaptativos. Simulações são feitas para mostrar que o desempenho dos novos algoritmos para filtros adaptativos é comparável ao método RLS.

PALAVRAS-CHAVE: Função de Liapunov, método iterativo, estabilidade, controlador PI/PD, gradiente conjugado, convergência, filtros adaptativos.

ABSTRACT

It is shown how to built a family of new iterative methods using a feedback control viewpoint together with appropriate choices of control Liapunov function, leading to a family of time varying PI/PD controllers. The new iterative methods are used to determine the optimal coefficients of adaptive filters. Simulations are performed in order to show that the new methods have the same performance as the RLS algorithm.

KEYWORDS: Control Liapunov function, iterative meth-

ods, stability, PI/PD controller, conjugate gradient algorithm, convergence, adaptive filters.

1 INTRODUÇÃO

O método do Gradiente Conjugado (CG) é considerado um dos melhores métodos iterativos para a classe de matrizes simétricas e definidas positivas (spd) (Saad e VanDerVorst (2000), VanDerVorst (2000), Saad (1996)). O método CG clássico é um método de direções conjugadas (Hestenes, 1980) que utiliza a ortogonalidade dos resíduos gerados pelo algoritmo para determinar os parâmetros necessários para a convergência. Em aplicações tais como filtragem adaptativa, a atualização das matrizes de correlação e de correlação cruzada a cada amostra acarreta a perda da ortogonalidade dos resíduos do algoritmo de atualização dos pesos do filtro. Esta perda de ortogonalidade resulta em uma perda de convergência do algoritmo e em um aumento do erro médio quadrático da saída do filtro. Neste trabalho o método CG é apresentado e analisado como um sistema dinâmico do tipo planta + controlador em realimentação unitária. Esta abordagem permite determinar os parâmetros do método, independentemente da ortogonalidade do resíduo, através de uma *control Liapunov function* (CLF), bem como projetar novos algoritmos. Esta técnica já foi utilizada em Bhaya e Kaszkurewicz (2004) para a análise/projeto dos métodos tipo Krylov, e consiste em uma escolha adequada de uma *Control Liapunov Function* (CLF). Estes autores mostraram, através de uma CLF, que o algoritmo do gradiente conjugado (CG)

Artigo submetido em 03/10/2005
1a. Revisão em 21/09/2006
Aceito sob recomendação do Editor Associado
Prof. Liu Hsu

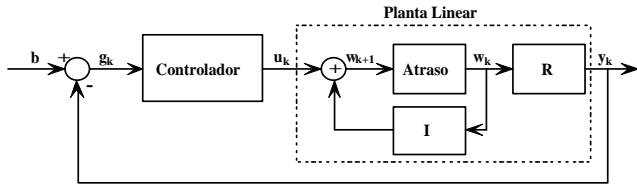


Figura 1: Diagrama de blocos ilustrativo do problema de regulação equivalente à resolução de (1) por métodos iterativos. O problema é: projetar um controlador tal que $\lim_{k \rightarrow \infty} \mathbf{g}_k = \mathbf{0} \Leftrightarrow \lim_{k \rightarrow \infty} \mathbf{R}\mathbf{w}_k = \mathbf{b}$.

pode ser interpretado como um sistema linear dinâmico sujeito a um tipo de controle proporcional e derivativo. Esta função permite determinar os valores ótimos dos parâmetros que tornam o processo assintoticamente estável. Mostra-se através de várias simulações que os métodos determinados desta forma apresentam um desempenho comparável ao do método RLS com um custo e complexidade computacional menor.

2 FORMULAÇÃO DO PROBLEMA

Considere o sistema linear

$$\mathbf{R}\mathbf{w} = \mathbf{b}. \quad (1)$$

Os métodos iterativos de resolução do sistema (1) podem ser descritos pela seguinte equação

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \quad (2)$$

onde \mathbf{w}_k é uma aproximação da solução de (1) e \mathbf{u}_k é uma correção calculado conforme o algoritmo utilizado, para gerar a cada instante uma aproximação mais precisa \mathbf{w}_{k+1} da solução de (1) (Saad, 1996), e levar o resíduo, definido como $\mathbf{g}_k = \mathbf{b} - \mathbf{R}\mathbf{w}_k$, a zero em um número finito de passos. O problema da resolução do sistema (1) por métodos iterativos pode ser representado por um problema de regulação de sinal, conforme mostra a figura 1. O problema da resolução de (1) por métodos iterativos consiste em projetar um controle \mathbf{u}_k para que o erro de regime seja nulo. Isto é $\lim_{k \rightarrow \infty} \mathbf{g}_k = \mathbf{0} \Leftrightarrow \lim_{k \rightarrow \infty} \mathbf{R}\mathbf{w}_k = \mathbf{b}$. O sistema discreto (3) representa a dinâmica deste problema de regulação:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \\ \mathbf{y}_k = \mathbf{R}\mathbf{w}_k \\ \mathbf{g}_k = \mathbf{b} - \mathbf{y}_k \\ \mathbf{u}_k = f(\mathbf{g}_k) \end{cases} \quad (3)$$

onde $f(\cdot)$ representa o controle adotado.

3 PROJETO DE CONTROLADORES

Na teoria de controle, tanto clássica como moderna, existem várias técnicas de resolução do problema de regulação de sinal (3) equivalente ao problema de resolução do sistema (1) que, na realidade, correspondem a maneiras diferentes de escolher o controlador $f(\cdot)$ em (3). Porém, conforme mostrado em Diene (2004), a maioria das técnicas consagradas na literatura (alocação de pólos, realimentação de estados) resulta em um controlador que usa explicitamente a inversa \mathbf{R}^{-1} da matriz do sistema. Neste trabalho o problema de regulação de sinal é resolvido por controladores do tipo proporcional derivativo e/ou proporcional integral que não utilizam a inversa de \mathbf{R} explicitamente. Para determinar os parâmetros necessários para estabilidade do sistema (equivalente a convergência do algoritmo de resolução de (1)), é usada a mesma técnica aplicada em Bhaya e Kaszkurewicz (2004) para analisar os métodos iterativos do tipo Krylov.

3.1 Controlador proporcional e derivativo (PD)

O sistema (3), representado na forma de diagrama de blocos na figura 1, leva a interpretação de um método iterativo como um sistema de controle realimentado (planta $\{\mathbf{I}, \mathbf{I}, \mathbf{R}, \mathbf{0}\}$ com controlador em configuração de realimentação unitária). O controlador proporcional e derivativo (derivada do estado) é tal que $\mathbf{u}_k = \alpha_k \mathbf{g}_k + \beta_k \Delta \mathbf{w}_k$, sendo que $\Delta \mathbf{w}_k = \mathbf{w}_k - \mathbf{w}_{k-1}$. Portanto, definindo

$$\mathbf{p}_k := \mathbf{g}_k + \gamma_k (\mathbf{w}_k - \mathbf{w}_{k-1}) \quad (4)$$

o sistema (3) resulta em

$$\begin{cases} \mathbf{u}_k = \alpha_k \mathbf{p}_k \\ \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \mathbf{g}_k - \alpha_k \mathbf{R}\mathbf{p}_k \\ \beta_k = \alpha_k \gamma_k \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k, \end{cases} \quad (5)$$

onde α_k é o ganho proporcional e β_k o ganho derivativo. A figura 2 mostra o diagrama de blocos da planta linear com o controlador discreto proporcional e derivativo em malha fechada.

Note que, considerando os parâmetros α_k e β_k como variáveis de controle e tomando \mathbf{g}_k e \mathbf{p}_k como as variáveis de estado, o sistema a ser controlado é do tipo bilinear. Isto é, o sistema é linear no controle se o estado é mantido constante, e linear no estado se o controle é mantido constante.

$$\begin{cases} \mathbf{g}_{k+1} = \mathbf{g}_k - \alpha_k \mathbf{R}\mathbf{p}_k \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k \end{cases} \quad (6)$$

O objetivo de controle é escolher os escalares α_k e β_k para levar os vetores de estado \mathbf{g}_k e \mathbf{p}_k a zero. Uma vez que o parâmetro α_k não é identicamente nulo, pode-se ver facilmente

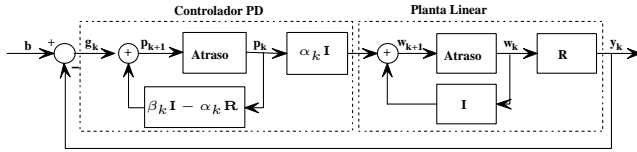


Figura 2: Diagrama de blocos da Planta Linear com um controlador PD. Os parâmetros α_k e β_k do controlador são calculados conforme as equações (11) e (16) para matrizes \mathbf{R} spd.

que o equilíbrio do sistema bilinear (6) se localiza na origem $\mathbf{g}_k = \mathbf{p}_k = \mathbf{0}$ para todo k . A escolha dos parâmetros α_k e β_k é feita em duas etapas. Na primeira, a partir de uma CLF, determina-se o valor de α_k que leva ao decrescimento de alguma norma de \mathbf{g}_k . Na segunda, uma outra função de Liapunov orienta a determinação de β_k que resulta no decrescimento de alguma norma de \mathbf{p}_k . A conclusão é que ambos os vetores \mathbf{g}_k e \mathbf{p}_k convergem para zero, como desejado.

Supondo que a matriz \mathbf{R} é simétrica positiva definida, a CLF escolhida para a determinação de α_k é dada pela equação (7). Note que a escolha da CLF é arbitrária e que cada escolha diferente de CLF resulta em um algoritmo diferente.

$$\mathbf{V}_g(k) = \langle \mathbf{g}_k, \mathbf{R}^{-1} \mathbf{g}_k \rangle. \quad (7)$$

Pela teoria de Liapunov (Slotine e Li (1991) e Khalil (2002)) o equilíbrio de (6) será assintoticamente estável se

$$\begin{aligned} \Delta \mathbf{V}_g &= \mathbf{V}_g(k+1) - \mathbf{V}_g(k) \\ &= \langle \mathbf{g}_{k+1}, \mathbf{R}^{-1} \mathbf{g}_{k+1} \rangle - \langle \mathbf{g}_k, \mathbf{R}^{-1} \mathbf{g}_k \rangle < 0. \end{aligned} \quad (8)$$

Substituindo o resíduo de (6) em (8) e simplificando resulta:

$$\Delta \mathbf{V}_g = -2\alpha_k \langle \mathbf{g}_k, \mathbf{p}_k \rangle + \alpha_k^2 \langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle. \quad (9)$$

O valor de α_k que minimiza $\Delta \mathbf{V}_g$ é determinado através de sua derivada fazendo:

$$\frac{\partial \Delta \mathbf{V}_g}{\partial \alpha_k} = -2 \langle \mathbf{g}_k, \mathbf{p}_k \rangle + 2\alpha_k \langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle = 0. \quad (10)$$

Portanto,

$$\alpha_k = \frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle}. \quad (11)$$

Substituindo (11) em (9) resulta:

$$\Delta \mathbf{V}_g = -\frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle} < 0. \quad (12)$$

Prosseguindo na determinação dos parâmetros estabilizantes do sistema (6), escolha-se a CLF

$$\mathbf{V}_p(k) = \langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle, \quad (13)$$

para orientar o cálculo de β_k . Portanto tem-se

$$\begin{aligned} \mathbf{V}_p(k+1) &= \langle \mathbf{p}_{k+1}, \mathbf{R} \mathbf{p}_{k+1} \rangle \\ &= \langle \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k, \mathbf{R} (\mathbf{g}_{k+1} + \beta_k \mathbf{p}_k) \rangle \\ &= \langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{g}_{k+1} \rangle + 2\beta_k \langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{p}_k \rangle + \\ &\quad + (\beta_k^2 - 1) \langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle + \langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle, \end{aligned}$$

o que implica que:

$$\Delta \mathbf{V}_p = \langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{g}_{k+1} \rangle + 2\beta_k \langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{p}_k \rangle + (\beta_k^2 - 1) \langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle. \quad (14)$$

Seguindo o mesmo raciocínio de função ótima de Liapunov, calcula-se

$$\frac{\partial \Delta \mathbf{V}_p}{\partial \beta_k} = 2 \langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{p}_k \rangle + 2\beta_k \langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle \quad (15)$$

assim,

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle} \quad (16)$$

é uma escolha ótima (torna $\partial \Delta \mathbf{V}_p / \partial \beta_k$ o mais negativo possível) que resulta em

$$\|\mathbf{p}_{k+1}\|_{\mathbf{R}}^2 = \|\mathbf{g}_{k+1}\|_{\mathbf{R}}^2 - \frac{\langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle} < \|\mathbf{g}_{k+1}\|_{\mathbf{R}}^2. \quad (17)$$

De (12) e pela equivalência de normas, pode-se concluir que \mathbf{g}_k decresce em qualquer norma induzida. Portanto, (17) implica que \mathbf{p}_{k+1} decresce na norma induzida pela matriz \mathbf{R} .

Note que o pseudocódigo do método iterativo decorrente do controlador PD, mostrado no algoritmo 1, corresponde ao método do gradiente conjugado (CG). Vale ressaltar que o método utilizado para a derivação dos parâmetros α_k e β_k do método CG, apresentada neste trabalho, permite ter uma noção da robustez do método CG. A análise por uma CLF, ao contrário da análise padrão (por exemplo Saad (1996)), não utiliza nenhuma informação sobre as propriedades do resíduo \mathbf{g}_k ou do vetor \mathbf{p}_k (como a ortogonalidade do resíduo \mathbf{g}_k , isto é $\mathbf{g}_{k+1}^T \mathbf{g}_k = 0$). Isto indica que, em uma implementação de precisão finita, mesmo que as propriedades como a ortogonalidade sejam perdidas, a escolha dos parâmetros α_k e β_k conforme as equações (11) e (16) garante o decrescimento das normas de \mathbf{g}_k e de \mathbf{p}_k ponderadas respectivamente pelas matrizes \mathbf{R}^{-1} e \mathbf{R} . Na presença de ortogonalidade, pode-se mostrar que os parâmetros α_k e β_k dadas pelas equações (11) e (16) são iguais aos parâmetros comumente encontrados na literatura (Hestenes e Stiefel, 1952).

Algoritmo 1

Calcular $\mathbf{g}_0 = \mathbf{b} - \mathbf{R} \mathbf{w}_0$, $\mathbf{p}_0 = \mathbf{g}_0$

Para $k = 0, 1, \dots$ até convergência

$$\alpha_k = \frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{g}_{k+1} = \mathbf{g}_k - \alpha_k \mathbf{R} \mathbf{p}_k$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k$$

Fim

Para garantir a ortogonalidade dos resíduos do método CG padrão, é necessário inicializar os vetores tais que $\mathbf{p}_0 = \mathbf{g}_0$ (Hestenes e Stiefel, 1952). Para ilustrar o problema gerado pela perda de ortogonalidade dos resíduos do método CG padrão, implementa-se o algoritmo 1 e o método CG padrão (veja, por exemplo, Saad (1996)) e compara-se o desempenho dos algoritmos quando $\mathbf{p}_0 = c\mathbf{g}_0$, c uma constante real. Neste exemplo, as matrizes são escolhidas tais que:

$$\mathbf{R} = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \mathbf{w}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Neste caso, note que $\mathbf{g}_0 = \mathbf{b}$, a tabela 1 apresenta o desempenho dos métodos. O critério de parada é dado pela condição $\|\mathbf{g}_k\| \leq 10^{-3}$, e o número máximo de iterações é igual a 50. Note que em todos os casos da tabela 1 ocorre uma perda de ortogonalidade dos resíduos dos algoritmos, exceto quando $\mathbf{p}_0 = \mathbf{g}_0$. Portanto, para a implementação padrão do CG, pode-se ver que, na ausência de ortogonalidade, quanto maior a razão $\|\mathbf{p}_0\|/\|\mathbf{g}_0\|$ menor será a taxa de convergência, e quando $\|\mathbf{p}_0\| \rightarrow 0$, o algoritmo perde a convergência. Em outras palavras, na ausência de ortogonalidade dos resíduos do método CG padrão, a escolha de um passo inicial α_0 resulta em uma convergência lenta do algoritmo enquanto a escolha de α_0 grande provoca a divergência do método. Por outro lado, o algoritmo com parâmetros determinados via CLF apresenta uma robustez no sentido da estabilidade de Liapunov.

3.2 Controlador proporcional e integral (PI)

A forma geral do controlador discreto proporcional e integral é $\mathbf{u}_k = \alpha_k \mathbf{g}_k + \beta_k \sum_{i=0}^k \mathbf{g}_i$, onde o primeiro termo é denominado *termo proporcional* e o segundo, *termo integral*. Isto pode ser reescrito na forma da seguinte equação de estados

$$\begin{cases} \mathbf{q}_{k+1} = \mathbf{q}_k + \mathbf{g}_k \\ \mathbf{u}_k = k_i \mathbf{q}_k \end{cases} \quad (18)$$

Introduzindo este novo estado ‘integrador’ em (3), o sistema resulta em

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} = \mathbf{g}_k - \alpha_k \mathbf{R} \mathbf{p}_k \\ \mathbf{q}_{k+1} = \mathbf{q}_k + \beta_k \mathbf{g}_k \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \gamma_k \mathbf{q}_{k+1} \end{cases} \quad (19)$$

onde α_k é o ganho proporcional, \mathbf{q}_k representa o estado do integrador, β_k é um ganho introduzido para que o estado do integrador seja nulo em regime permanente, \mathbf{p}_k é um estado auxiliar e γ_k multiplicado por α_k representa o ganho integral, conforme mostra a figura 3.

A análise do sistema com controlador PI é semelhante à do sistema com controlador PD. O sistema bilinear considerado

Cond. Ini.	Método	Iterações	$\ \mathbf{g}_k\ $
$\mathbf{p}_0 = \mathbf{g}_0$	CG CLF	2	0
	CG	2	0
$\mathbf{p}_0 = 2\mathbf{g}_0$	CG CLF	2	0
	CG	13	5.98×10^{-4}
$\mathbf{p}_0 = 5\mathbf{g}_0$	CG CLF	2	0
	CG	36	9.43×10^{-4}
$\mathbf{p}_0 = 20\mathbf{g}_0$	CG CLF	2	0
	CG	50	1.67×10^{-2}
$\mathbf{p}_0 = 0.5\mathbf{g}_0$	CG CLF	2	0
	CG	50	1.76
$\mathbf{p}_0 = 0.2\mathbf{g}_0$	CG CLF	2	1.57×10^{-16}
	CG	50	1.03×10^{20}

Tabela 1: Comparação do desempenho do método CG utilizando parâmetros determinados via CLF com o método CG assumindo ortogonalidade. A perturbação é gerada na inicialização do vetor \mathbf{p}_0 . Neste caso \mathbf{p}_0 é escolhido aleatoriamente (normalmente $\mathbf{p}_0 = \mathbf{g}_0$). O algoritmo com parâmetros determinados via CLF apresenta uma robustez no sentido da estabilidade de Liapunov enquanto o algoritmo tradicional perde a convergência. O número máximo de iterações é 50 e o critério de parada é $\|\mathbf{g}_k\| \leq 10^{-3}$.

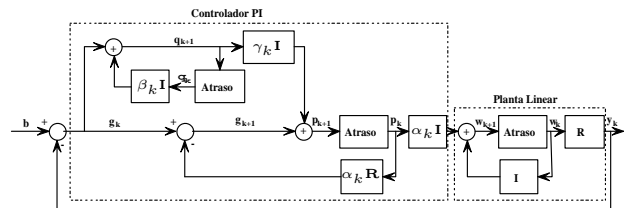


Figura 3: Diagrama de blocos da Planta Linear com um controlador PI. Os parâmetros α_k , β_k e γ_k do controlador são calculados conforme as equações (21), (22) e (23) para \mathbf{R} spd.

neste caso é dado por:

$$\begin{cases} \mathbf{g}_{k+1} = \mathbf{g}_k - \alpha_k \mathbf{R} \mathbf{p}_k \\ \mathbf{q}_{k+1} = \mathbf{q}_k + \beta_k \mathbf{g}_k \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \gamma_k \mathbf{q}_{k+1} \end{cases} \quad (20)$$

Utilizando a mesma CLF, $\mathbf{V}_g(k)$ dada pela equação (7), pode-se mostrar facilmente que as escolhas

$$\alpha_k = \frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R} \mathbf{p}_k \rangle} \quad (21)$$

$$\beta_k = -\frac{\langle \mathbf{g}_k, \mathbf{R} \mathbf{q}_k \rangle}{\langle \mathbf{q}_k, \mathbf{R} \mathbf{q}_k \rangle} \quad (22)$$

$$\gamma_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R} \mathbf{q}_{k+1} \rangle}{\langle \mathbf{q}_{k+1}, \mathbf{R} \mathbf{q}_{k+1} \rangle} \quad (23)$$

resultam em

$$\Delta V_g = -\frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{R}\mathbf{p}_k \rangle} < 0 \quad (24)$$

$$\begin{aligned} \|\mathbf{q}_{k+1}\|_{\mathbf{R}}^2 &= \|\mathbf{g}_k\|_{\mathbf{R}}^2 - \frac{\langle \mathbf{g}_k, \mathbf{R}\mathbf{q}_k \rangle^2}{\langle \mathbf{q}_k, \mathbf{R}\mathbf{q}_k \rangle} \\ &< \|\mathbf{g}_k\|_{\mathbf{R}}^2 \end{aligned} \quad (25)$$

$$\begin{aligned} \|\mathbf{p}_{k+1}\|_{\mathbf{R}}^2 &= \|\mathbf{g}_{k+1}\|_{\mathbf{R}}^2 - \frac{\langle \mathbf{g}_{k+1}, \mathbf{R}\mathbf{q}_{k+1} \rangle^2}{\langle \mathbf{q}_{k+1}, \mathbf{R}\mathbf{q}_{k+1} \rangle} \\ &< \|\mathbf{g}_{k+1}\|_{\mathbf{R}}^2 \end{aligned} \quad (26)$$

respectivamente.

O método de projeto de controlador Proporcional Integral (PI) proposto resulta no algoritmo 2, apresentado na forma de pseudocódigo. Note que os parâmetros α_k , β_k e γ_k do método são também determinados através de CLF's sem utilizar nenhuma propriedade dos vetores \mathbf{g}_k , \mathbf{p}_k e \mathbf{q}_k . Isto sugere que, em aplicações nas quais não se pode garantir a propriedade básica $\mathbf{p}_0 = \mathbf{g}_0$ (que, por sua vez garante a ortogonalidade dos resíduos), seria vantajoso utilizar os métodos variantes CG-CLF e PI-CLF, ao invés de utilizar o CG padrão, que poderia divergir.

Algoritmo 2

Calcular $\mathbf{r}_0 = \mathbf{b} - \mathbf{R}\mathbf{w}_0$, $\mathbf{p}_0 = \mathbf{g}_0$, $\mathbf{q}_0 \neq \mathbf{g}_0$

Para $k = 0, 1, \dots$, até convergência

$$\alpha_k = \frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}\mathbf{p}_k \rangle}$$

$$\beta_k = -\frac{\langle \mathbf{g}_k, \mathbf{R}\mathbf{q}_k \rangle}{\langle \mathbf{q}_k, \mathbf{R}\mathbf{q}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{g}_{k+1} = \mathbf{g}_k - \alpha_k \mathbf{R}\mathbf{p}_k$$

$$\mathbf{q}_{k+1} = \mathbf{g}_k + \beta_k \mathbf{q}_k$$

$$\gamma_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}\mathbf{q}_{k+1} \rangle}{\langle \mathbf{q}_{k+1}, \mathbf{R}\mathbf{q}_{k+1} \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \gamma_k \mathbf{q}_{k+1}$$

Fim

O algoritmo 2 apresenta um produto matriz/vetor a mais que o algoritmo 1. Porém, o pseudocódigo pode ser reorganizado notando que $\beta_k = \gamma_{k-1}$, portanto $\mathbf{q}_{k+1} = \mathbf{g}_k + \gamma_{k-1} \mathbf{q}_k = \mathbf{p}_k$. Isto implica que $\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \gamma_k \mathbf{p}_k$ e $\gamma_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}\mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}\mathbf{p}_k \rangle}$. Neste caso, o algoritmo 2 corresponde ao método do gradiente conjugado (algoritmo 1), a menos da inicialização. Em geral, os algoritmos 2 e 1 apresentam o mesmo desempenho porém, podem existir casos em que essa diferença na inicialização resulte em uma diferença no desempenho dos dois algoritmos. Vale ressaltar que o método CG pode então ser interpretado como um sistema dinâmico constituído pela planta linear $\{\mathbf{I}, \mathbf{I}, \mathbf{R}, \mathbf{0}\}$ com um controlador proporcional e integral variante no tempo em realimentação unitária.

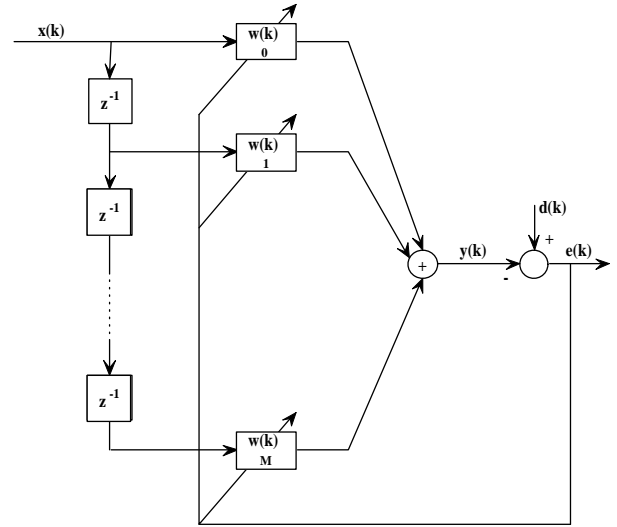


Figura 4: Filtro adaptativo de resposta finita ao impulso.

4 APLICAÇÃO EM FILTRAGEM ADAPTATIVA

O problema de filtragem adaptativa consiste basicamente em determinar o valor atual de um sinal variante no tempo, a partir de valores passados deste mesmo, corrompidos por um ruído (também variante no tempo). A realização mais básica de um filtro adaptativo é obtido através da forma direta de filtro digital de resposta finita ao impulso (FIR), como ilustrado na figura 4, cuja saída é dada por

$$y(k) = \sum_{i=0}^M w_i(k)x(k-i) = \mathbf{w}^T(k)\mathbf{x}(k) \quad (27)$$

onde $\mathbf{x}(k) = [x(k) \ x(k-1) \ \dots \ x(k-M)]^T$ e $\mathbf{w}(k) = [w_0(k) \ w_1(k) \ \dots \ w_M(k)]^T$ são os vetores de entrada e de pesos respectivamente, e M é a dimensão do tap do filtro (número de atrasadores). Uma das funções objetivo mais utilizadas para ajustar os pesos de um filtro adaptativo é o erro médio quadrático definido como

$$\begin{aligned} F(e(k)) &= \xi(k) = E[e^2(k)] \\ &= E[d^2(k) - 2d(k)y(k) + y^2(k)] \end{aligned} \quad (28)$$

onde $e(k) = d(k) - y(k)$, $y(k)$ é a saída do filtro adaptativo e $d(k)$ é o sinal desejado (Haykin, 1991). Neste caso, a função objetivo pode ser reescrita como

$$\begin{aligned} E[e^2(k)] &= \xi(k) \\ &= E[d^2(k) - 2d(k)\mathbf{w}^T(k)\mathbf{x}(k) + \mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k)] \\ &= E[d^2(k)] - 2E[d(k)\mathbf{w}^T(k)\mathbf{x}(k)] + E[\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k)] \end{aligned} \quad (29)$$

Portanto, para um filtro com coeficientes fixos, a função do erro médio quadrático é dada por

$$\xi = E[d^2(k)] - 2\mathbf{w}^T \mathbf{b} + \mathbf{w}^T \mathbf{R} \mathbf{w} \quad (30)$$

onde $\mathbf{b} = E[d(k)\mathbf{x}(k)]$ é o vetor de correlação cruzada entre a resposta desejada e o sinal de entrada, e $\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^T(k)]$ é a matriz de autocorrelação do sinal de entrada. Como a função objetivo ξ é uma função quadrática do vetor de pesos \mathbf{w} , o problema de minimizar ξ é equivalente ao problema de achar \mathbf{w} tal que

$$\mathbf{g}_w = \nabla \xi = \frac{\partial \xi}{\partial \mathbf{w}} = -2\mathbf{b} + 2\mathbf{R}\mathbf{w} = \mathbf{0}. \quad (31)$$

Pela equação (31) e notando que $\frac{\partial^2 \xi}{\partial \mathbf{w}^2} = \mathbf{R} > 0$, pode-se dizer que o vetor \mathbf{w}^* tal que $\mathbf{R}\mathbf{w}^* = \mathbf{b}$ é o argumento que minimiza a função objetivo ξ . Em outras palavras, \mathbf{w}^* é solução de

$$\min_{\mathbf{w}} \{\xi = E[d^2(k)] - 2\mathbf{w}^T \mathbf{b} + \mathbf{w}^T \mathbf{R} \mathbf{w}\}. \quad (32)$$

Para resolver o problema da equação (32), métodos iterativos tais como o método CG podem ser utilizados, como mostrado em Boray e Srinath (1992) e Chang e Willson (2000). Em Boray e Srinath (1992) as matrizes \mathbf{R} e \mathbf{b} são estimadas utilizando uma janela deslizante de dados. Neste caso o gradiente da função objetivo é estimado por uma média de n_w valores passados conforme a seguinte equação

$$\mathbf{g}_w(k) = \left(\frac{2}{n_w} \right) \sum_{j=k-n_w+1}^k \{\mathbf{w}^T(j)\mathbf{x}(k) - d(k)\}\mathbf{x}(k). \quad (33)$$

Note que para cada amostra k serão necessários $\min(M, n_w)$ iterações para a convergência do algoritmo CG. Portanto, a escolha de n_w , a janela sobre qual a média do gradiente da função custo é avaliada, tem um impacto direto sobre a convergência dos pesos do filtro. Desta forma, um aumento na dimensão do *tap* e/ou dos dados de entrada causa diretamente um aumento no custo computacional do método, podendo ultrapassar o custo computacional da inversão da matriz \mathbf{R} . No trabalho de Chang e Willson (2000), uma modificação é feita utilizando uma janela de dados exponencialmente decrescente. Esta modificação permite realizar apenas uma iteração a cada amostragem, porém para manter a convergência é necessário a introdução de um fator η , determinado experimentalmente, no cálculo do parâmetro α_k , e do método Polak-Ribiere para o cálculo do parâmetro β_k . Neste trabalho mostra-se através de simulações que, utilizando os métodos desenvolvidos por CLF's, não é necessário a introdução de nenhum fator empírico e/ou de nenhum outro método não linear para garantir a convergência dos métodos em ambientes variantes no tempo.

Utilizando a janela de dados exponencialmente decrescente, proposta por Chang e Willson (2000), as funções de autocorrelação e de correlação cruzada são dadas por

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T \quad (34)$$

$$\mathbf{b}_{k+1} = \lambda_f \mathbf{b}_k + d_{k+1} \mathbf{x}_{k+1} \quad (35)$$

onde λ_f é o fator de esquecimento. Para um processamento amostra por amostra, pode-se achar uma formula recursiva do vetor de resíduo utilizando as equações (2), (34) e (35):

$$\begin{aligned} \mathbf{g}_{k+1} &= \mathbf{b}_{k+1} - \mathbf{R}_{k+1} \mathbf{w}_{k+1} \\ &= \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \\ &\quad + \mathbf{x}_{k+1} (d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k). \end{aligned} \quad (36)$$

Introduzindo as equações (34), (35) e (36) nos algoritmos 1 e 2, obtêm-se as versões para filtragem adaptativa dos algoritmos CG e PI, apresentadas nos algoritmos 3 e 4.

Algoritmo 3 Método CG-CLF para filtros adaptativos

Calcular $\mathbf{R}_0 = \mathbf{x}_0 \mathbf{x}_0^T$, $\mathbf{b}_0 = d_0 \mathbf{x}_0$

Calcular $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$, $\mathbf{p}_0 = \mathbf{g}_0$

Para $k = 0, 1, \dots, N-1$

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T$$

$$\alpha_k = \frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} (d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k)$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k$$

Fim

Algoritmo 4 Método PI para filtros adaptativos

Calcular $\mathbf{R}_0 = \mathbf{x}_0 \mathbf{x}_0^T$, $\mathbf{b}_0 = d_0 \mathbf{x}_0$

Calcular $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$, $\mathbf{p}_0 = \mathbf{g}_0$, $\mathbf{q}_0 \neq \mathbf{g}_0$

Para $k = 0, 1, \dots, N-1$

$$\beta_k = -\frac{\langle \mathbf{g}_k, \mathbf{R}_k \mathbf{q}_k \rangle}{\langle \mathbf{q}_k, \mathbf{R}_k \mathbf{q}_k \rangle}$$

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T$$

$$\alpha_k = \frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{p}_k + \mathbf{x}_{k+1} (d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k)$$

$$\mathbf{q}_{k+1} = \mathbf{g}_k + \beta_k \mathbf{q}_k$$

$$\gamma_{k+1} = -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}_{k+1} \mathbf{q}_{k+1} \rangle}{\langle \mathbf{q}_{k+1}, \mathbf{R}_{k+1} \mathbf{q}_{k+1} \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \gamma_{k+1} \mathbf{q}_{k+1}$$

Fim

Note que o algoritmo 3 não introduz nenhum custo computacional adicional em relação ao algoritmo de Chang e Willson (2000) enquanto o algoritmo 4 envolve a computação adicional de um produto matriz/vetor.

5 SIMULAÇÕES

Várias simulações são feitas utilizando as três configurações básicas (Sayed, 2003) e (Haykin, 1991): Equalização, Identificação de sistema e Predição linear. Note que, em todos os casos simulados neste artigo, a diferença entre as variações do método CG (algoritmos 1 e 2) não resulta em uma

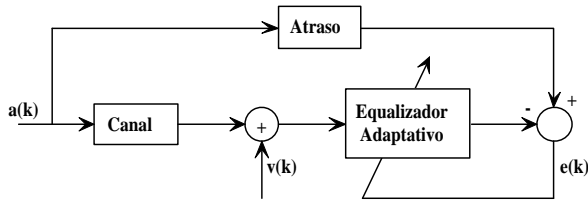


Figura 5: Diagrama de blocos do equalizador adaptativo.

diferença significativa no desempenho dos algoritmos 3 e 4. Portanto, apenas o desempenho do algoritmo 3 será apresentado.

5.1 Equalizador adaptativo

A figura 5 mostra o diagrama de blocos do equalizador adaptativo transversal utilizado para equalizar a distorção introduzida por um canal de banda limitada. O exemplo é tirado de Haykin (1991). A entrada do sistema é uma seqüência aleatória $\{a(k)\}$ com valores ± 1 , portanto de média nula. A seqüência é distorcida por um canal de resposta em frequência dada por

$$\begin{aligned} h(k) &= \frac{1}{2}[1 + \cos(2\pi(k-2)/W)], & k = 1, 2, 3 \\ &= 0, & k > 3 \end{aligned} \quad (37)$$

onde W controla a amplitude da distorção no canal. Um aumento em W causa um aumento da distorção do canal e resulta no aumento do espalhamento ρ dos autovalores da matriz de autocorrelação \mathbf{R} das entradas do equalizador. O canal é corrompido por um vetor de ruído branco aditivo de média nula e variância σ^2 . O equalizador é composto por 11 *taps*. Como a resposta ao impulso do canal é simétrica em relação a $k = 2$, os pesos dos *taps* do equalizador são simétricos em relação ao sexto *tap*. Desta forma, a entrada $\{a(k)\}$ do canal é atrasada por 7 amostras para gerar a resposta desejada do equalizador. A experiência é feita com dois valores de W , 2.9 e 3.5, que resultam em espalhamentos ρ de \mathbf{R} iguais a 6.07 e 46.821 respectivamente. Nas figuras 6, 7, 8, e 9 mostra-se as *curvas de aprendizado*, gráfico do erro instantâneo quadrático $e^2(k) = (d_k - y_k)^2$ (onde d_k é o sinal desejado e y_k é a saída do equalizador) em função da iteração k , dos equalizadores dos algoritmos 3, Polak-Ribiere (Chang e Willson, 2000), CG padrão (Boray e Srinath, 1992) e RLS (Haykin, 1991) respectivamente. Os parâmetros escolhidos neste exemplo são tais que $\lambda_f = \eta = 0.99$ e $\sigma^2 = 0.001$ e o *ensemble* é constituído por 200 amostras independentes. A tabela 2 apresenta os valores de regime para quais converge o erro médio quadrático dos equalizadores em função de W após aproximadamente 40 iterações. Nota-se que os dois métodos propostos neste trabalho apresentam um desempenho

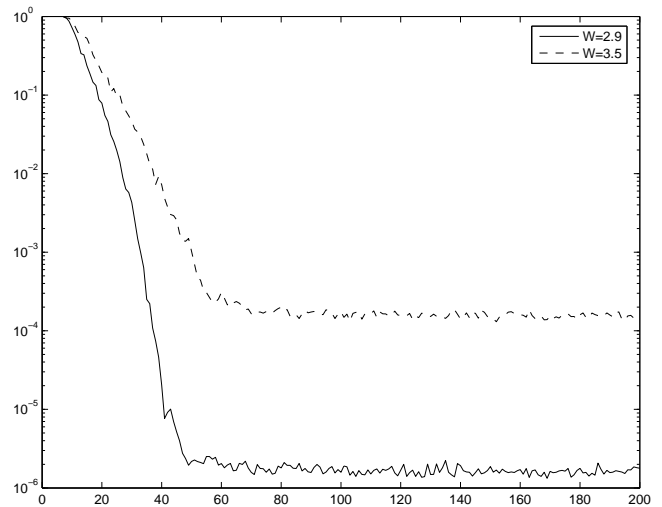


Figura 6: Curva de aprendizado do equalizador adaptativo utilizando o método CG determinado por CLF's. O erro médio quadrático converge para valores de regime iguais a 1.74×10^{-6} e 1.68×10^{-4} para $W = 2.9$ e $W = 3.5$ respectivamente. Note que a taxa de convergência não depende de quanto o sinal de entrada é perturbado.

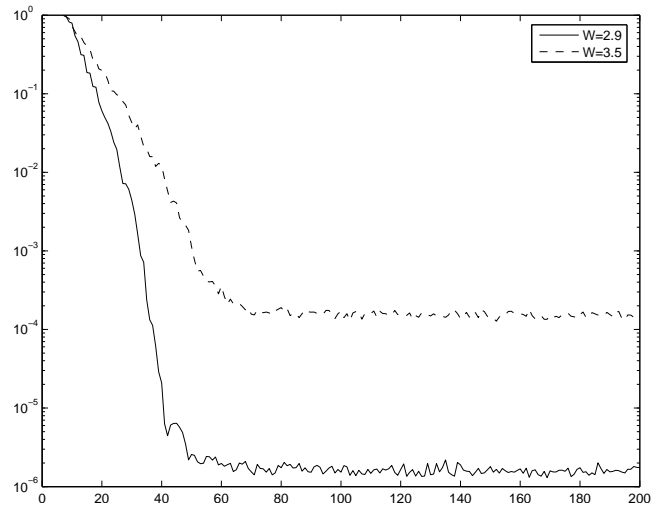


Figura 7: Curva de aprendizado do equalizador adaptativo utilizando o método CG Polak Ribiere de Chang e Willson (2000). O erro médio quadrático converge para valores de regime iguais a 1.69×10^{-6} e 1.63×10^{-4} para $W = 2.9$ e $W = 3.5$ respectivamente. Note que a taxa de convergência não depende da amplitude da perturbação.

comparável ao método de Chang e Willson (2000) e ao método RLS. No caso do método CG padrão, como é realizado apenas uma iteração para cada par $(\mathbf{R}_k, \mathbf{b}_k)$, seria necessário

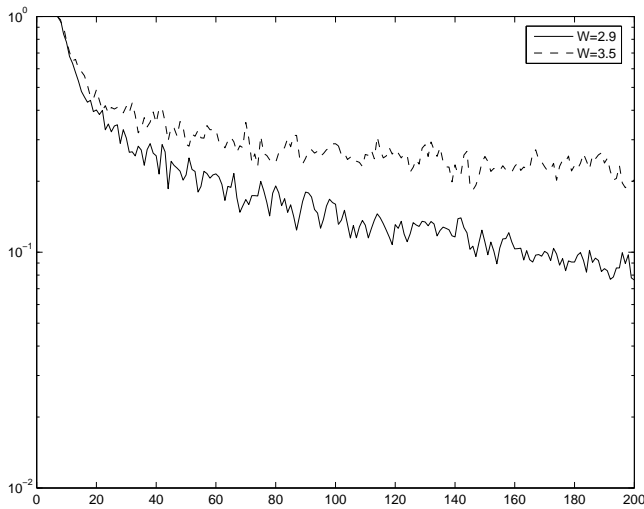


Figura 8: Curva de aprendizado do equalizador adaptativo utilizando o método CG padrão de Boray e Srinath (1992). O erro médio quadrático converge para valores de regime iguais a 1.36×10^{-1} e 2.62×10^{-1} para $W = 2.9$ e $W = 3.5$ respectivamente. Note que a taxa de convergência não depende da amplitude da perturbação.

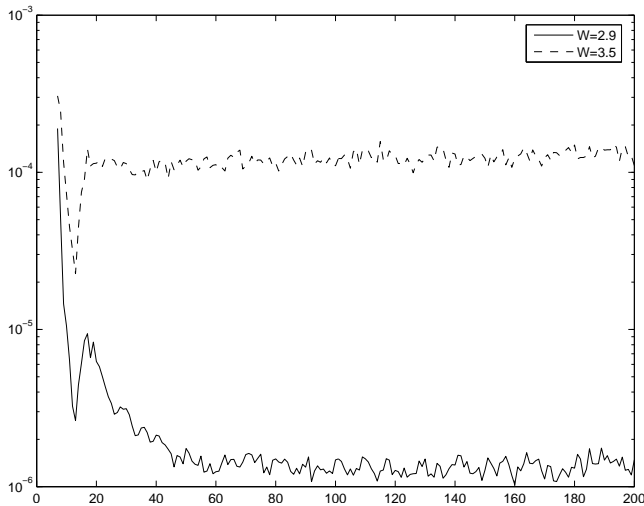


Figura 9: Curva de aprendizado do equalizador adaptativo utilizando o método RLS Haykin (1991). O erro médio quadrático converge para valores de regime iguais a 1.40×10^{-6} e 1.20×10^{-4} para $W = 2.9$ e $W = 3.5$ respectivamente. Note que a taxa de convergência não depende da amplitude da perturbação.

fazer $\mathbf{p}_k = \mathbf{g}_k$ em cada iteração (isto corresponde a reiniciar o método CG tal que $\mathbf{p}_0 = \mathbf{g}_0$ a cada iteração k) para garantir a ortogonalidade dos resíduos gerados. Portanto, no-

Equalizador	$W = 2.9$	$W = 3.5$
CG-CLF	1.74×10^{-6}	1.68×10^{-4}
SD-CLF	1.82×10^{-6}	3.70×10^{-3}
Polak-Ribière	1.69×10^{-6}	1.63×10^{-4}
CG	1.36×10^{-1}	2.62×10^{-1}
RLS	1.40×10^{-6}	1.20×10^{-4}

Tabela 2: Erro médio quadrático dos equalizadores em função de W . Os métodos propostos neste trabalho apresentam um desempenho comparável ao método de Chang e Willson (2000) e ao método RLS.

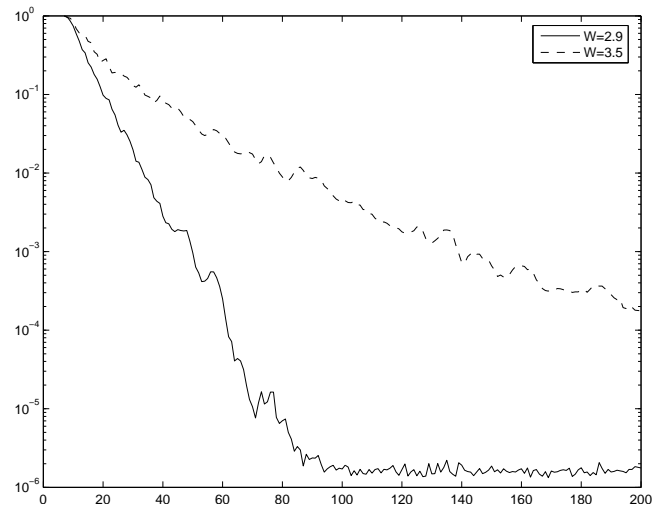


Figura 10: Curva de aprendizado do equalizador adaptativo utilizando o método SD. O erro médio quadrático converge para valores de regime iguais a 1.82×10^{-6} e 3.70×10^{-3} para $W = 2.9$ e $W = 3.5$ respectivamente. Note que a taxa de convergência não depende de quanto o sinal de entrada é perturbado.

tando que $\mathbf{p}_k = \mathbf{g}_k + \beta_{k-1} \mathbf{p}_{k-1}$, pode-se dizer que ocorre uma perda de ortogonalidade dos resíduos ($\mathbf{g}_{k+1}^T \mathbf{g}_k \neq 0$) do método CG padrão (Hestenes e Stiefel, 1952) utilizado em Boray e Srinath (1992), causando um desempenho pior do método.

A partir desta constatação, propõe-se uma outra modificação do método CG no algoritmo 5. A modificação consiste em substituir o ultimo passo do método CG por $\mathbf{p}_{k+1} = \mathbf{g}_{k+1}$, isto equivale a substituir no algoritmo todo o vetor \mathbf{p}_k por \mathbf{g}_k . Vale ressaltar que esta nova modificação do método CG resulta no método *Steepest Descent* (SD). A figura 10 e a tabela 2 apresentam respectivamente a curva de aprendizagem e os valores de regime do método SD. Pode-se notar que o SD-CLF e os outros métodos convergem para o mesmo valor

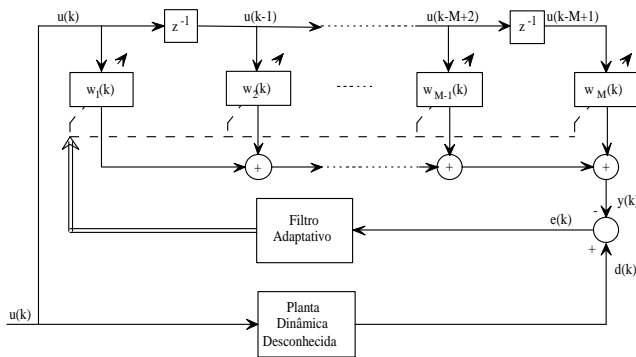


Figura 11: Identificação de uma planta dinâmica desconhecida por filtro adaptativo com *tap* atrasador. Os dados de entrada u e os pesos do *tap* w são reais (Haykin, 1991).

de erro médio quadrático em regime, porém a taxa de convergência do método SD-CLF é menor e é inversamente proporcional ao espalhamento ρ dos autovalores da matriz de autocorrelação (quanto maior ρ , menor será a taxa de convergência). Vale ressaltar que, mesmo que tenha uma taxa de convergência menor que o método RLS em um ambiente variante no tempo, o método SD-CLF pode ser utilizado em casos onde o método LMS (que possui uma taxa de convergência comparável, mas com erro médio quadrático de regime maior e custo computacional maior) é aplicável.

Algoritmo 5 Método SD para filtros adaptativos

Calcular $\mathbf{R}_0 = \mathbf{x}_0 \mathbf{x}_0^T$, $\mathbf{b}_0 = d_0 \mathbf{x}_0$

Calcular $\mathbf{g}_0 = \mathbf{b}_0 - \mathbf{R}_0 \mathbf{w}_0$

Para $k = 0, 1, \dots, N - 1$

$$\alpha_k = \frac{\langle \mathbf{g}_k, \mathbf{g}_k \rangle}{\langle \mathbf{g}_k, \mathbf{R}_{k+1} \mathbf{g}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{g}_k$$

$$\mathbf{R}_{k+1} = \lambda_f \mathbf{R}_k + \mathbf{x}_{k+1} \mathbf{x}_{k+1}^T$$

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \mathbf{R}_{k+1} \mathbf{g}_k + \mathbf{x}_{k+1} (d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k)$$

Fim

Também note que o algoritmo 5 tem menos passos e um custo computacional menor que os algoritmos de Chang e Willson (2000), Boray e Srinath (1992) e RLS.

5.2 Identificação de Sistema

Na configuração de identificação de sistema considera-se que a planta desconhecida é um filtro de resposta ao impulso finita (FIR) de ordem 20 e o sinal de entrada é um ruído branco Gaussiano de variância $\sigma^2 = 1$. A figura 11 mostra o diagrama de blocos da configuração de identificação de sistema. As figuras 12, 13, 14, 15, e 16 mostram as curvas de aprendizado dos filtros adaptativos utilizando respectiva-

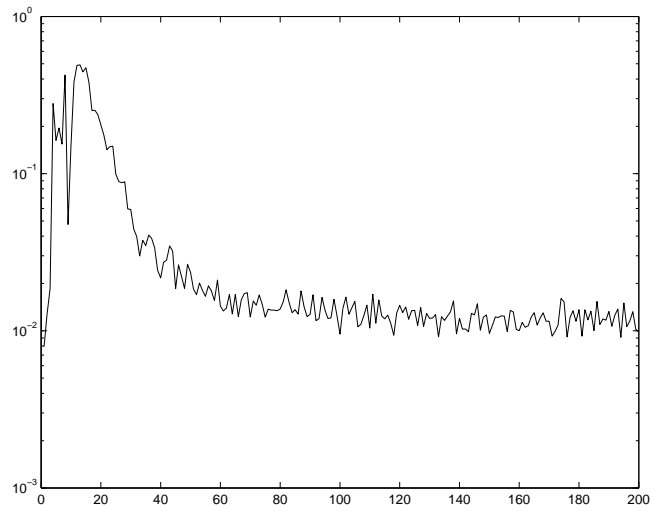


Figura 12: Curva de aprendizado do filtro adaptativo na configuração de identificação de sistema utilizando o método CG determinado por CLF's. O erro médio quadrático converge para um valor de regime igual a 1.35×10^{-2} .

mente os métodos dos algoritmos 3, 5, Polak-Ribiere (Chang e Willson, 2000), CG padrão (Boray e Srinath, 1992), e RLS (Haykin, 1991) na configuração de identificação de sistema. Neste caso o *ensemble* é constituído por 100 amostras independentes. A tabela 3 apresenta os valores apresenta os valores de regime para quais converge o erro médio quadrático dos filtros adaptativos após aproximadamente 40 iterações. Como no caso do equalizador adaptativo, nota-se que os métodos propostos neste trabalho e o método Polak-Ribiere de Chang e Willson (2000) apresentam um desempenho comparável ao método RLS. Por outro lado a perda de convergência do método CG padrão de Boray e Srinath (1992), devido à perda de ortogonalidade dos resíduos, é mais acentuada na configuração de identificação de sistema.

Filtro Adaptativo	Erro Médio Quadrático
CG-CLF	1.35×10^{-2}
SD-CLF	1.35×10^{-2}
Polak-Ribière	1.36×10^{-2}
CG	5.05×10^{-1}
RLS	6.90×10^{-3}

Tabela 3: Erro médio quadrático dos filtros adaptativos na configuração de identificação de sistema. Os métodos propostos neste trabalho apresentam um desempenho comparável ao método de Chang e Willson (2000) e ao método RLS.

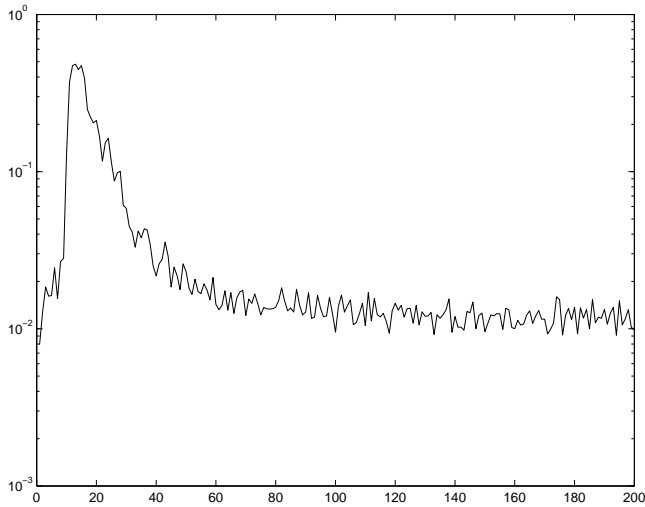


Figura 13: Curva de aprendizado do filtro adaptativo na configuração de identificação de sistema utilizando o método SD determinado por CLF's. O erro médio quadrático converge para um valor de regime igual a 1.35×10^{-2} .

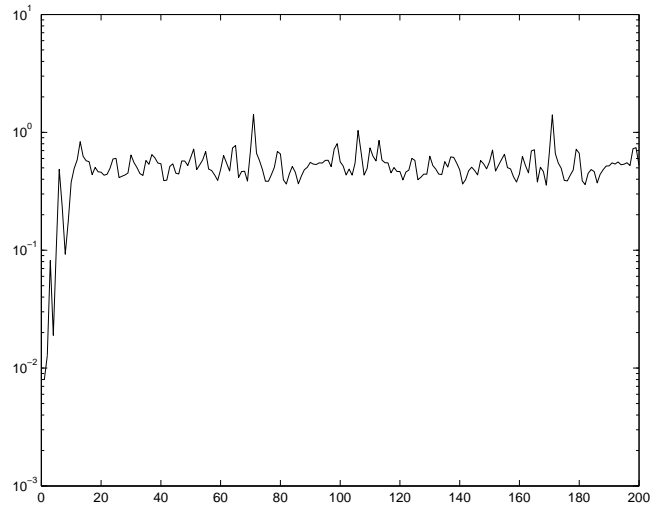


Figura 15: Curva de aprendizado do filtro adaptativo na configuração de identificação de sistema utilizando o método CG padrão de Boray e Srinath (1992). O erro médio quadrático converge para um valor de regime igual a 5.05×10^{-1} .

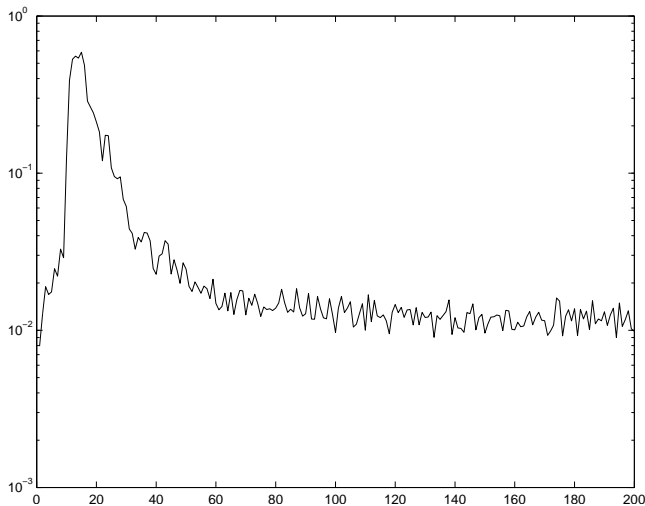


Figura 14: Curva de aprendizado do filtro adaptativo na configuração de identificação de sistema utilizando o método CG Polak Ribiere de Chang e Willson (2000). O erro médio quadrático converge para um valor de regime igual a 1.36×10^{-2} .

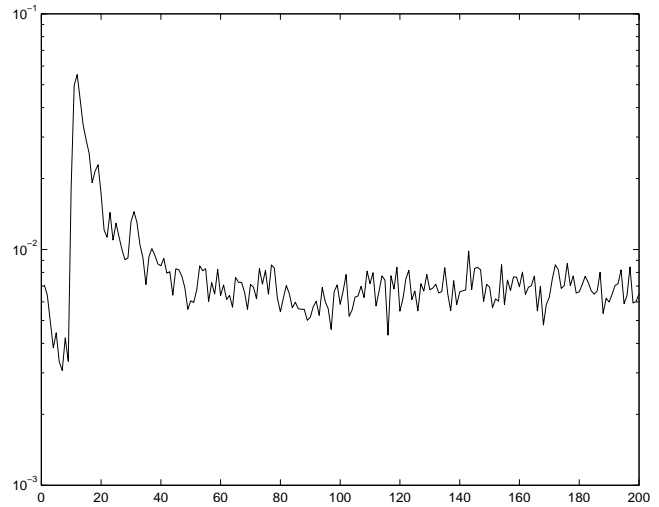


Figura 16: Curva de aprendizado do filtro adaptativo na configuração de identificação de sistema utilizando o método RLS (Haykin, 1991). O erro médio quadrático converge para um valor de regime igual a 6.90×10^{-3} .

5.3 Predição Linear

A terceira configuração de filtro adaptativo simulada neste trabalho é o cenário de predição linear. A figura 17 mostra a configuração do filtro adaptativo de 2 taps no cenário de predição linear. As entradas dos taps $u(k-1)$ e $u(k-2)$ são extraídas do processo autoregressivo (AR) de valores reais

dada pela seguinte equação

$$u(k) + a_1u(k-1) + a_2u(k-2) = v(k) \quad (38)$$

onde $v(k)$ é um ruído branco de média nula e variância σ_v^2 . Os valores de a_1 e a_2 são determinados pelo espalhamento dos autovalores da matriz de autocorrelação dos dados de entrada. Para uma tentativa do experimento, são obtidas 500

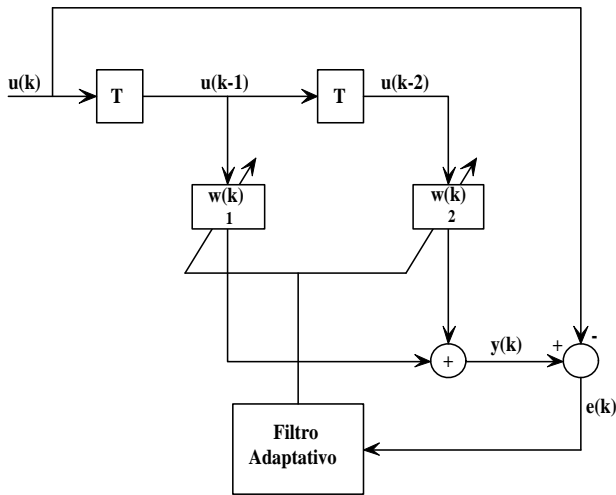


Figura 17: Diagrama de blocos de um filtro adaptativo na configuração de predição linear.

amostras do ruído branco $\{v(i)\}$ a partir de um gerador de números aleatórios de média nula e variância ajustável. O *ensemble* é constituído por 200 tentativas independentes e o experimento é feito para valores de espalhamento $\rho = 3$ e $\rho = 10$. As figuras 18, 19, 20, 21, e 22 apresentam as curvas de aprendizagem dos filtros adaptativos no cenário de predição linear. A tabela 4 apresenta os valores de regime para quais converge o erro quadrático dos preditores lineares. Neste caso também, nota-se que os métodos propostos neste artigo apresentam propriedades de convergência comparáveis aos métodos Polak-Ribière de Chang e Willson (2000) e RLS. Vale ressaltar que a convergência lenta do método SD-CLF, que depende do espalhamento dos autovalores, observado no caso do equalizador adaptativo, não ocorre. Para o método CG padrão, a perda de ortogonalidade resulta em uma perda de convergência para pequenos valores de espalhamento. A tabela 5 compara a porcentagem de desajuste definida como

$$M = \frac{J_{\infty} - J_{min}}{J_{min}}, \quad (39)$$

onde J_{∞} é o valor esperado do erro médio quadrático em regime e J_{min} é o erro médio quadrático mínimo (Boray e Srinath, 1992).

6 CUSTO COMPUTACIONAL

Para determinar o custo computacional dos métodos considerados neste artigo, determina-se o número de operações básicas (adição, subtração, multiplicação e divisão) de ponto flutuante (FLOP) realizadas. A tabela 7 apresenta a comparação do custo computacional dos novos métodos propostos

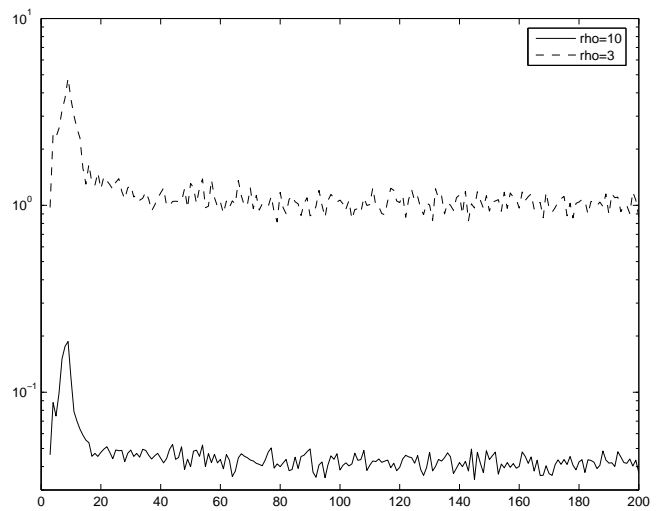


Figura 18: Curva de aprendizado do filtro adaptativo na configuração de predição linear utilizando o método CG determinado por CLF's. O erro médio quadrático converge para valores de regime iguais a 4.27×10^{-2} e 1.02 para $\rho = 10$ e $\rho = 3$ respectivamente.

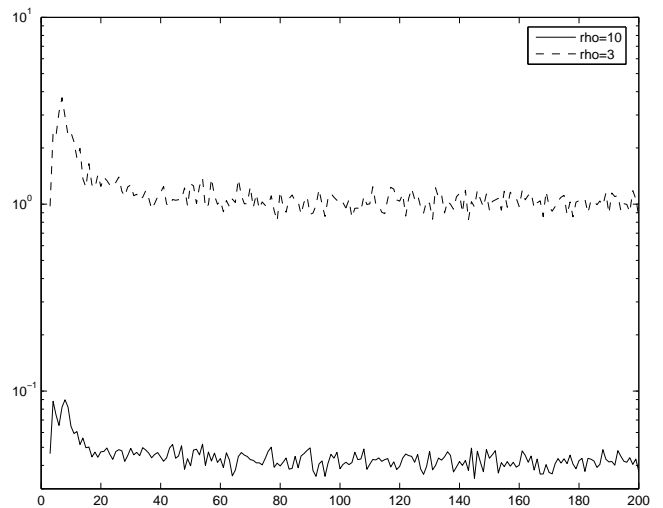


Figura 19: Curva de aprendizado do filtro adaptativo na configuração de predição linear utilizando o método SD determinado por CLF's. O erro médio quadrático converge para valores de regime iguais a 4.27×10^{-2} e 1.02 para $\rho = 10$ e $\rho = 3$ respectivamente.

neste artigo com os métodos RLS (algoritmo 6), CG padrão de Boray e Srinath (1992) e Polak-Ribière de Chang e Willson (2000).

Como já dito anteriormente, o método SD-CLF é o método

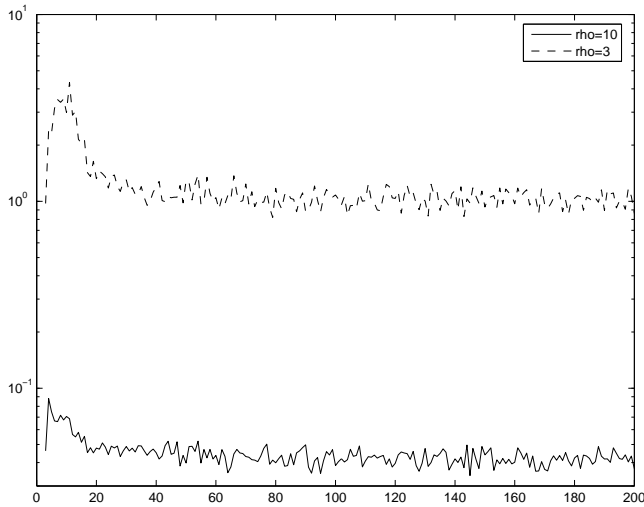


Figura 20: Curva de aprendizado do filtro adaptativo na configuração de predição linear utilizando o método CG Polak Ribiere de Chang e Willson (2000). O erro médio quadrático converge para valores de regime iguais a 4.28×10^{-2} e 1.02 para $\rho = 10$ e $\rho = 3$ respectivamente.

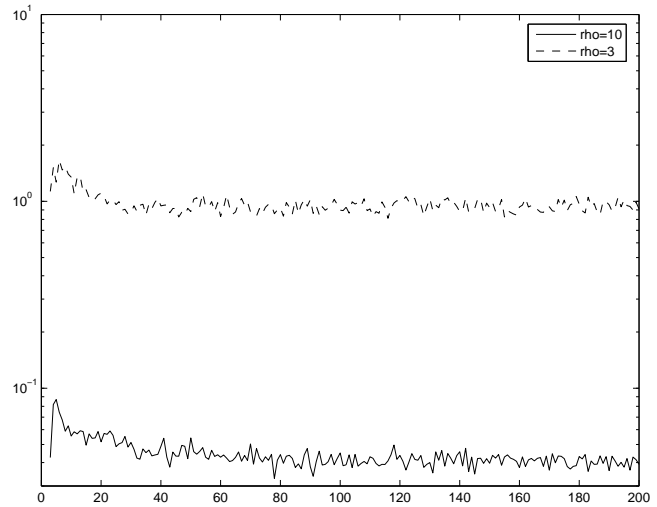


Figura 22: Curva de aprendizado do filtro adaptativo na configuração de predição linear utilizando o método RLS (Haykin, 1991). O erro médio quadrático converge para valores de regime iguais a 4.08×10^{-2} e 0.95 para $\rho = 10$ e $\rho = 3$ respectivamente.

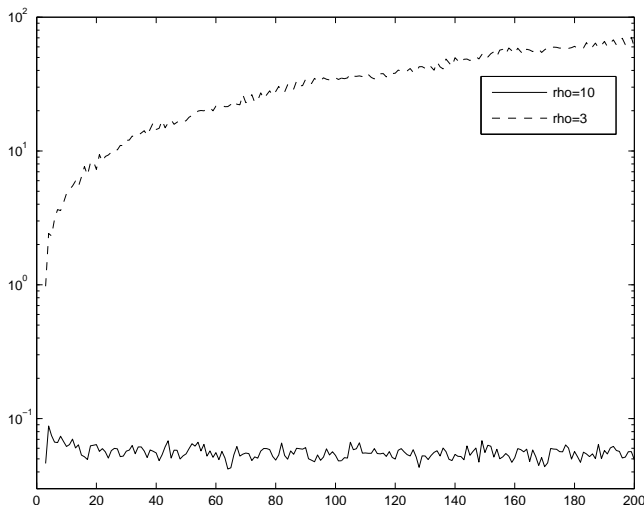


Figura 21: Curva de aprendizado do filtro adaptativo na configuração de predição linear utilizando o método CG padrão de Boray e Srinath (1992). O erro médio quadrático converge para valores de regime iguais a 5.55×10^{-2} e 80.38 para $\rho = 10$ e $\rho = 3$ respectivamente.

Preditor linear	$\rho = 10$	$\rho = 3$
CG-CLF	4.27×10^{-2}	1.02
SD-CLF	4.27×10^{-2}	1.02
Polak-Ribière	4.28×10^{-2}	1.02
CG	5.55×10^{-2}	80.38
RLS	4.08×10^{-2}	0.95

Tabela 4: Erro médio quadrático dos filtros adaptativos na configuração de predição linear. Os métodos propostos neste trabalho apresentam um desempenho comparável ao método de Chang e Willson (2000) e ao método RLS.

ρ	J_{\min}	CG-CLF	SD-CLF	P.R.	CG
3	0.0731	12.95	12.95	12.95	1098.59
10	0.0322	0.33	0.33	0.33	0.72

Tabela 5: Porcentagem de Desajuste $M(\%)$

que possui menos etapas e portanto tem custo computacional menor. O método CG padrão apresenta um custo computacional menor que o método CG-CLF, com uma diferença correspondente ao custo do cálculo de um produto interno ($2M - 1$). Os métodos Polak-Ribière de Chang e Willson

(2000) e RLS, por realizarem mais operações com vetores e/ou matrizes, apresentam um custo computacional maior que os métodos propostos neste artigo.

7 CONCLUSÃO

Neste artigo, os métodos iterativos de resolução de sistemas lineares de equações são analisados do ponto de vista da teoria de controle. O problema da resolução da equação li-

Método	FLOP
CG-CLF	$5M^2 + 17M - 2$
SD-CLF	$5M^2 + 13M - 2$
Polak-Ribière	$5M^2 + 20M - 2$
RLS	$6M^2 + 6M + 2$
CG	$5M^2 + 15M - 1$

Tabela 6: Comparação do custo computacional dos métodos.

Algoritmo 6 Método RLS para filtros adaptativos

$\mathbf{w}_0 = \mathbf{0}$

Calcular $\mathbf{P}_0 = \delta^{-1}\mathbf{I}$

Para $k = 0, 1, \dots, N - 1$

$$\mathbf{v}_{k+1} = \frac{\lambda^{-1}\mathbf{P}_k\mathbf{x}_{k+1}}{1 + \lambda^{-1}\mathbf{x}_{k+1}^T\mathbf{P}_k\mathbf{x}_{k+1}}$$

$$\alpha_{k+1} = d_{k+1} - \mathbf{w}_k^T\mathbf{x}_{k+1}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{v}_{k+1}\alpha_{k+1}$$

$$\mathbf{P}_{k+1} = \lambda^{-1}\mathbf{P}_k - \lambda^{-1}\mathbf{v}_{k+1}\mathbf{x}_{k+1}^T\mathbf{P}_k$$

Fim

near $\mathbf{R}\mathbf{w} = \mathbf{b}$ é transformado em um problema de regulação de sinal e, o processo iterativo é representado por um sistema dinâmico constituído pela planta linear $\{\mathbf{I}, \mathbf{I}, \mathbf{R}, \mathbf{0}\}$ e um controlador em malha fechada. Esta formulação permite desenvolver métodos iterativos robustos utilizando técnicas de controle do tipo PI/PD com parâmetros variantes no tempo e funções de Liapunov para determinar esses parâmetros. Vale ressaltar que, com a ordenação adequada das etapas e na presença de ortogonalidade dos resíduos, o método PI corresponde ao método CG padrão com uma inicialização diferente.

Estes métodos iterativos são aplicados na determinação dos pesos ótimos de filtros adaptativos em cenários de equalização, identificação de sistemas e predição linear como já feito em Boray e Srinath (1992) e Chang e Willson (2000) com o método CG. Chang e Willson (2000) analisam um sistema linear variante no tempo, entretanto utilizam a formula Polak-Ribière, desenvolvida para funções não lineares gerais, para determinar um dos parâmetros necessários para a convergência. Dentro desta perspectiva, este artigo propõe uma análise que utiliza a estrutura linear do sistema variante no tempo ($\mathbf{R}\mathbf{w} = \mathbf{b}$) e a estrutura bilinear específica dos sistemas que surgem na análise do método CG aplicado a esta situação. Mostra-se através de várias simulações que os métodos propostos neste artigo apresentam um desempenho comparável aos métodos RLS e Chang e Willson (2000). Uma das vantagens dos métodos propostos, em relação aos dois supracitados, é o fato de determinar os pesos ótimos dos filtros sem determinar explicitamente a inversa da matriz de autocorrelação \mathbf{R} e sem utilizar técnicas não lineares e/ou parâmetros determinados empiricamente. A prova sistemática da convergência dos métodos iterativos para sistemas lineares va-

riantes no tempo sob hipóteses adequadas é um tópico sob investigações.

AGRADECIMENTOS

Esta pesquisa foi parcialmente financiada pelos projetos 141566/2004-3, 551863/2002-1 do CNPq, e também contou com o apoio das agências CAPES e FAPERJ.

REFERÊNCIAS

- Bhaya, A. e Kaszkurewicz, E. (2004). Análise e projeto de métodos iterativos vistos como sistemas dinâmicos com controle realimentado, *CBA Gramado: RS*.
- Boray, G. K. e Srinath, M. D. (1992). Conjugate gradient techniques for adaptive filtering, *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications* **39**: 1–10.
- Chang, P. S. e Willson, A. N. (2000). Analysis of conjugate gradient algorithms for adaptive filtering, *IEEE Transactions on Signal Processing* **48**: 409–418.
- Diene, O. (2004). *Métodos iterativos lineares: Análise e projeto através de ferramentas da teoria de controle*, Master's thesis, Universidade Federal do Rio de Janeiro.
- Haykin, S. (1991). *Adaptive Filter Theory*, Prentice Hall, Englewood Cliffs, New Jersey 07632.
- Hestenes, M. R. (1980). *Conjugate Direction Methods in Optimization*, Springer-Verlag, New York.
- Hestenes, M. R. e Stiefel, E. (1952). Methods of conjugate gradients for solving linear systems, *Journal of Research of the National Bureau of Standards* **49**: 409–436.
- Khalil, H. K. (2002). *Nonlinear Systems*, Prentice Hall, Upper Saddle River, New Jersey 07458.
- Saad, Y. (1996). *Iterative methods for sparse linear systems*, PWS Publishing Company, 20 Park Plaza Boston, MA 02116.
- Saad, Y. e VanDerVorst, H. A. (2000). Iterative solution of linear systems in the 20th century, *J. Computational and Applied Mathematics* **122**: 1–33.
- Sayed, A. H. (2003). *Fundamentals of Adaptive Filtering*, John Wiley and Sons, 111 River Street, Hoboken, New Jersey 07030.
- Slotine, J. J. E. e Li, W. (1991). *Applied nonlinear control*, Prentice Hall, Englewood Cliffs, New Jersey.
- VanDerVorst, H. A. (2000). Krylov subspace iteration, *Computing in Science and Engineering* **2**: 32–37.

LINEARLY CONSTRAINED ADAPTIVE FILTERING ALGORITHMS DESIGNED USING CONTROL LIAPUNOV FUNCTIONS

OUMAR DIENE*, AMIT BHAYA*

**Universidade Federal do Rio de Janeiro
COPPE - Programa de Engenharia Elétrica
Núcleo de Atendimento em Computação de Alto Desempenho
Cidade Universitária - Ilha do Fundão
21.945-970 - Rio de Janeiro - R. J.*

Emails: oumar@coep.ufrj.br, amit@nacad.ufrj.br

Abstract— The standard conjugate gradient (CG) method uses orthogonality of the residues to simplify the formulas for the parameters necessary for convergence. In adaptive filtering, the sample-by-sample update of the correlation matrix and the cross-correlation vector causes a loss of the residue orthogonality in a modified online algorithm, which, in turn, results in loss of convergence and an increase of the filter quadratic mean error. This paper extends a recently proposed optimality and convergence proof of the degenerated CG method to the case of linearly constrained adaptive filtering, and proposes a constrained Steepest Descent (CSD) method.

Keywords— Linearly constrained adaptive filtering algorithms, single-user DS-CDMA, control Liapunov function, conjugate gradient method, steepest descent method.

1 Introduction

The Conjugate Gradient (CG) is considered to be one of the best iterative methods for linear systems of equations with symmetric positive definite (spd) coefficient matrices (Saad, 1996). The standard CG method uses orthogonality of the residues to simplify the formulas for the parameters necessary for convergence (Hestenes and Stiefel, 1952). In applications such as adaptive filtering, the sample-by-sample update of the correlation matrix and the cross-correlation vector causes a loss of the residue orthogonality in a modified online algorithm. This loss of orthogonality results in loss of convergence and an increase of the filter quadratic mean error (Diene and Bhaya, 2006). Boray and Srinath (1992) proposed a minimization of the output-error energy of an adaptive finite impulse response (FIR) filter by using the CG method with a sliding data window. In this case, several CG iterations are made at each sample. Chang and Willson (2000) proposed an alternative degenerated CG method in which an exponentially decaying window performs only one iteration of the CG method at each sample and leads to an algorithm with performance comparable to the RLS algorithm. Using the recently proposed control Liapunov (CLF) analysis of the CG method viewed as a dynamic system in the standard feedback configuration (Bhaya and Kaszkurewicz, 2006), the optimality of the parameters and the convergence of the degenerated CG method were proved, and a Steepest Descent (SD) method was proposed in Diene and Bhaya (2006). The constrained version of the degenerated CG algorithm of Chang and Willson (2000) was proposed, in Apolinário et al. (2000), for the minimization of the output-error energy of an adaptive

FIR filter subject to a set of linear constraints, i.e., $\min_{\mathbf{w}} E[e^2]$ subject to $\mathbf{C}^T \mathbf{w} = \mathbf{f}$, where \mathbf{w} is the length M coefficient vector, e is the filter output error, \mathbf{C} is the $M \times p$ constraint matrix, and \mathbf{f} is the length p gain vector. This paper extends the optimality and convergence proof of the degenerated CG method (Diene and Bhaya, 2006) to the case of linearly constrained adaptive filtering, and proposes a constrained Steepest Descent (CSD) method. The main technique consists of an appropriate choice of a CLF, and has been used in Bhaya and Kaszkurewicz (2003) and Diene and Bhaya (2004) to analyze and design Krylov and new iterative methods.

2 Preliminaries

Consider the linear system

$$\mathbf{R}\mathbf{w} = \mathbf{b}. \quad (1)$$

Iterative methods for solving the system (1) can be described by the following equation

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \quad (2)$$

where \mathbf{w}_k is an approximation of the solution and \mathbf{u}_k is a correction calculated in order to increase the accuracy of the approximated solution \mathbf{w}_{k+1} at each iteration (Saad, 1996), and then drive the residue, defined as $\mathbf{g}_k = \mathbf{b} - \mathbf{R}\mathbf{w}_k$, to zero in a finite number of iterations. The problem of solving (1) by iterative methods can be represented by the signal regulation problem:

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{u}_k \\ \mathbf{y}_k = \mathbf{R}\mathbf{w}_k \\ \mathbf{g}_k = \mathbf{b} - \mathbf{y}_k \\ \mathbf{u}_k = f(\mathbf{g}_k), \end{cases} \quad (3)$$

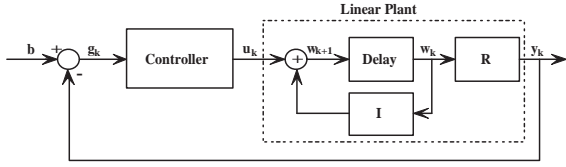


Figure 1: Block diagram representing the regulation problem equivalent to resolution (1) of by iterative methods. The problem is to design a controller such that $\lim_{k \rightarrow \infty} \mathbf{g}_k = \mathbf{0} \Leftrightarrow \lim_{k \rightarrow \infty} \mathbf{R}\mathbf{w}_k = \mathbf{b}$.

i.e., designing control signal \mathbf{u}_k in order to zero the steady state error by choice of the control law $f(\cdot)$ i.e., $\lim_{k \rightarrow \infty} \mathbf{g}_k = \mathbf{0}$, or, equivalently, $\lim_{k \rightarrow \infty} \mathbf{R}\mathbf{w}_k = \mathbf{b}$.

System (3), represented in block diagram form in figure 1, models an iterative method as a feedback control system (plant $\{\mathbf{I}, \mathbf{I}, \mathbf{R}, \mathbf{0}\}$ with controller in unitary feedback configuration). In control theory, both classical and modern, there are various techniques of solving the signal regulation problem (3) representing the solution of (1), which correspond to different choices of the controller $f(\cdot)$ in (3). However, as shown in Diene (2004), many of these techniques (pole allocation, state feedback) result in controllers that explicitly use the inverse matrix \mathbf{R}^{-1} . In Bhaya and Kaszkurewicz (2003) and Bhaya and Kaszkurewicz (2006), the authors show that, by using appropriate quadratic CLFs, choosing a proportional derivative controller for $f(\cdot)$ in (3) leads to the CG method, shown in algorithm 1. The CLF method proposed in Bhaya and Kaszkurewicz (2003) and Bhaya and Kaszkurewicz (2006) gives an idea of the robustness of the CG method, since it does not use the assumption $\mathbf{p}_0 = \mathbf{g}_0$ (initialization of conjugate directions), used to obtain residue orthogonality (see, e.g. Hestenes and Stiefel (1952)). This indicates that, in situations where the assumption does not hold, the determination of CG parameters by CLF still ensures the decrease of some norm of the residual vector. When the assumption $\mathbf{p}_0 = \mathbf{g}_0$ holds, it can be shown that the parameters determined by CLFs are the same as those commonly found in the literature (Hestenes and Stiefel, 1952, p.411, eqs 3:1b,3:1e).

Algorithm 1 *Hestenes-Stiefel CG algorithm for constant Linear Systems*

Calculate $\mathbf{g}_0 = \mathbf{b} - \mathbf{R}\mathbf{w}_0$, $\mathbf{p}_0 = \mathbf{g}_0$

For $k = 0, 1, \dots$, until convergence

$$\begin{aligned} \alpha_k &= \frac{\langle \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}\mathbf{p}_k \rangle} \\ \mathbf{w}_{k+1} &= \mathbf{w}_k + \alpha_k \mathbf{p}_k \\ \mathbf{g}_{k+1} &= \mathbf{g}_k - \alpha_k \mathbf{R}\mathbf{p}_k \\ \beta_k &= -\frac{\langle \mathbf{g}_{k+1}, \mathbf{R}\mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{R}\mathbf{p}_k \rangle} \\ \mathbf{p}_{k+1} &= \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k \end{aligned}$$

End

3 Designing new Linearly Constrained Adaptive Filtering Algorithms

The problem of linearly constrained adaptive filters can be formulated as follows:

$$\begin{aligned} &\text{Solve } \mathbf{R}_k \mathbf{w}_k = \mathbf{b}_k, \\ &\text{subject to } \mathbf{C}^T \mathbf{w}_k = \mathbf{f} \end{aligned} \quad (4)$$

where $\mathbf{b}_k = E[d_k \mathbf{x}_k]$ is the length M cross-correlation vector between the scalar desired response d_k and the length M input signal \mathbf{x}_k , $\mathbf{R}_k = E[\mathbf{x}_k \mathbf{x}_k^T]$ is the $M \times M$ correlation matrix of the input signal, \mathbf{w}_k is the length M coefficient vector, \mathbf{C} is the $M \times p$ constraint matrix, and \mathbf{f} is the length p gain vector. To solve this problem, a structure was presented in Griffiths and Jim (1982) in which only a small set of coefficients are updated, and this set is confined to the subspace orthogonal to the space spanned by the constraint matrix \mathbf{C} . This strategy, referred to as the generalized sidelobe cancelation (GSC), is able to transform the linearly constrained minimization problem into an unconstrained minimization problem, and subsequently any adaptation algorithm can be used (Apolinário et al., 2000). Use of GSC consists of solving the linear unconstrained minimization problem:

$$\mathbf{R}'_k \mathbf{w}'_k = \mathbf{b}'_k, \quad (5)$$

where $\mathbf{b}'_k = E[d_k \mathbf{x}'_k]$, $\mathbf{R}'_k = E[\mathbf{x}'_k \mathbf{x}'_k{}^T]$, $\mathbf{x}'_k = \mathbf{B}^T \mathbf{x}_k$, \mathbf{B} is the blocking matrix such that $\mathbf{B}^T \mathbf{C} = \mathbf{0}$. The last step is to determine the solution of the constrained problem (4) using equation (6) below

$$\mathbf{w}_k = \mathbf{F} - \mathbf{B}\mathbf{w}'_k, \quad (6)$$

where $\mathbf{F} = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{f}$. Notice that equation (6) guarantees that $\mathbf{C}^T \mathbf{w}_k = \mathbf{f}$. Using the exponentially decaying data window, proposed in Chang and Willson (2000), the recurrences of the auto-correlation matrix \mathbf{R}'_k and the cross-correlation vector \mathbf{b}'_k are given by:

$$\mathbf{R}'_{k+1} = \lambda_f \mathbf{R}'_k + \mathbf{x}'_{k+1} \mathbf{x}'_{k+1}{}^T \quad (7)$$

$$\mathbf{b}'_{k+1} = \lambda_f \mathbf{b}'_k + d_{k+1} \mathbf{x}'_{k+1} \quad (8)$$

where $\lambda_f < 1$ is the forgetting factor. For sample-by-sample processing, a recursive formulation for the residual vector can be found by using (2), (7) and (8), resulting in

$$\mathbf{g}'_{k+1} = \lambda_f \mathbf{g}'_k - \alpha_k \mathbf{R}'_{k+1} \mathbf{p}'_k + \mathbf{x}'_{k+1} (d_{k+1} - \mathbf{x}'_{k+1}{}^T \mathbf{w}'_k). \quad (9)$$

Then, for the unconstrained minimization problem (5), the system (3) can be rewritten as

$$\begin{cases} \mathbf{w}'_{k+1} = \mathbf{w}'_k + \alpha_k \mathbf{p}'_k \\ \mathbf{R}'_{k+1} = \lambda_f \mathbf{R}'_k + \mathbf{x}'_{k+1} \mathbf{x}'_{k+1}{}^T \\ \mathbf{g}'_{k+1} = \lambda_f \mathbf{g}'_k - \alpha_k \mathbf{R}'_{k+1} \mathbf{p}'_k + \mathbf{x}'_{k+1} (d_{k+1} - \mathbf{x}'_{k+1}{}^T \mathbf{w}'_k) \\ \mathbf{p}'_{k+1} = \mathbf{g}'_{k+1} + \beta_k \mathbf{p}'_k \end{cases} \quad (10)$$

Choosing a blocking matrix \mathbf{B} such that $\mathbf{B}^T \mathbf{B} = \mathbf{I}$, it can be seen that

$$\mathbf{R}'_k = \mathbf{B}^T \mathbf{R}_k \mathbf{B} \quad (11)$$

Then, multiplying both sides of (9) by \mathbf{B} and choosing

$$\mathbf{g}_k = \mathbf{B} \mathbf{g}'_k \quad (12)$$

$$\mathbf{p}_k = \mathbf{B} \mathbf{p}'_k, \quad (13)$$

it can be shown that (Apolinário et al., 2000)

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k - \bar{\mathbf{x}}_{k+1} e_{k+1} \quad (14)$$

where

$$\bar{\mathbf{R}}_{k+1} = \mathbf{P} \mathbf{R}_{k+1} \mathbf{P} = \mathbf{B} \mathbf{R}'_k \mathbf{B}^T \quad (15)$$

$$\bar{\mathbf{x}}_{k+1} = \mathbf{P} \mathbf{x}_{k+1} \quad (16)$$

$$\mathbf{P} = \mathbf{B} \mathbf{B}^T \quad (17)$$

$$e_{k+1} = d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k \quad (18)$$

In order to design new constrained algorithms, it is necessary to determine the system of equations representing the equivalent regulation problem (3). This can be done noting that, from (7), (15), (10), (12) and (13) it is straightforward to show that

$$\bar{\mathbf{R}}_{k+1} = \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^T \quad (19)$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k \quad (20)$$

The weight updating equation (6) can be rewritten as

$$\mathbf{w}_{k+1} = \mathbf{F} - \mathbf{B} \mathbf{w}'_{k+1} = \mathbf{P} \mathbf{w}_k + \mathbf{F} - \alpha_k \mathbf{p}_k \quad (21)$$

Then, for the constrained minimization problem (4), the system (3) can be rewritten as

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \mathbf{F} - \alpha_k \mathbf{p}_k \\ e_{k+1} = d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k \\ \bar{\mathbf{R}}_{k+1} = \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^T \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k - \bar{\mathbf{x}}_{k+1} e_{k+1} \\ \mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k. \end{cases} \quad (22)$$

For the system (22), the main result of this paper is stated as follows:

Theorem 1 *The following assumption is made:*

$$\inf_k \lambda_{\min}(\bar{\mathbf{R}}_k^{-1}) = \zeta_1 > 0, \quad \inf_k \lambda_{\min}(\mathbf{R}_k) = \zeta_2 > 0 \quad (23)$$

With this assumption, the choices

$$\alpha_k = \frac{\langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \quad (24)$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \quad (25)$$

are optimal and ensure that the equilibrium of the system (10) is asymptotically stable (i.e. $\mathbf{g}_k \rightarrow 0$ and $\mathbf{p}_k \rightarrow 0$).

Proof: It is desired that $e_{k+1} \rightarrow 0$. Thus (14) can be rewritten as

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k. \quad (26)$$

Choose the control Liapunov function $V_{\mathbf{g}_k}(k) = \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle = \|\mathbf{g}_k\|_{\bar{\mathbf{R}}_k^{-1}}^2$, then

$$\Delta V_{\mathbf{g}} = \langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1}^{-1} \mathbf{g}_{k+1} \rangle - \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle. \quad (27)$$

Substituting (26) into (27) results in

$$\begin{aligned} \Delta V_{\mathbf{g}} &= \lambda_f^2 \langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1}^{-1} \mathbf{g}_k \rangle - 2\alpha_k \langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle \\ &\quad + \alpha^2 \langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle - \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle \end{aligned} \quad (28)$$

The optimal value of α_k (which makes $\Delta V_{\mathbf{g}}$ as negative as possible), given by (24), is determined calculating $\frac{\partial \Delta V_{\mathbf{g}}}{\partial \alpha_k}$ and setting it to zero. Substituting (24) into (28) yields

$$\begin{aligned} \Delta V_{\mathbf{g}} &= -\frac{\langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} + \lambda_f^2 \langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1}^{-1} \mathbf{g}_k \rangle \\ &\quad - \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle \end{aligned} \quad (29)$$

Using the matrix inversion lemma (Haykin, 1991), one can write

$$\bar{\mathbf{R}}_{k+1}^{-1} = \lambda_f^{-1} \bar{\mathbf{R}}_k^{-1} - c \lambda_f^{-2} \bar{\mathbf{R}}_k^{-1} \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^T \bar{\mathbf{R}}_k^{-1}, \quad (30)$$

where $c = (1/(1 + \lambda_f^{-1} \bar{\mathbf{x}}_{k+1}^T \bar{\mathbf{R}}_k^{-1} \bar{\mathbf{x}}_{k+1})) > 0$. Inserting this into (29) results in

$$\begin{aligned} \Delta V_{\mathbf{g}} &= -\frac{\langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} - (1 - \lambda_f) \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle \\ &\quad - c \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^T \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle. \end{aligned} \quad (31)$$

Then, from hypothesis (23), it follows that $\lambda_f < 1$ implies that

$$\Delta V_{\mathbf{g}} < -\zeta_1 \|\mathbf{g}_k\|_2^2 < 0, \quad (32)$$

which implies that $\|\mathbf{g}_k\|_{\bar{\mathbf{R}}_k^{-1}}$ is a decreasing sequence.

For the choice of β_k , choose the CLF $V_{\mathbf{p}_k}(k) = \langle \mathbf{p}_k, \bar{\mathbf{R}}_k \mathbf{p}_k \rangle = \|\mathbf{p}_k\|_{\bar{\mathbf{R}}_k}^2$. Then,

$$\begin{aligned} \Delta V_{\mathbf{p}} &= \langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{g}_{k+1} \rangle + 2\beta_k \langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle \\ &\quad + \beta_k^2 \langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle - \langle \mathbf{p}_k, \bar{\mathbf{R}}_k \mathbf{p}_k \rangle. \end{aligned}$$

In this case, the optimal value of β_k , given by (25), is obtained calculating $\frac{\partial \Delta V_{\mathbf{p}}}{\partial \beta_k}$ and setting it to zero, yielding

$$\begin{aligned} V_{\mathbf{p}_{k+1}}(k+1) &= \|\mathbf{g}_{k+1}\|_{\bar{\mathbf{R}}_{k+1}}^2 - \frac{\langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle} \\ &< \|\mathbf{g}_{k+1}\|_{\bar{\mathbf{R}}_{k+1}}^2. \end{aligned} \quad (33)$$

From (32), the equivalence of norms and hypothesis (23), it can be concluded that $\|\mathbf{g}_{k+1}\|_{\bar{\mathbf{R}}_{k+1}}$ is a decreasing sequence. Thus (33) implies that \mathbf{p}_{k+1} decreases in $\bar{\mathbf{R}}_{k+1}$ -norm, not necessarily monotonically, concluding the proof. \square

Observation 1 Notice that equation (31) can be used to determine the value of λ_f which makes $\Delta V_{\mathbf{g}}$ as negative as possible: using the expression (31) and solving the unidimensional minimization problem $\min_{\lambda_f} \Delta V_{\mathbf{g}}$ allows, in theory, the determination of an optimal value of λ_f .

Observation 2 Also notice that for the constrained algorithm, the projection matrix $\mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T$, thus it is not necessary to determine explicitly the blocking matrix \mathbf{B} .

The pseudocode of the extension of the Hestenes-Stiefel CG method for linearly constrained adaptive filtering algorithms, based on theorem 1 and referenced in this paper as CCG-CLF, is presented in algorithm 2.

Algorithm 2 CCG-CLF method for linearly constrained adaptive filters

Calculate $\mathbf{w}_0 = \mathbf{F} = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{f}$

$\bar{\mathbf{R}}_0 = \mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T$

$\mathbf{g}_0 = \mathbf{p}_0 = \mathbf{0}$

For $k = 0, 1, \dots, N - 1$

$$\alpha_k = \frac{\langle \lambda_f \mathbf{g}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \mathbf{F} - \alpha_k \mathbf{p}_k$$

$$\bar{\mathbf{x}}_{k+1} = \mathbf{P} \mathbf{x}_{k+1}$$

$$\bar{\mathbf{R}}_{k+1} = \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^T$$

$$e_{k+1} = d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k$$

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{p}_k - \bar{\mathbf{x}}_{k+1} e_{k+1}$$

$$\beta_k = -\frac{\langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \bar{\mathbf{R}}_{k+1} \mathbf{p}_k \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{g}_{k+1} + \beta_k \mathbf{p}_k$$

End

Note that the algorithm in algorithm 2 does not introduce any additional computation and requires $3M$ floating point operations (FLOP) less than the algorithm in Apolinário et al. (2000). Choosing the ad hoc parameter η in Apolinário et al. (2000) equal to λ_f gives our formula for α_k which is shown above to be an optimal choice in the sense of making $\Delta V_{\mathbf{g}}$ as negative as possible, although, in nonstationary environments, a smaller value ($\eta < \lambda_f$ which keeps $\Delta V_{\mathbf{g}} < 0$) may be advisable.

In Diene and Bhaya (2006), a steepest descent (SD) method for adaptive filters was proposed. The results in Diene and Bhaya (2006) show that it is possible for the SD method for adaptive filters to have a performance comparable with the CG and RLS methods. The same extension, made above for the CG method, can be done for the SD method in order to design, by CLFs, a SD method for linearly constrained adaptive filters (CSD-CLF). The system of equations representing the regulation problem, equivalent to solving the linearly constrained adaptive filtering problem (4) by the CSD-CLF method, is given by system (34) below

$$\begin{cases} \mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \mathbf{F} - \alpha_k \mathbf{g}_k \\ e_{k+1} = d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k \\ \bar{\mathbf{R}}_{k+1} = \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^T \\ \mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k - \bar{\mathbf{x}}_{k+1} e_{k+1}. \end{cases} \quad (34)$$

For the system (34), another result of this paper is stated as follows:

Theorem 2 With assumption (23) valid, the choice

$$\alpha_k = \frac{\langle \lambda_f \mathbf{g}_k, \mathbf{g}_k \rangle}{\langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle} \quad (35)$$

is optimal and ensures that the equilibrium of the system (34) is asymptotically stable (i.e. $\mathbf{g}_k \rightarrow 0$).

Proof: Once again, it is desired that $e_{k+1} \rightarrow 0$. Thus substituting in (34) gives

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k. \quad (36)$$

Again choose the control Liapunov function $V_{\mathbf{g}_k}(k) = \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle = \|\mathbf{g}_k\|_{\bar{\mathbf{R}}_k}^2$, then

$$\Delta V_{\mathbf{g}} = \langle \mathbf{g}_{k+1}, \bar{\mathbf{R}}_{k+1}^{-1} \mathbf{g}_{k+1} \rangle - \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle. \quad (37)$$

Substituting (36) into (37) results in

$$\begin{aligned} \Delta V_{\mathbf{g}} &= \lambda_f^2 \langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1}^{-1} \mathbf{g}_k \rangle - 2\alpha_k \langle \lambda_f \mathbf{g}_k, \mathbf{g}_k \rangle \\ &\quad + \alpha^2 \langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle - \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle \end{aligned} \quad (38)$$

The optimal value of α_k (which makes $\Delta V_{\mathbf{g}}$ as negative as possible), given by (35), is determined calculating $\frac{\partial \Delta V_{\mathbf{g}}}{\partial \alpha_k}$ and setting it to zero. Substituting (35) into (38) yields

$$\begin{aligned} \Delta V_{\mathbf{g}} &= -\frac{\langle \lambda_f \mathbf{g}_k, \mathbf{g}_k \rangle^2}{\langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle} + \lambda_f^2 \langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1}^{-1} \mathbf{g}_k \rangle \\ &\quad - \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle \end{aligned} \quad (39)$$

Inserting equation (30) into (39) results in

$$\begin{aligned} \Delta V_{\mathbf{g}} &= -\frac{\langle \lambda_f \mathbf{g}_k, \mathbf{g}_k \rangle^2}{\langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle} - (1 - \lambda_f) \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle \\ &\quad - c \langle \mathbf{g}_k, \bar{\mathbf{R}}_k^{-1} \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^T \bar{\mathbf{R}}_k^{-1} \mathbf{g}_k \rangle. \end{aligned} \quad (40)$$

Then, from hypothesis (23), it follows that $\lambda_f < 1$ implies that

$$\Delta V_{\mathbf{g}} < -\zeta_1 \|\mathbf{g}_k\|_2^2 < 0, \quad (41)$$

which implies that $\|\mathbf{g}_k\|_{\bar{\mathbf{R}}_k}^2$ is a decreasing sequence. \square

The pseudocode of the extension of the SD method for linearly constrained adaptive filtering algorithms, based on theorem 2 and referenced in this paper as CSD-CLF, is presented in algorithm 3.

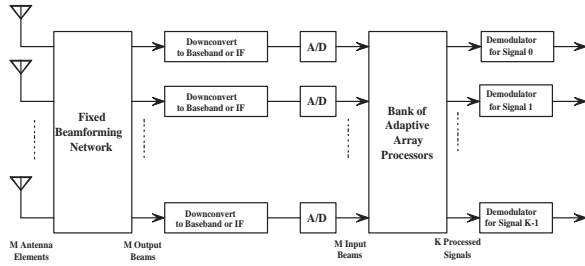


Figure 2: Simple model for a downlink synchronous transmission of K users. The structure of each array processor is shown in figure 3 below.

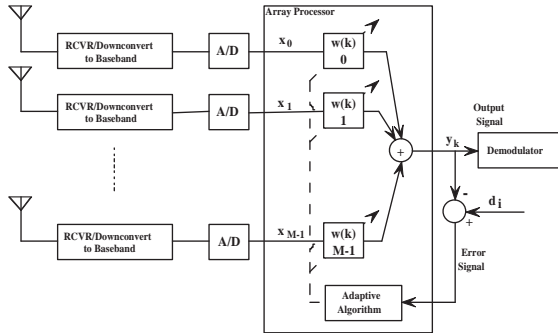


Figure 3: Adaptive array processor. Note that d_i represents an estimate or replica of the desired signal for the i^{th} user at the array output.

Algorithm 3 *CSD-CLF method for linearly constrained adaptive filters*

Calculate $\mathbf{w}_0 = \mathbf{F} = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{f}$

$\bar{\mathbf{R}}_0 = \mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T$

$\mathbf{g}_0 = \mathbf{0}$

For $k = 0, 1, \dots, N - 1$

$$\alpha_k = \frac{\langle \lambda_f \mathbf{g}_k, \mathbf{g}_k \rangle}{\langle \mathbf{g}_k, \bar{\mathbf{R}}_{k+1} \mathbf{g}_k \rangle}$$

$$\mathbf{w}_{k+1} = \mathbf{P} \mathbf{w}_k + \mathbf{F} - \alpha_k \mathbf{g}_k$$

$$\bar{\mathbf{x}}_{k+1} = \mathbf{P} \mathbf{x}_{k+1}$$

$$\bar{\mathbf{R}}_{k+1} = \lambda_f \bar{\mathbf{R}}_k + \bar{\mathbf{x}}_{k+1} \bar{\mathbf{x}}_{k+1}^T$$

$$e_{k+1} = d_{k+1} - \mathbf{x}_{k+1}^T \mathbf{w}_k$$

$$\mathbf{g}_{k+1} = \lambda_f \mathbf{g}_k - \alpha_k \bar{\mathbf{R}}_{k+1} \mathbf{g}_k - \bar{\mathbf{x}}_{k+1} e_{k+1}$$

End

Also note that algorithm 3 has 2 steps and $7M$ FLOP less than the algorithm in Apolinário et al. (2000).

4 Simulation Results

In order to test the proposed algorithms, we compare the performance of the proposed CLF algorithms with the algorithms in Apolinário et al. (2000), applied to a single-user detection in a DS-CDMA mobile communication system. For this experiment, we assumed a simple model for a downlink synchronous transmission of K users, as shown in figure 2 (Liberti and Rappaport, 1999). The received continuous-time signal is passed through a chip-matched filter and is sampled at a chip rate such that the received discrete time input-signal vector may be

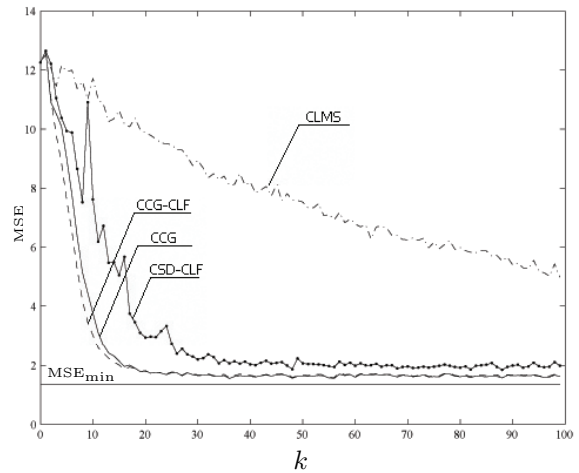


Figure 4: Learning curves of the linearly constrained adaptive filters applied to a single-user detection in a DS-CDMA mobile communication system.

expressed as $\mathbf{x}_k = \mathbf{S} \mathbf{A} \mathbf{c}_k + \mathbf{n}_k$ where $\mathbf{S} = [s_1 \ s_2 \ \dots \ s_K]$ is the spreading matrix containing the sampled spreading sequences of users 1 to K , $\mathbf{A} = \text{diag}[A_1 \ A_2 \ \dots \ A_K]$ contains the amplitudes of signals for each user, $\mathbf{c}_k = [c_{1k} \ c_{2k} \ \dots \ c_{Kk}]^T$ contains the information bits, and \mathbf{n}_k is the sampled noise sequence. For this example, the constraint is such that $\mathbf{C} = s_1$, and $\mathbf{f} = 1$. The number of users was set to $K = 5$, with Gold codes of length 7 used for spreading. The SNR for user one was made equal to 8 dB, and the relative power of interfering users was set to 20 dB, i.e., $10 \log(P_i/P_1) = 20$.

In order to compare the CLF algorithms with the algorithms in Apolinário et al. (2000), the same parameters are used. Figures 4 and 5 show the learning curves and the coefficient errors of the linearly constrained adaptive filters applied to a single-user detection in a DS-CDMA mobile communication system. It can be observed that the CLF methods proposed in this paper have convergence properties comparable with the degenerated CG algorithm used in Apolinário et al. (2000) and with the linearly constrained RLS algorithm. It is important to notice that, although it has a rate of convergence lower than that of the RLS algorithm in a noisy environment, the CSD-CLF method can be used in cases where the LMS algorithm (which has lower rate of convergence, higher steady state mean squared error and comparable computational cost) is applicable.

In order to compare the computational cost of the proposed algorithms with the CCG of Apolinário et al. (2000) and the CLMS, the number of basic floating point operations FLOP (addition, subtraction, multiplication and division) of the principal loop is determined. The results, presented in table 4, show that the CSD-CLF has less $4M$ FLOP than the CCG-CLF, which has less $3M$ FLOP than the CCG. The CLMS presents

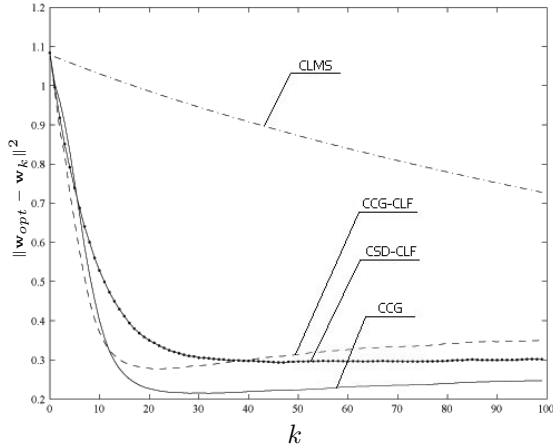


Figure 5: Coefficient errors of the linearly constrained adaptive filters applied to a single-user detection in a DS-CDMA mobile communication system.

Algorithm	FLOP
CSD-CLF	$9M^2 + 10M - 5$
CCG-CLF	$9M^2 + 14M - 5$
CCG	$9M^2 + 17M - 5$
CLMS	$4M^2 + 3M - 1$

Table 1: Computational cost comparisons.

a lower computational cost than the other algorithms because it does not compute the correlation matrix \mathbf{R}_{k+1} and the matrix/vector multiplication $\mathbf{R}_{k+1}\mathbf{g}_k$ or $\mathbf{R}_{k+1}\mathbf{p}_k$.

5 Conclusion

The control viewpoint (Bhaya and Kaszkurewicz, 2003; Bhaya and Kaszkurewicz, 2006) reveals that the formulas for the CG parameters α_k and β_k in Hestenes and Stiefel (1952, p.411, eqs 3:2a and 3:2b), actually enhance the robustness of the CG algorithm with respect to the standard equivalent choices ($\alpha_k = \langle \mathbf{g}_k, \mathbf{g}_k \rangle / \langle \mathbf{p}_k, \mathbf{R}\mathbf{p}_k \rangle$) and ($\beta_k = \langle \mathbf{g}_{k+1}, \mathbf{g}_{k+1} \rangle / \langle \mathbf{g}_k, \mathbf{g}_k \rangle$), since the latter make use of the assumption $\mathbf{p}_0 = \mathbf{g}_0$, to obtain orthogonality of residues. Hestenes and Stiefel (1952, p. 411) recognize that “*although these formulas are slightly more complicated, they have advantage that scale factors (introduced to increase accuracy) are more easily changed during the course of the computation*”. Our results, both theoretical and simulated, show that since the “more complicated formulas” correspond to optimal choices in the CLF method, they endow the system with robustness, since there is some leeway corresponding to neighboring non-optimal choices of the parameters α_k and β_k . The authors in Apolinário et al. (2000) have presented a constrained CG method for linearly constrained adaptive filtering problems, however their convergence analysis was carried out by numerical simulations. This paper proposes an analysis using the linear structure of

the time varying system (3) and the bilinear structure of the dynamical system solvers (22) and (34) considered, and the asymptotic stability of the algorithms is proved in theorems 1 and 2.

Acknowledgment

This research was partially financed by Project Nos 141566/2004-3, 551863/2002-1 of CNPq and also by the agencies CAPES and FAPERJ. The authors would like to thank Profs. Campos and Apolinário for making their CCG software available.

References

- Apolinário, J. A., de Campos, M. L. R. and Bernal, C. P. O. (2000). The constrained conjugate gradient algorithm, *IEEE Signal Processing Letters* **7**: 351–354.
- Bhaya, A. and Kaszkurewicz, E. (2003). Iterative methods as dynamical systems with feedback control, *Proc. 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, USA, pp. 2374–2380.
- Bhaya, A. and Kaszkurewicz, E. (2006). *Control Perspectives in Numerical Algorithms and Matrix Problems*, SIAM, Philadelphia.
- Boray, G. K. and Srinath, M. D. (1992). Conjugate gradient techniques for adaptive filtering, *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications* **39**: 1–10.
- Chang, P. S. and Willson, A. N. (2000). Analysis of conjugate gradient algorithms for adaptive filtering, *IEEE Transactions on Signal Processing* **48**: 409–418.
- Diene, O. (2004). *Linear iterative methods: Design and analysis using control theory tools*, Master’s thesis, COPPE/UFRJ. In Portuguese, www.coep.ufrj.br/~oumar/publication.html.
- Diene, O. and Bhaya, A. (2004). Linear iterative methods: Design and analysis using control theory tools, *Proc. 15th Brazilian Conference on Automatica*, Gramado, RS, Brazil. In Portuguese, www.coep.ufrj.br/~oumar/publication.html.
- Diene, O. and Bhaya, A. (2006). Adaptive filtering algorithms designed using control liapunov functions, *IEEE Signal Processing Letters* **4**: 224–227.
- Griffiths, L. J. and Jim, C. W. (1982). An alternative approach to linearly constrained adaptive beamforming, *IEEE Transactions on Antennas Propagation* **AP-30**: 27–34.
- Haykin, S. (1991). *Adaptive Filter Theory*, Prentice Hall, Englewood Cliffs, New Jersey.
- Hestenes, M. R. and Stiefel, E. (1952). Methods of conjugate gradients for solving linear systems, *Journal of Research of the National Bureau of Standards* **49**: 409–436.
- Liberti, J. C. and Rappaport, T. S. (1999). *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*, Prentice Hall, New Jersey.
- Saad, Y. (1996). *Iterative methods for sparse linear systems*, PWS Publishing Company, Boston.

Manuscript Number: NEUCOM-D-08-00143

Title: Perceptron Training Algorithms designed using Discrete-Time Control Liapunov Functions

Article Type: Full Length Article (FLA)

Keywords: Perceptron training algorithms, control Liapunov function, conjugate gradient method, steepest descent method.

Corresponding Author: Mr Oumar Diene, M.Sc.

Corresponding Author's Institution: Federal University of Rio de Janeiro

First Author: Oumar Diene, M.Sc.

Order of Authors: Oumar Diene, M.Sc.; Amit Bhaya, Ph.D.

Abstract: Perceptrons, proposed in the seminal paper McCulloch--Pitts of 1943, have remained of interest to neural network community because of their simplicity and usefulness in classifying linearly separable data and can be viewed as implementing iterative procedures for "solving" linear inequalities. Gradient descent and conjugate gradient methods, normally used for linear equalities, can be used to solve linear inequalities by simple modifications that have been proposed in the literature but not been analyzed completely. This paper takes a recently proposed control-inspired approach to the design of iterative perceptron training algorithms, by regarding certain training/algorithm parameters as controls and then using a control Liapunov technique to choose appropriate values of these parameters.

Perceptron Training Algorithms designed using Discrete-Time Control Liapunov Functions

Oumar Diene^{*}, Amit Bhaya

*Department of Electrical Engineering
NACAD-COPPE / Federal University of Rio de Janeiro
P.O Box 68504, 21945-970, Rio de Janeiro, RJ, BRAZIL
E-mails: oumar@ufrj.br, amit@nacad.ufrj.br
Tel.: +55-21-2562-8081; fax: +55-21-2290-6626*

Abstract

Perceptrons, proposed in the seminal paper McCulloch–Pitts of 1943, have remained of interest to neural network community because of their simplicity and usefulness in classifying linearly separable data and can be viewed as implementing iterative procedures for “solving” linear inequalities. Gradient descent and conjugate gradient methods, normally used for linear equalities, can be used to solve linear inequalities by simple modifications that have been proposed in the literature but not been analyzed completely. This paper applies a recently proposed control-inspired approach to the design of iterative steepest descent and conjugate gradient algorithms for perceptron training in batch mode, by regarding certain parameters of the training/algorithm as controls and then using a control Liapunov technique to choose appropriate values of these parameters.

Key words: Perceptron training algorithms, control Liapunov function, conjugate gradient method, steepest descent method.

^{*} Corresponding author

Email address: oumar@ufrj.br (Oumar Diene).

¹ This research was partially financed by Project Nos 141566/2004-3, 551863/2002-1 of CNPq and also by the agencies CAPES and FAPERJ.

1 Introduction

Perceptrons, proposed in the seminal paper McCulloch–Pitts of 1943, have remained of interest to neural network community because of their simplicity and usefulness in classifying linearly separable data. The recent focus on support vector machines, which can be viewed as an outgrowth of perceptrons, has also rekindled interest in the basic properties of perceptrons and their so-called training algorithms. Training a perceptron involves (recursively) solving a set of linear inequalities and several algorithms have been proposed in the literature.

The perceptron training rule is an error-correcting rule used to generate the weights and threshold for a linear threshold gate (perceptron) such that a linear decision surface is synthesized that discriminates between two linearly separable class samples clustered in feature space. The main advantage of the perceptron training rule is its low computational requirements and its ability to guarantee convergence to a solution for linearly separable problems [16]. The main drawback of this rule is the fact that it does not converge to useful approximate solutions for linearly nonseparable problems [7].

The Ho-Kahsyap algorithm determines a mean squared solution by using the pseudo-inverse of the pattern matrix [12]. The main drawbacks of this algorithm are the need to calculate the pseudo-inverse which could become unstable as the matrix $\mathbf{A}^T \mathbf{A}$ (\mathbf{A} is the pattern matrix) becomes close to singular, and the fact that it is computationally expensive for large scale problems. The adaptive versions of the Ho-Kashyap algorithm do not use the pseudo-inverse but require empirical learning parameters [10].

The Conjugate Gradient (CG) is considered to be one of the best iterative methods for linear systems of equations with symmetric positive definite (spd) coefficient matrices [17]. The standard CG method uses orthogonality of the residues to simplify the formulas for the parameters necessary for convergence [11]. Gradient descent and conjugate gradient are two widely used techniques for solving a set of linear inequalities. In finite precision implementation, the numerical errors could cause a loss of the residue orthogonality, which, in turn, results in loss of convergence [6]. A conjugate gradient algorithm for perceptron training is proposed in [13, 14], where the authors used heuristic techniques based on reinitialization of the CG method [9]. It should be pointed out that the rate of convergence of the conjugate gradient technique is greater than the gradient descent technique. However the CG technique requires more floating point operations, therefore, the overall computational cost (“time spent”) of the CG method is greater than the gradient descent techniques in some cases. This paper takes a control-inspired approach, recently proposed in [2, 3], to the design of iterative perceptron training algorithms, by regarding certain

training/algorithm parameters as controls and then using a control Liapunov technique to choose appropriate values of these parameters. It is observed that for the linearly nonseparable cases, the proposed algorithms converge to an approximated solution that classify correctly the maximum number of patterns.

2 Preliminaries

Consider the linear system

$$\mathbf{Ax} = \mathbf{b}. \quad (1)$$

Iterative methods for solving the system (1) can be described by the following equation

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k \quad (2)$$

where \mathbf{x}_k is an approximation of the solution and \mathbf{u}_k is a correction calculated in order to increase the accuracy of the approximated solution \mathbf{x}_k at each iteration [17], and then drive the residue, defined as $\mathbf{r}_k = \mathbf{b} - \mathbf{Ax}_k$, to zero in a finite number of iterations. The problem of solving (1) by iterative methods can be represented by the signal regulation problem:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k \\ \mathbf{y}_k = \mathbf{Ax}_k \\ \mathbf{r}_k = \mathbf{b} - \mathbf{y}_k \\ \mathbf{u}_k = f(\mathbf{r}_k), \end{cases} \quad (3)$$

i.e., designing control signal \mathbf{u}_k in order to zero the steady state error by choice of the control law $f(\cdot)$ i.e., $\lim_{k \rightarrow \infty} \mathbf{r}_k = \mathbf{0}$, or, equivalently, $\lim_{k \rightarrow \infty} \mathbf{Ax}_k = \mathbf{b}$.

System (3) models an iterative method as a feedback control system (plant state space realization $\{\mathbf{I}, \mathbf{I}, \mathbf{A}, \mathbf{0}\}$ with controller in unitary feedback configuration). In control theory, both classical and modern, there are various techniques of solving the signal regulation problem (3) representing the solution of (1), which correspond to different choices of the controller $f(\cdot)$ in (3). However, as shown in [5], many of these techniques (pole allocation, state feedback) result in controllers that explicitly use the inverse matrix \mathbf{A}^{-1} . In [2] and [3], the authors show that, by using appropriate quadratic *control Liapunov functions* (CLFs), choosing a proportional-derivative controller for $f(\cdot)$ in (3) leads to the CG method, shown in algorithm 1. The CLF method proposed in [2] and [3] gives an idea of the robustness of the CG method, since it does not utilize the assumption $\mathbf{p}_0 = \mathbf{r}_0$ (initialization of conjugate directions), utilized to obtain residue orthogonality (see, e.g. [11]). This indicates that, in situations where the assumption does not hold, the determination of CG parameters by

CLF still ensures the decrease of some norm of the residual vector. When the assumption $\mathbf{p}_0 = \mathbf{r}_0$ holds, it can be shown that the parameters determined by CLFs are the same as those commonly found in the literature [11, p.411,eqs 3:1b,3:1e].

Algorithm 1 *Hestenes-Stiefel CG algorithm for constant Linear Systems*

Calculate $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$, $\mathbf{p}_0 = \mathbf{r}_0$

For $k = 0, 1, \dots$, until convergence

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k$$

$$\beta_k = -\frac{\langle \mathbf{r}_{k+1}, \mathbf{A}\mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle}$$

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$$

End

Nagaraja and Bose derived conjugate gradient algorithms for perceptron training in both batch and adaptive modes, but presented an analytical proof of convergence only for the batch mode algorithm [13]. They also used heuristics based on reinitialization techniques of the CG method [9] in order to ensure the convergence of the algorithms. In this paper, the CLF technique of [3], is used to derive steepest descent and conjugate gradient like algorithms, with analytical proof of convergence in the batch mode, but without the use of any heuristic technique, in contrast to [13].

3 Batch mode algorithms

Let $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_j, \mathbf{a}_{j+1}, \mathbf{a}_{j+2}, \dots, \mathbf{a}_m$ be, m patterns where each \mathbf{a}_i is an n -dimensional vector belonging to a class denominated as $+1$ for $1 \leq i \leq j$, and belonging to a class denominated as -1 for $j+1 \leq i \leq m$; such that if \mathbf{a}_i belongs to class -1 then $-\mathbf{a}_i$ belongs to class $+1$. Let \mathbf{A} denote the $m \times n+1$ bias augmented matrix of m patterns defined as

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \dots & \mathbf{a}_j & -\mathbf{a}_{j+1} & \dots & -\mathbf{a}_m \\ 1 & \dots & 1 & -1 & \dots & -1 \end{bmatrix}^T. \quad (4)$$

Let \mathbf{x} represent the $(n+1)$ -dimensional weight vector. Then, the training of a single perceptron can be formulated as: find an $(n+1)$ -vector \mathbf{x} such that

$$\mathbf{A}\mathbf{x} > \mathbf{0}. \quad (5)$$

This can be formulated [10] as the problem of finding an element (\mathbf{x}, \mathbf{b}) belonging to the set

$$\mathcal{S}^+ := \{\mathbf{x} \in R^{n+1}, \mathbf{b} \in R^m : \mathbf{A}\mathbf{x} = \mathbf{b} > \mathbf{0}\}. \quad (6)$$

The positive vector $\mathbf{b} \in \mathbb{R}^m$ is henceforth referred to as the *margin vector*. Ho and Kashyap [12] have proposed an iterative algorithm for finding an element of \mathcal{S}^+ . In this case, the components of the margin vector \mathbf{b} are first initialized to small positive values, and the Moore-Penrose inverse is used to generate an MSE minimum norm solution for \mathbf{x}_k which minimizes the objective function $J(\mathbf{x}_k, \mathbf{b}_k) = \|\mathbf{A}\mathbf{x}_k - \mathbf{b}_k\|^2$:

$$\mathbf{x}_k = \mathbf{A}^\dagger \mathbf{b}_k \quad (7)$$

where $(\cdot)^\dagger$ denotes the Moore-Penrose inverse. Next, a new estimate for the margin vector is computed by performing the constrained descent

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \frac{1}{2}(|\mathbf{r}_k| - \mathbf{r}_k), \quad (8)$$

where $\mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k$. A new estimate of \mathbf{x} , \mathbf{x}_{k+1} can now be computed using (7) and employing the updated margin vector from (8), this process is iterated until all the components of \mathbf{r}_k are zero, which is an indication of linear separability of the training set, or until $\mathbf{r}_k > \mathbf{0}$, which in this case is an indication that linear separability of the training set may not be possible [10]. New algorithms can be obtained by using steepest descent and conjugate gradient techniques to obtain the solution \mathbf{x}_k which minimizes the objective function $J(\mathbf{x}_k, \mathbf{b}_k)$ and compute a new estimate for the margin vector. The main technique consists of analyzing the perceptron training method as a dynamic system in the standard feedback configuration, and choosing an appropriate CLF. This technique has been used in [5] to analyze and design Krylov methods and in [6] to design new adaptive filtering algorithms.

Combining the steepest descent technique with the Ho-Kashyap technique [10], system (3) can be rewritten as:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{r}_k \\ \mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k) \\ \mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k \end{cases} \quad (9)$$

One of the main results of this paper is stated as follows:

Theorem 1 *Assume that*

$$\forall \mathbf{v} > \mathbf{0}, \mathbf{A}^T \mathbf{v} \neq \mathbf{0}. \quad (10)$$

Define

$$\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|. \quad (11)$$

The choice

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} \quad (12)$$

is optimal and ensures that the equilibrium of the system (9) is asymptotically stable ($\mathbf{r}_k \rightarrow 0$), i.e. the equilibrium of the system (9) tends to the set $\mathcal{S}^+ := \{\mathbf{x} \in R^{n+1}, \mathbf{b} \in R^m : \mathbf{A}\mathbf{x} = \mathbf{b} > \mathbf{0}\}$.

Proof 1 From (9), one can write

$$\begin{aligned} \mathbf{r}_{k+1} &= \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1} \\ &= \mathbf{b}_k + \alpha_k(|\mathbf{r}_k| - \mathbf{r}_k) - \mathbf{A}\mathbf{x}_k - \alpha_k\mathbf{A}\mathbf{A}^T\mathbf{r}_k \\ &= \mathbf{r}_k - \alpha_k((\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|) \\ &= \mathbf{r}_k - \alpha_k\mathbf{r}'_k, \end{aligned} \quad (13)$$

where \mathbf{r}'_k is given by (11). Choose the control Liapunov function

$$\mathbf{V}_r(k) = \langle \mathbf{r}_k, \mathbf{r}_k \rangle. \quad (14)$$

Then

$$\begin{aligned} \Delta\mathbf{V}_r &= \mathbf{V}_r(k+1) - \mathbf{V}_r(k) = \langle \mathbf{r}_{k+1}, \mathbf{r}_{k+1} \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k - \alpha_k\mathbf{r}'_k, \mathbf{r}_k - \alpha_k\mathbf{r}'_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k, \mathbf{r}_k \rangle - 2\alpha_k\langle \mathbf{r}_k, \mathbf{r}'_k \rangle + \alpha_k^2\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= -2\alpha_k\langle \mathbf{r}_k, \mathbf{r}'_k \rangle + \alpha_k^2\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle. \end{aligned} \quad (15)$$

The optimal value of α_k (which makes $\Delta\mathbf{V}_r$ as negative as possible), given by (12), is determined calculating $\frac{\partial\Delta\mathbf{V}_r}{\partial\alpha_k}$ and setting it to zero. Substituting (12) into (15) yields

$$\Delta\mathbf{V}_r = -\frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle^2}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} \leq \mathbf{0}. \quad (16)$$

From (11) it can be written that

$$\langle \mathbf{r}_k, \mathbf{r}'_k \rangle = \mathbf{r}_k^T\mathbf{A}\mathbf{A}^T\mathbf{r}_k + \mathbf{r}_k^T\mathbf{r}_k - \mathbf{r}_k^T|\mathbf{r}_k| \quad (17)$$

On the right hand side of (17), the first term is positive, unless $\mathbf{A}^T\mathbf{r}_k = \mathbf{0}$ and the sum of the second and third terms is also always positive, unless $\mathbf{r}_k > \mathbf{0}$. From this it can be concluded that

$$\langle \mathbf{r}_k, \mathbf{r}'_k \rangle \geq 0 \quad (18)$$

and is equal to zero if and only if $\mathbf{r}_k > \mathbf{0}$ and $\mathbf{A}^T \mathbf{r}_k = \mathbf{0}$. Therefore, from (10)

$$\Delta \mathbf{V}_r = -\frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle^2}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} < \mathbf{0}, \quad (19)$$

which implies that $\|\mathbf{r}_k\|$ is a decreasing sequence.

In order to prove the positivity of \mathbf{b}_k , notice that $|\mathbf{r}_k| - \mathbf{r}_k \geq \mathbf{0}$ and from (12) and (18) $\alpha_k \geq 0$. Therefore if $\mathbf{b}_0 > \mathbf{0}$, then from (9) $\mathbf{b}_k > \mathbf{0}$ for all k .

The pseudocode of the extension of the SD method for batch mode perceptron training, based on theorem 1 and referenced in this paper as SD-CLF, is presented in algorithm 2.

Algorithm 2 *Batch mode SD-CLF for Perceptron Training*

Choose $\mathbf{x}_0, \mathbf{b}_0 > \mathbf{0}$

Calculate $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{A}\mathbf{x}_0$

For $k = 0, 1, \dots$, until convergence

$$\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|$$

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{r}_k$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k)$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{r}'_k$$

End

Another algorithm can be obtained by combining the conjugate gradient technique with the Ho-Kashyap technique [10], in order to take advantage of the well known faster convergence of CG type algorithms for equalities. In this case system (3) can be rewritten as:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_{\mathbf{x}_k} \\ \mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k} \\ \mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k \\ \mathbf{p}_{\mathbf{x}_k} = \beta_k \mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{A}^T \mathbf{r}_k \\ \mathbf{p}_{\mathbf{b}_k} = \beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + |\mathbf{r}_k| - \mathbf{r}_k \end{cases} \quad (20)$$

The other main result of this paper is stated as follows:

Theorem 2 *With the assumption (10), define*

$$\mathbf{p}_k = \mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}. \quad (21)$$

The choices

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \quad (22)$$

$$\beta_{k+1} = -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \quad (23)$$

ensure that $\mathbf{r}_k \rightarrow 0$ and $\mathbf{p}_k \rightarrow 0$, i.e. the equilibrium of the system (20) tends to the set $\mathcal{S} := \{\mathbf{x} \in R^n, \mathbf{b} \in R^m : \mathbf{A}\mathbf{x} = \mathbf{b}\}$.

Proof 2 The parameters α_k and β_k are considered as control variables and then can be determined by an appropriate choice of CLF. Notice that

$$\mathbf{r}_{k+1} = \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{r}_k - \alpha_k(\mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}). \quad (24)$$

Define

$$\begin{aligned} \mathbf{p}_k &= \mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k} \\ &= \beta_k \mathbf{A}\mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{A}\mathbf{A}^T \mathbf{r}_k - \beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + \mathbf{r}_k - |\mathbf{r}_k| \\ &= \beta_k (\mathbf{A}\mathbf{p}_{\mathbf{x}_{k-1}} - \mathbf{p}_{\mathbf{b}_{k-1}}) + (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k| \\ &= \beta_k \mathbf{p}_{k-1} + \mathbf{r}'_k, \end{aligned}$$

where

$$\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|.$$

Then the signal regulation problem can be rewritten in $\mathbf{r}_k, \mathbf{p}_k$ coordinates as follows

$$\begin{cases} \mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{p}_k \\ \mathbf{p}_{k+1} = \beta_{k+1} \mathbf{p}_k + \mathbf{r}'_{k+1} \end{cases} \quad (25)$$

The objective is now to determine the parameters α_k and β_k which make the equilibrium $(\mathbf{0}, \mathbf{0})$ of the system (25) asymptotically stable. At the equilibrium of (25) one can write $\mathbf{A}\mathbf{x}_k = \mathbf{b}_k$ and $\mathbf{A}\mathbf{p}_{\mathbf{x}_k} = \mathbf{p}_{\mathbf{b}_k}$, then $\mathbf{A}\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \alpha_k \mathbf{A}\mathbf{p}_{\mathbf{x}_k} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k} = \mathbf{b}_{k+1}$. This means that, at the equilibrium of (25), the equilibrium of (20) belongs to the set \mathcal{S} . To determine α_k , once again choose the CLF $V_{\mathbf{r}}(k)$ given by (14), then

$$\begin{aligned} \Delta V_{\mathbf{r}} &= V_{\mathbf{r}}(k+1) - V_{\mathbf{r}}(k) \\ &= \langle \mathbf{r}_{k+1}, \mathbf{r}_{k+1} \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k - \alpha_k \mathbf{p}_k, \mathbf{r}_k - \alpha_k \mathbf{p}_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k, \mathbf{r}_k \rangle - 2\alpha_k \langle \mathbf{r}_k, \mathbf{p}_k \rangle + \alpha_k^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= -2\alpha_k \langle \mathbf{r}_k, \mathbf{p}_k \rangle + \alpha_k^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle. \end{aligned} \quad (26)$$

The value of α_k , given by (22), is determined calculating $\frac{\partial \Delta V_{\mathbf{r}}}{\partial \alpha_k}$ and setting it

to zero. Since $\mathbf{p}_k^T \mathbf{r}_k \neq \mathbf{0}$, substituting (22) into (26) yields

$$\Delta \mathbf{V}_r = -\frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} < \mathbf{0}. \quad (27)$$

Notice from (25), (22), (10), (17) and (18)

$$\begin{aligned} \langle \mathbf{r}_k, \mathbf{p}_k \rangle &= \beta_k \langle \mathbf{r}_k, \mathbf{p}_{k-1} \rangle + \langle \mathbf{r}_k, \mathbf{r}'_k \rangle = \langle \mathbf{r}_k, \mathbf{r}'_k \rangle \\ &= \mathbf{r}_k^T \mathbf{A} \mathbf{A}^T \mathbf{r}_k + \mathbf{r}_k^T \mathbf{r}_k - \mathbf{r}_k^T |\mathbf{r}_k| > \mathbf{0}. \end{aligned} \quad (28)$$

From (27), it can be concluded that the 2-norm of the residue \mathbf{r}_k decreases monotonically.

The next step is to determine the parameter β_k , and for this purpose define

$$\begin{aligned} \mathbf{V}_{\mathbf{p}_{k+1}} &= \langle \mathbf{p}_{k+1}, \mathbf{p}_{k+1} \rangle \\ &= \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle + 2\beta_{k+1} \langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle + \beta_{k+1}^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle. \end{aligned} \quad (29)$$

Then, one can write

$$\frac{\partial \Delta \mathbf{V}_{\mathbf{p}_{k+1}}}{\partial \beta_{k+1}} = \mathbf{0} \implies \beta_{k+1} = -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \quad (30)$$

Substituting (30) in (29) leads

$$\mathbf{V}_{\mathbf{p}_{k+1}} = \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle - \frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \leq \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle \quad (31)$$

From (27) and (11) it can be concluded that \mathbf{r}'_{k+1} decreases in 2-norm, but not necessarily monotonically. Thus (31) implies that \mathbf{p}_{k+1} decreases in 2-norm, although not necessarily monotonically, concluding the proof.

The above derivations can be summarized in the form of the following algorithm

Algorithm 3 Batch mode CG-CLF for Perceptron Training

Choose $\mathbf{x}_0, \mathbf{b}_0 > \mathbf{0}$

Calculate $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{A}\mathbf{x}_0, \mathbf{p}_{\mathbf{x}_0} = \mathbf{A}^T \mathbf{r}_0$

Calculate $\mathbf{p}_{\mathbf{b}_0} = |\mathbf{r}_0| - \mathbf{r}_0, \mathbf{p}_0 = \mathbf{A}\mathbf{p}_{\mathbf{x}_0} - \mathbf{p}_{\mathbf{b}_0}$

For $k = 0, 1, \dots$, until convergence with $\mathbf{b}_{final} > \mathbf{0}$

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_{\mathbf{x}_k}$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k}$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{p}_k$$

$$\mathbf{r}'_{k+1} = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_{k+1} - |\mathbf{r}_{k+1}|$$

$$\begin{aligned}\beta_{k+1} &= -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \\ \mathbf{p}_{\mathbf{x}_{k+1}} &= \beta_{k+1} \mathbf{p}_{\mathbf{x}_k} + \mathbf{A}^T \mathbf{r}_{k+1} \\ \mathbf{p}_{\mathbf{b}_{k+1}} &= \beta_{k+1} \mathbf{p}_{\mathbf{b}_k} + |\mathbf{r}_{k+1}| - \mathbf{r}_{k+1} \\ \mathbf{p}_{k+1} &= \beta_{k+1} \mathbf{p}_k + \mathbf{r}'_{k+1}\end{aligned}$$

End

Notice that algorithm 2 has 4 steps less than algorithm 3 and correspondingly a lower computational cost, but slower convergence in general.

Observation 1 Notice that theorem 2 guarantees that algorithm 3 will converge to a solution of $\mathbf{A}\mathbf{x}_k = \mathbf{b}_k$, but does not ensure that $\mathbf{b}_k > \mathbf{0}$. However, the experimental results below show that algorithm 3 converges to a solution \mathbf{x}_k such that $\mathbf{A}\mathbf{x}_k = \mathbf{b}_k > \mathbf{0}$, and a rigorous proof that convergence to \mathcal{S}^+ , instead of just to \mathcal{S} , occurs is being sought.

4 Numerical simulations

In order to verify the convergence properties of the proposed batch mode algorithms, several simulations are performed. The performance of the proposed algorithms are compared with the Ho-Kashyap (HK) and its batch mode adaptive version (AHK) [10] and the conjugate gradient algorithm of [13] (called CGA). The experiment was performed for various linearly and non-linearly separable sets. The set chosen is the iris plants database from the UCI ML repository [15]. This is a frequently used benchmark dataset in the pattern recognition literature and contains three classes of 50 instances each, where each class refers to a type of iris plant. Each data point consists of four numeric, predictive attributes and the class. The attributes are sepal length, sepal width, petal length and petal width, all measured in cm. The three class labels, *iris setosa*, *iris versicolor* and *iris virginica* correspond to the three types of iris plant. It is known that *iris setosa* is linearly separable from the other two classes, but the latter are not linearly separable from each other even with respect to all four attributes. Though this a three-class problem, it can be converted to a two-class problem thus allowing the use of perceptron learning [13]. One way to do this is to convert it into 3 two class problems by making samples belonging to class w_i ($1 \leq w_i \leq 3$) positive and those not belonging to it, negative. In the first case, samples belonging to *iris setosa* are marked as class +1 and those belonging to *iris versicolor* are assigned a class label -1. In the second case, samples belonging to *iris setosa* are again taken as belonging to the class +1 and those belonging to *iris virginica* are assigned a class label -1. In the third and last case, the *iris versicolor* samples are labeled as class +1 while the *iris virginica* samples are class -1.

Figures 1, 2 and 3 plot the separating hyperplanes obtained by using the

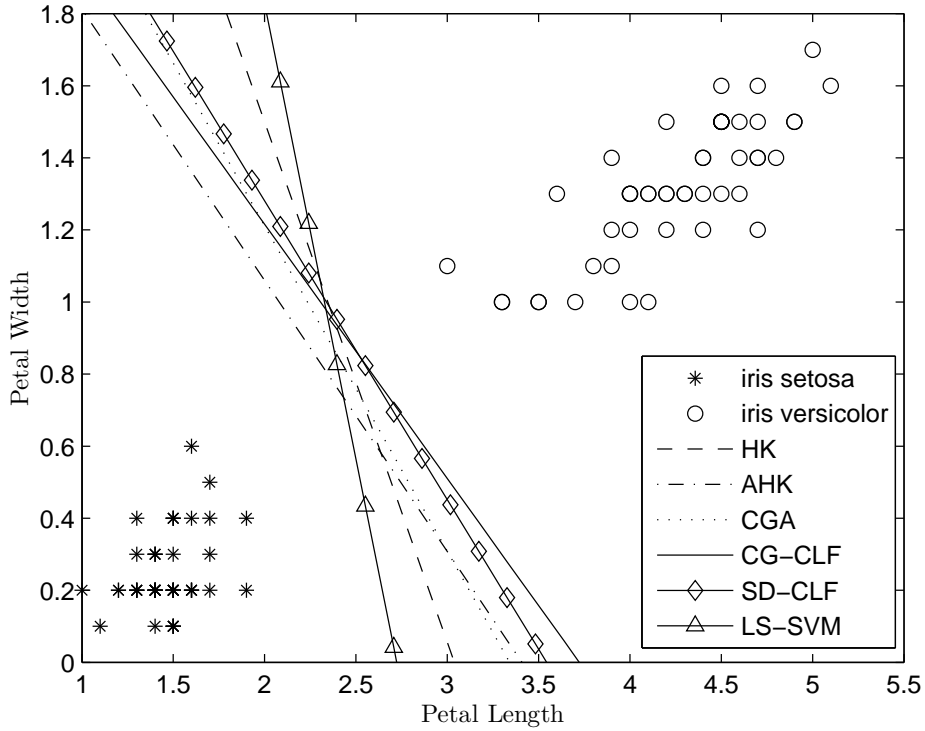


Fig. 1. Separating hyperplanes of classes *iris setosa* and *iris versicolor*. All the considered algorithms classify all the samples correctly. The distance margin M obtained after convergence is shown in table 1. The stopping criterion is $\|\mathbf{r}_k\| \leq 0.001$ and the maximum number of iterations is 1000.

algorithms 3, 2, HK, AHK and CGA. As the perceptron training problem can be solved by support vector machine (SVM), the obtained results are compared with the results of the least-square SVM (LS-SVM) of [1,4,8,18,19]. It can be noticed that, for the two linearly separable cases, all the considered algorithms classify correctly all the samples, but only the algorithms 3 and HK reach the stopping criterion set as $\|\mathbf{r}_k\| \leq 0.001$. The HK and AHK algorithms are very sensitive to the choice of the parameters ρ_1 and ρ_2 , i.e., improper choice of the parameter values ρ_1 and ρ_2 would lead to poor performance. Then the choices $\rho_1 = 1, 5$ and $\rho_2 = 1/\lambda_{max}$ (λ_{max} is the highest eigenvalue of $\mathbf{A}^T \mathbf{A}$), as in [13], result in a residual vector with relatively high norm, as shown in table 1 where

$$\%CC = \frac{\text{Number of samples correctly classified}}{\text{Number of samples}} \times 100, \quad (32)$$

and M is the margin distance defined as the minimum of the distances from the correctly classified samples to the separating hyperplane, the stopping criterion is $\|\mathbf{r}_k\| \leq 0.001$ and the maximum number of iterations is equal to 1000.

The results in table 1 show that algorithm 3 returns a residual vector with

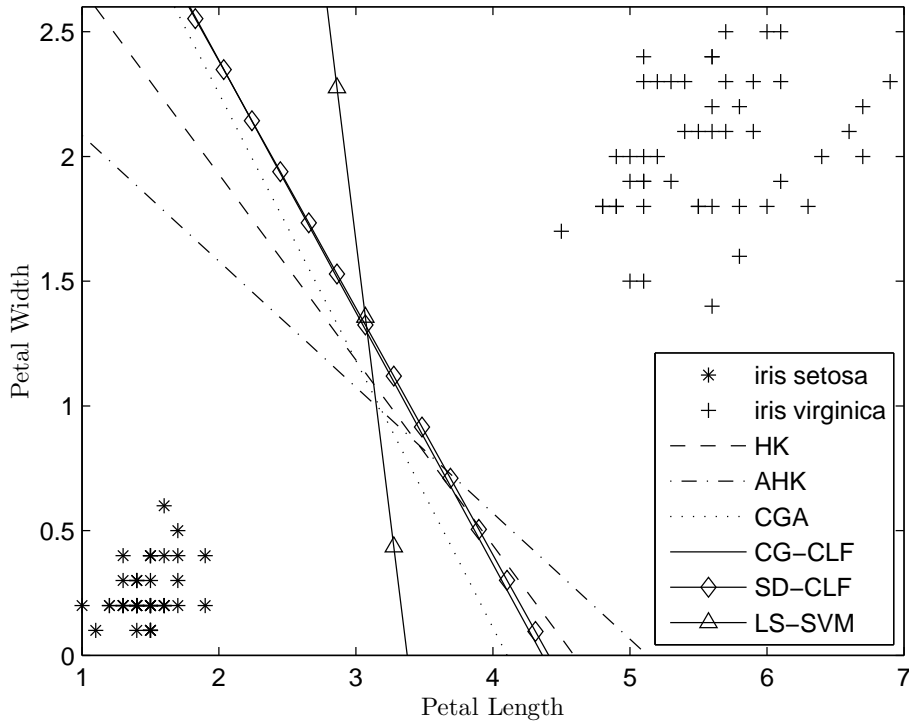


Fig. 2. Separating hyperplanes of classes *iris setosa* and *iris virginica*. All the considered algorithms classify all the samples correctly. The distance margin M obtained after convergence is shown in table 1. The stopping criterion is $\|\mathbf{r}_k\| \leq 0.001$ and the maximum number of iterations is 1000.

the lowest norm, in linearly separable cases. The proposed algorithms present basically the same performance in the linearly non-separable case, except the AHK algorithm which fails, classifying all samples as belonging to the same class.

Notice that the computational cost of one iteration of algorithm 2 is lower than one of algorithm 3, but if the number of iterations of algorithm 2 is too high, its total computational cost and time can be higher than the total computational cost of algorithm 3. To illustrate this, we run an experiment in order to calculate the time necessary for the convergence of each algorithm. The experiment consists in running the same algorithm 100 times with the same initial conditions and determining the mean time t (in seconds) and the number of iterations. The stopping criterion is again set as $\|\mathbf{r}_k\| \leq 0.001$. The results of this experiment are shown in table 2. As expected, although algorithm 2 is simpler than algorithm 3, in some cases algorithm 3 converges more rapidly and therefore spends less mean time than the algorithms 2, AHK and CGA.

Other numerical tests were performed using the sets “Ionosphere”, “BUPA

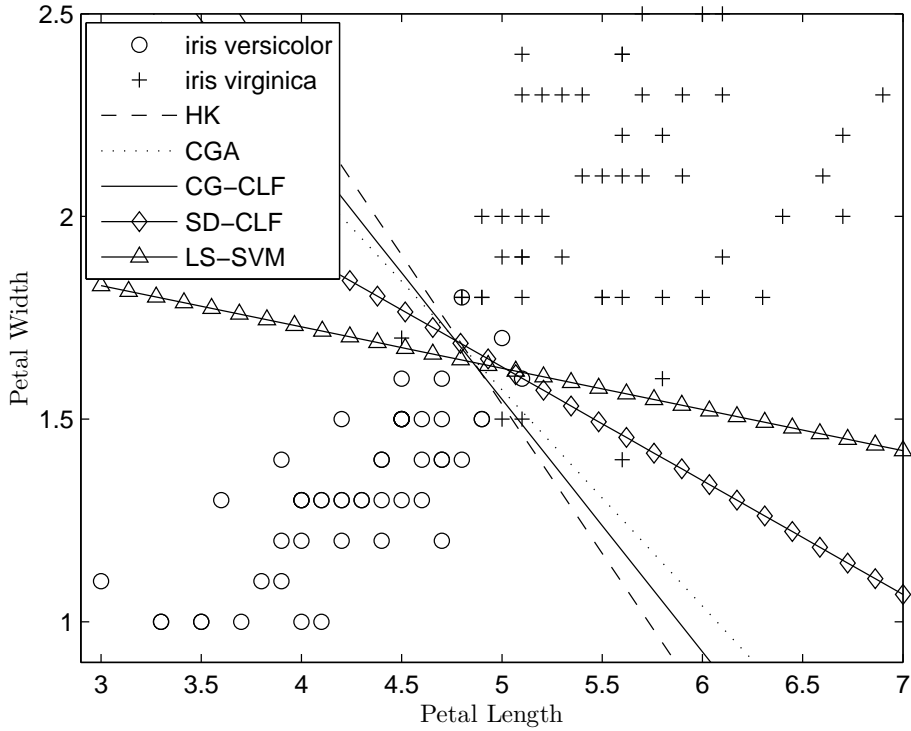


Fig. 3. Separating hyperplanes of classes *iris versicolor* and *iris virginica*. Table 1 shows the classification statistics and the distance margin M obtained after reaching the maximum number of iterations. The stopping criterion is $\|\mathbf{r}_k\| \leq 0.001$ and the maximum number of iterations is 1000.

Liver Disorders” and “Pima Indian Diabetes”, confirming Observation 1. The classification results are shown in table 3. More details on these three sets can be found in [15].

We also perform a numerical test with the classical toy X-OR problem. In this case we have two classes such that $y = 0$ for the class 1 composed by two vectors $((0, 0)$ and $(1, 1))$ and $y = 1$ for the class 2 composed by two vectors $((0, 1)$ and $(1, 0))$. The classification results are shown in figure 4, it can be observed that the proposed algorithms, the CGA and the AHK algorithms classify correctly the maximum number of patterns while the HK algorithm fails to separate the two classes.

The numerical results obtained show that the proposed batch mode conjugate gradient algorithm presents equivalent or better performance than the algorithms HK, AHK and CGN without using any empirical learning parameter and any heuristic reinitialization of the direction vector. It is observed that for the linearly nonseparable cases (i.e. $\mathcal{S}^+ = \emptyset$), the proposed algorithms converge to a margin vector $\mathbf{b}_k > \mathbf{0}$ and a solution \mathbf{x}_k that minimizes $\|\mathbf{b}_k - \mathbf{A}\mathbf{x}_k\|$, which corresponds to the correct classification of the maximum number of patterns.

Table 1
Comparative performance of the methods.

<i>Iris</i>	Algorithm	%CC	M	$\ \mathbf{r}_k\ $
<i>Setosa</i> × <i>Versicolor</i>	HK	100	1.3050	9.96×10^{-4}
	AHK	100	1.3118	3.41×10^{-1}
	CGA	100	1.3105	7.30×10^{-3}
	CG-CLF	100	1.2989	8.01×10^{-4}
	SD-CLF	100	1.2893	9.51×10^{-1}
	LS-SVM	100	1.3133	×
<i>Setosa</i> × <i>Virginica</i>	HK	100	1.2561	9.98×10^{-4}
	AHK	100	1.2625	1.74×10^{-1}
	CGA	100	1.2554	2.70×10^{-3}
	CG-CLF	100	1.2504	8.42×10^{-4}
	SD-CLF	100	1.2463	6.96×10^{-1}
	LS-SVM	100	1.2664	×
<i>Versicolor</i> × <i>Virginica</i>	HK	95	3.3292	2.01
	AHK	50	0.8636	2.75×10^{77}
	CGA	94	3.3297	3.59
	CG-CLF	95	3.3295	2.08
	SD-CLF	94	3.3318	3.52
	LS-SVM	95	2.9931	×

5 Conclusions

We have presented conjugate gradient and steepest descent algorithms for perceptron training in batch mode and proved the asymptotic stability of the algorithms in theorems 1 and 2. Their performance have been compared with the algorithms CGA [13], HK, AHK [10] and LS-SVM [1, 4, 8, 18, 19]. The HK algorithm uses the pseudo-inverse of the pattern matrix in order to determine the solution vector. However the determination of the pseudo-inverse becomes unstable as the matrix $\mathbf{A}^T \mathbf{A}$ becomes close to singular, on the other hand a great increase of the computational cost of this process can be observed for large scale problems, i.e., separation of classes with large number of attributes. The AHK algorithm does not calculate the pseudo-inverse of the pattern matrix, but the empirical choice of the learning parameters ρ_1 and ρ_2 is critical for the convergence of the algorithm. Numerical results in [13] and this paper show that improper choice of the values of the learning parameters ρ_1 and ρ_2 lead to poor performance of the AHK algorithm. Nagaraja and Bose [13] have

Table 2
Comparison of mean time and iteration number.

<i>Iris</i>	Algorithm	t [s]	iter
<i>Setosa</i> × <i>Versicolor</i>	HK	0.0234	462
	AHK	0.0562	1000
	CGA	0.1094	1000
	CG-CLF	0.0327	79
	SD-CLF	0.3656	1000
<i>Setosa</i> × <i>Virginica</i>	HK	0.0172	295
	AHK	0.0563	1000
	CGA	0.1109	1000
	CG-CLF	0.0421	107
	SD-CLF	0.3609	1000
<i>Versicolor</i> × <i>Virginica</i>	HK	0.0407	1000
	AHK	0.0579	1000
	CGA	0.1125	1000
	CG-CLF	0.3547	1000
	SD-CLF	0.3548	1000

presented a conjugate gradient algorithm for the perceptron training, however they use an heuristic technique based on conjugate gradient reinitialization [9] in order to ensure the convergence of the algorithm. This is due to the fact that the standard conjugate algorithm uses orthogonality of the residues to simplify the formulas of the parameters necessary for convergence [11]. In an finite precision implementation the numerical errors could cause a loss of the orthogonality of the residues, leading to a non-convergence of the method. To solve this problem, this paper does not use any heuristic technique, in contrast to [13]. The new idea proposed is to use an analysis based on the linear structure of system (3) and the bilinear structure of the dynamical system solvers (9) and (20) considered. The asymptotic stability of the algorithms is proved in theorems 1 and 2. The proof of the positivity of the margin vector \mathbf{b}_k found by algorithm 3 is a topic under investigation. The numerical results obtained show that the proposed batch mode algorithms presents equivalent or better performance than the algorithms HK, AHK and CGN without using any empirical learning parameter and any heuristic reinitialization of the direction vector.

Table 3

Comparative performance of the methods. The HK method fails to calculate the pseudo-inverse of the “Ionosphere” data matrix, and the AHK method diverges for the “BUPA” and “Diabetes” sets.

<i>Set</i>	Algorithm	% <i>CC</i>	$\ \mathbf{r}_k\ $
Ionosphere	HK	×	×
	AHK	89.7	8.07
	CGA	91.3	7.83
	CG-CLF	91.3	7.89
	SD-CLF	88.5	9.05
BUPA	HK	66.2	15.54
	AHK	×	×
	CGA	66.9	15.61
	CG-CLF	66.9	15.54
	SD-CLF	67.2	15.73
Diabetes	HK	73.7	19.05
	AHK	×	×
	CGA	74.3	19.79
	CG-CLF	73.9	19.02
	SD-CLF	66.4	21.85

Acknowledgment

The authors would like to thank the anonymous reviewers for helpful comments.

References

- [1] D. Anguita, S. Ridella, S. Rovetta, Circuitual implementation of support vector machines, *Electronic Letters* 34 (1998) 1596–1597.
- [2] A. Bhaya, E. Kaszkurewicz, Iterative methods as dynamical systems with feedback control, in: *Proc. 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, USA, 2003.
- [3] A. Bhaya, E. Kaszkurewicz, *Control Perspectives on Numerical Algorithms and Matrix Problems*, SIAM, PA, 2006.
- [4] C. Cortes, V. Vapnik, Support-vector networks, *Machine Learning* 20 (3) (1995) 273–297.

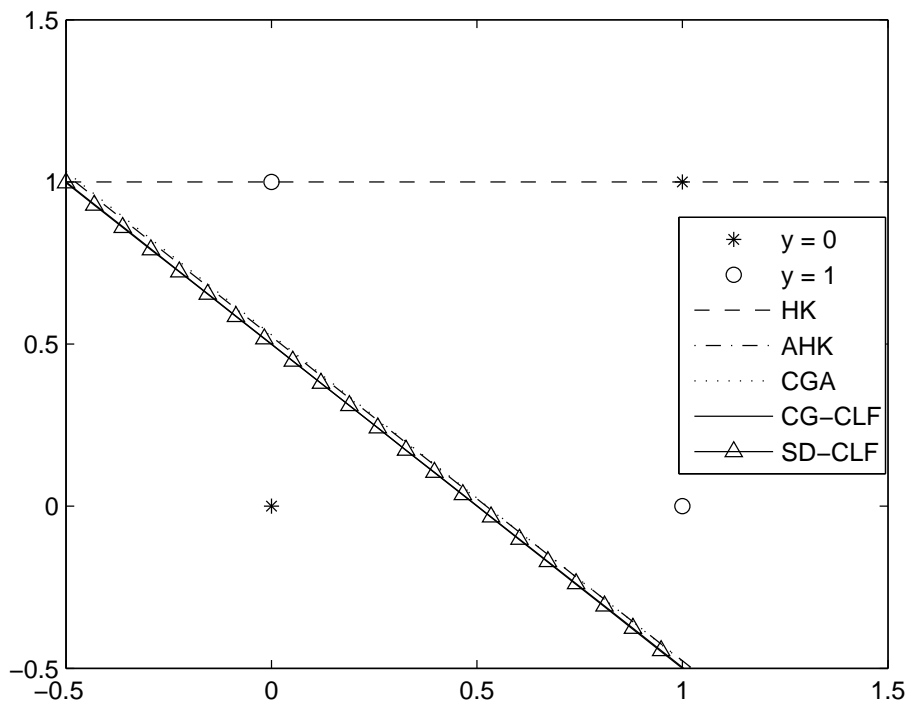


Fig. 4. Separating hyperplanes of an X-OR problem. The proposed algorithms, the CGA and the AHK algorithms classify correctly the maximum number of patterns while the HK algorithm fails to separate the two classes.

- [5] O. Diene, Linear iterative methods: Design and analysis using control theory tools, Master's thesis, COPPE/UFRJ, in Portuguese (April 2004).
- [6] O. Diene, A. Bhaya, Adaptive filtering algorithms designed using control liapunov functions, *IEEE Signal Processing Letters* 13 (4) (2006) 224–227, correction in vol. 14 (12) (2007) 1047.
- [7] R. O. Duda, P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley and Sons, New York, 1973.
- [8] L. V. Ferreira, E. Kaszkurewicz, A. Bhaya, Support vector classifiers via gradient systems with discontinuous righthand sides, *Neural Networks* 19 (10) (2006) 1612–1623.
- [9] R. Fletcher, C. M. Reeves, Function minimization by conjugate gradients, *Comput. J.* 7 (1964) 149–154.
- [10] M. H. Hassoun, J. Song, Adaptive Ho-Kashyap rules for perceptron training, *IEEE Trans. on Neural Networks* 3 (1) (1992) 51–61.
- [11] M. R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, *Journal of Research of the National Bureau of Standards* 49 (1952) 409–436.

- [12] Y. C. Ho, R. L. Kashyap, An algorithm for linear inequalities and its applications, IEEE Trans. on Electron. Comput. EC-14 (1965) 683–688.
- [13] G. Nagaraja, R. P. J. C. Bose, Adaptive conjugate gradient algorithm for perceptron training, Neurocomputing 69 (2006) 368–386.
- [14] G. Nagaraja, G. Krishna, Solution of linear inequalities: a conjugate gradient algorithm for linear constraints, in: Proceeding of Joint Automatic Control Conference, vol. 19-21, 1974.
URL http://www.cse.iitb.ac.in/~gn/CGA_Constraints.pdf
- [15] D. J. Newman, S. Hettich, C. L. Blake, C. Merz, UCI repository of machine learning databases (1998).
URL <http://www.ics.uci.edu/~mllearn/MLRepository.html>
- [16] N. J. Nilsson, The mathematical foundations of learning machines, Morgan Kaufmann, San Mateo, CA, 1965.
- [17] Y. Saad, Iterative methods for sparse linear systems, PWS Publishing Company, Boston, 1996.
- [18] B. Schölkopf, A. J. Smola, R. C. Williamson, P. L. Bartlett, New support vector algorithms, Neural Computation 12 (2000) 1207–1245.
- [19] J. A. Suykens, T. V. Gestel, J. D. Brabanter, B. De Moor, J. Vandewalle, Least squares support vector machines, World Scientific Publishing Co.

Perceptron Training Algorithms designed using Discrete-Time Control Liapunov Functions

Oumar Diene and Amit Bhaya

Abstract—Perceptrons, proposed in the seminal paper McCulloch–Pitts of 1943, have remained of interest to neural network community because of their simplicity and usefulness in classifying linearly separable data. Gradient descent and conjugate gradient are two widely used techniques for solving a set of linear inequalities. In finite precision implementation, the numerical errors could cause a loss of the residue orthogonality, which, in turn, results in loss of convergence. This paper takes a recently proposed control-inspired approach, to the design of iterative perceptron training algorithms, by regarding certain training/algorithm parameters as controls and then using a control Liapunov technique to choose appropriate values of these parameters.

I. INTRODUCTION

PERCEPTRONS, proposed in the seminal paper McCulloch–Pitts of 1943, have remained of interest to neural network community because of their simplicity and usefulness in classifying linearly separable data. The recent focus on support vector machines, which can be viewed as an outgrowth of perceptrons, has also rekindled interest in the basic properties of perceptrons and their so-called training algorithms. Training a perceptron involves (recursively) solving a set of linear inequalities and several algorithms have been proposed in the literature.

The perceptron training rule is an error-correcting rule used to generate the weights and threshold for a linear threshold gate (perceptron) such that a linear decision surface is synthesized that discriminates between two linearly separable class samples clustered in feature space. The main advantage of the perceptron training rule is its low computational requirements and its ability to guarantee convergence to a solution for linearly separable problems [1]. The main drawback of this rule is the fact that it does not converge to useful approximate solutions for linearly nonseparable problems [2].

The Ho-Kahsyap algorithm determines a mean squared solution by using the pseudo-inverse of the pattern matrix [3]. The main drawback of this algorithm is need to calculate the pseudo-inverse which could become unstable as the matrix $\mathbf{A}^T \mathbf{A}$ (\mathbf{A} is the pattern matrix) becomes close to singular, and is computationally expensive for large scale problems. The adaptive versions of the Ho-Kashyap algorithm do not use the pseudo-inverse but require empirical learning parameters [4].

This research was partially financed by Project Nos 141566/2004-3, 551863/2002-1 of CNPq and also by the agencies CAPES and FAPERJ.

The authors are with the Department of Electrical Engineering at the Federal University of Rio de Janeiro, PEE/COPPE/UFRJ, PO Box 68504, Rio de Janeiro, RJ 21945-970, BRAZIL. oumar@ufrj.br, amit@nacad.ufrj.br

The Conjugate Gradient (CG) is considered to be one of the best iterative methods for linear systems of equations with symmetric positive definite (spd) coefficient matrices [5]. The standard CG method uses orthogonality of the residues to simplify the formulas for the parameters necessary for convergence [6]. Gradient descent and conjugate gradient are two widely used techniques for solving a set of linear inequalities. In finite precision implementation, the numerical errors could cause a loss of the residue orthogonality, which, in turn, results in loss of convergence [7]. A conjugate gradient algorithm for perceptron training is proposed in [8], [9], where the authors used heuristic techniques based on reinitialization of the CG method [10]. This paper takes a control-inspired approach, recently proposed in [11], [12], to the design of iterative perceptron training algorithms, by regarding certain training/algorithm parameters as controls and then using a control Liapunov technique to choose appropriate values of these parameters.

II. PRELIMINARIES

Consider the linear system

$$\mathbf{Ax} = \mathbf{b}. \quad (1)$$

Iterative methods for solving the system (1) can be described by the following equation

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k \quad (2)$$

where \mathbf{x}_k is an approximation of the solution and \mathbf{u}_k is a correction calculated in order to increase the accuracy of the approximated solution \mathbf{x}_k at each iteration [5], and then drive the residue, defined as $\mathbf{r}_k = \mathbf{b} - \mathbf{Ax}_k$, to zero in a finite number of iterations. The problem of solving (1) by iterative methods can be represented by the signal regulation problem:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k \\ \mathbf{y}_k = \mathbf{Ax}_k \\ \mathbf{r}_k = \mathbf{b} - \mathbf{y}_k \\ \mathbf{u}_k = f(\mathbf{r}_k), \end{cases} \quad (3)$$

i.e., designing control signal \mathbf{u}_k in order to zero the steady state error by choice of the control law $f(\cdot)$ i.e., $\lim_{k \rightarrow \infty} \mathbf{r}_k = \mathbf{0}$, or, equivalently, $\lim_{k \rightarrow \infty} \mathbf{Ax}_k = \mathbf{b}$.

System (3), represented in block diagram form in figure 1, models an iterative method as a feedback control system (plant $\{\mathbf{I}, \mathbf{I}, \mathbf{A}, \mathbf{0}\}$ with controller in unitary feedback configuration). In control theory, both classical and modern, there are various techniques of solving the signal regulation

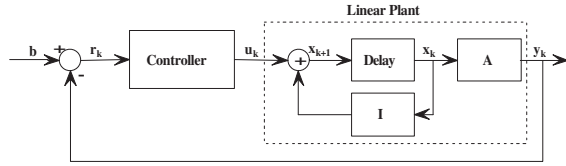


Fig. 1. Block diagram representing the regulation problem equivalent to resolution (1) of by iterative methods. The problem is to design a controller such that $\lim_{k \rightarrow \infty} \mathbf{r}_k = \mathbf{0} \Leftrightarrow \lim_{k \rightarrow \infty} \mathbf{A}\mathbf{x}_k = \mathbf{b}$.

problem (3) representing the solution of (1), which correspond to different choices of the controller $f(\cdot)$ in (3). However, as shown in [13], many of these techniques (pole allocation, state feedback) result in controllers that explicitly use the inverse matrix \mathbf{A}^{-1} . In [11] and [12], the authors show that, by using appropriate quadratic CLFs, choosing a proportional-derivative controller for $f(\cdot)$ in (3) leads to the CG method, shown in algorithm 1. The CLF method proposed in [11] and [12] gives an idea of the robustness of the CG method, since it does not use the assumption $\mathbf{p}_0 = \mathbf{r}_0$ (initialization of conjugate directions), used to obtain residue orthogonality (see, e.g. [6]). This indicates that, in situations where the assumption does not hold, the determination of CG parameters by CLF still ensures the decrease of some norm of the residual vector. When the assumption $\mathbf{p}_0 = \mathbf{r}_0$ holds, it can be shown that the parameters determined by CLFs are the same as those commonly found in the literature [6, p.411,eqs 3:1b,3:1e].

Algorithm 1: Hestenes-Stiefel CG algorithm for constant Linear Systems

Calculate $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, \mathbf{p}_0 = \mathbf{r}_0$

For $k = 0, 1, \dots$, until convergence

$$\begin{aligned} \alpha_k &= \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle} \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \mathbf{p}_k \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k \\ \beta_k &= -\frac{\langle \mathbf{r}_{k+1}, \mathbf{A}\mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle} \\ \mathbf{p}_{k+1} &= \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k \end{aligned}$$

End

Nagaraja and Bose derived conjugate gradient algorithms for perceptron training in both batch and adaptive modes, but presented an analytical proof of convergence only for the batch mode algorithm [9]. They also used heuristics based on reinitialization techniques of the CG method [10] in order to ensure the convergence of the algorithms. In this paper, the CLF technique of [12], is used to derive conjugate gradient and steepest descent like algorithms, with analytical proof of convergence in the batch mode, but without the use of any heuristic technique, in contrast to [9].

III. BATCH MODE ALGORITHMS

Let $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m$ be, m patterns where each \mathbf{a}_i is an n -dimensional vector. Let \mathbf{A} denote the $m \times n$ matrix of m

patterns defined as

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1^T \\ \vdots \\ \mathbf{a}_m^T \end{bmatrix}. \quad (4)$$

Let \mathbf{x} represent the n -dimensional weight vector. Then, the training of a single perceptron can be formulated as: find an n -vector \mathbf{x} such that

$$\mathbf{A}\mathbf{x} > \mathbf{0}. \quad (5)$$

This can be formulated [4] as the problem of finding an element (\mathbf{x}, \mathbf{b}) belonging to the set

$$\mathcal{S}^+ := \{\mathbf{x} \in \mathbb{R}^n, \mathbf{b} \in \mathbb{R}^m : \mathbf{A}\mathbf{x} = \mathbf{b} > \mathbf{0}\}. \quad (6)$$

The positive vector $\mathbf{b} \in \mathbb{R}^m$ is henceforth referred to as the *margin vector*. Ho and Kashyap [3] have proposed an iterative algorithm for finding an element of \mathcal{S}^+ . In this case, the components of the margin vector \mathbf{b} are first initialized to small positive values, and the Moore-Penrose inverse is used to generate an MSE minimum norm solution for \mathbf{x}_k which minimizes the objective function $J(\mathbf{x}_k, \mathbf{b}_k) = \|\mathbf{A}\mathbf{x}_k - \mathbf{b}_k\|^2$:

$$\mathbf{x}_k = \mathbf{A}^\dagger \mathbf{b}_k \quad (7)$$

where $(\cdot)^\dagger$ denotes the Moore-Penrose inverse. Next, a new estimate for the margin vector is computed by performing the constrained descent

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \frac{1}{2}(|\mathbf{r}_k| - \mathbf{r}_k), \quad (8)$$

where $\mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k$. A new estimate of \mathbf{x} , \mathbf{x}_{k+1} can now be computed using (7) and employing the updated margin vector from (8), this process is iterated until all the components of \mathbf{r}_k are zero, which is an indication of linear separability of the training set, or until $\mathbf{r}_k > \mathbf{0}$, which in this case is an indication of nonlinear separability of the training set [4]. A new algorithm can be obtained by using conjugate gradient techniques to obtain the solution \mathbf{x}_k which minimizes the objective function $J(\mathbf{x}_k, \mathbf{b}_k)$ and compute a new estimate for the margin vector. A recently proposed control Liapunov function (CLF) analysis of the CG method viewed as a dynamic system in the standard feedback configuration [12] is applied to the case of perceptron training. The main technique consists of an appropriate choice of a *control Liapunov function* (CLF), and has been used in [12] to analyze and design Krylov methods and in [7] to design new adaptive filtering algorithms.

Combining the conjugate gradient technique with the Ho-Kashyap technique [4], system (3) can be rewritten as:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_{\mathbf{x}_k} \\ \mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k} \\ \mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k \\ \mathbf{p}_{\mathbf{x}_k} = \beta_k \mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{A}^T \mathbf{r}_k \\ \mathbf{p}_{\mathbf{b}_k} = \beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + |\mathbf{r}_k| - \mathbf{r}_k \end{cases} \quad (9)$$

Figure 2 shows the block diagram representing system (9), and the main result of this paper is stated as follows:

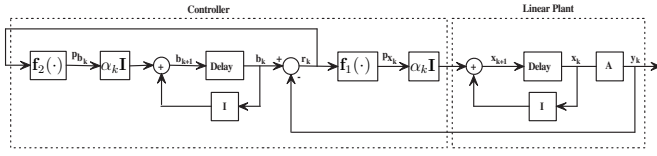


Fig. 2. Block diagram representing the regulation problem (9) equivalent to perceptron training by iterative methods, where $\mathbf{f}_1(\mathbf{r}_k) = \beta_k \mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{A}^T \mathbf{r}_k$ and $\mathbf{f}_2(\mathbf{r}_k) = \beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + |\mathbf{r}_k| - \mathbf{r}_k$.

Theorem 1: Assume that

$$\text{There is no vector } \mathbf{v} > \mathbf{0} \text{ such that } \mathbf{A}^T \mathbf{v} = \mathbf{0}. \quad (10)$$

Define

$$\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k| \quad (11)$$

$$\mathbf{p}_k = \mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}. \quad (12)$$

The choices

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \quad (13)$$

$$\beta_{k+1} = -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \quad (14)$$

ensure that $\mathbf{r}_k \rightarrow 0$ and $\mathbf{p}_k \rightarrow 0$, i.e. the equilibrium of the system (9) tends to the set $\mathcal{S} := \{\mathbf{x} \in R^n, \mathbf{b} \in R^m : \mathbf{A}\mathbf{x} = \mathbf{b}\}$.

Proof: The parameters α_k and β_k are considered as control variables and then can be determined by an appropriate choice of CLF. Notice that

$$\mathbf{r}_{k+1} = \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{r}_k - \alpha_k(\mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k}). \quad (15)$$

Define

$$\begin{aligned} \mathbf{p}_k &= \mathbf{A}\mathbf{p}_{\mathbf{x}_k} - \mathbf{p}_{\mathbf{b}_k} \\ &= \beta_k \mathbf{A}\mathbf{p}_{\mathbf{x}_{k-1}} + \mathbf{A}\mathbf{A}^T \mathbf{r}_k - \beta_k \mathbf{p}_{\mathbf{b}_{k-1}} + \mathbf{r}_k - |\mathbf{r}_k| \\ &= \beta_k (\mathbf{A}\mathbf{p}_{\mathbf{x}_{k-1}} - \mathbf{p}_{\mathbf{b}_{k-1}}) + (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k| \\ &= \beta_k \mathbf{p}_{k-1} + \mathbf{r}'_k, \end{aligned}$$

where

$$\mathbf{r}'_k = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|.$$

Then the signal regulation problem can be rewritten in \mathbf{r}_k , \mathbf{p}_k coordinates as follows

$$\begin{cases} \mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{p}_k \\ \mathbf{p}_{k+1} = \beta_{k+1} \mathbf{p}_k + \mathbf{r}'_{k+1} \end{cases} \quad (16)$$

The objective is now to determine the parameters α_k and β_k which make the equilibrium $(\mathbf{0}, \mathbf{0})$ of the system (16) asymptotically stable. At the equilibrium of (16) one can write $\mathbf{A}\mathbf{x}_k = \mathbf{b}_k$ and $\mathbf{A}\mathbf{p}_{\mathbf{x}_k} = \mathbf{p}_{\mathbf{b}_k}$, then $\mathbf{A}\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \alpha_k \mathbf{A}\mathbf{p}_{\mathbf{x}_k} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k} = \mathbf{b}_{k+1}$. This means that, at the equilibrium of (16), the equilibrium of (9) belong to the set \mathcal{S} . To determine α_k , choose the CLF

$$\mathbf{V}_{\mathbf{r}}(k) = \langle \mathbf{r}_k, \mathbf{r}_k \rangle. \quad (17)$$

Then,

$$\begin{aligned} \Delta \mathbf{V}_{\mathbf{r}} &= \mathbf{V}_{\mathbf{r}}(k+1) - \mathbf{V}_{\mathbf{r}}(k) \\ &= \langle \mathbf{r}_{k+1}, \mathbf{r}_{k+1} \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k - \alpha_k \mathbf{p}_k, \mathbf{r}_k - \alpha_k \mathbf{p}_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k, \mathbf{r}_k \rangle - 2\alpha_k \langle \mathbf{r}_k, \mathbf{p}_k \rangle + \alpha_k^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= -2\alpha_k \langle \mathbf{r}_k, \mathbf{p}_k \rangle + \alpha_k^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle. \end{aligned} \quad (18)$$

The value of α_k , given by (13), is determined calculating $\frac{\partial \Delta \mathbf{V}_{\mathbf{r}}}{\partial \alpha_k}$ and setting it to zero. Since $\mathbf{p}_k^T \mathbf{r}_k \neq 0$, substituting (13) into (18) yields

$$\Delta \mathbf{V}_{\mathbf{r}} = -\frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} < 0. \quad (19)$$

From (16), (13), (11), and (10), that

$$\begin{aligned} \langle \mathbf{r}_k, \mathbf{p}_k \rangle &= \beta_k \langle \mathbf{r}_k, \mathbf{p}_{k-1} \rangle + \langle \mathbf{r}_k, \mathbf{r}'_k \rangle = \langle \mathbf{r}_k, \mathbf{r}'_k \rangle \\ &= \mathbf{r}_k^T \mathbf{A}\mathbf{A}^T \mathbf{r}_k + \mathbf{r}_k^T (\mathbf{r}_k - |\mathbf{r}_k|) > 0. \end{aligned} \quad (20)$$

From (19), it can be concluded that the 2-norm of the residue \mathbf{r}_k decreases monotonically.

The next step is to determine the parameter β_k , and for this purpose define

$$\begin{aligned} \mathbf{V}_{\mathbf{p}_{k+1}} &= \langle \mathbf{p}_{k+1}, \mathbf{p}_{k+1} \rangle \\ &= \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle + 2\beta_{k+1} \langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle + \beta_{k+1}^2 \langle \mathbf{p}_k, \mathbf{p}_k \rangle. \end{aligned} \quad (21)$$

Then, one can write

$$\frac{\partial \Delta \mathbf{V}_{\mathbf{p}_{k+1}}}{\partial \beta_{k+1}} = \mathbf{0} \implies \beta_{k+1} = -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \quad (22)$$

Substituting (22) in (21) leads

$$\mathbf{V}_{\mathbf{p}_{k+1}} = \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle - \frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle^2}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle} \leq \langle \mathbf{r}'_{k+1}, \mathbf{r}'_{k+1} \rangle \quad (23)$$

From (19) and (11) it can be concluded that \mathbf{r}'_{k+1} decreases in 2-norm, but not necessarily monotonically. Thus (23) implies that \mathbf{p}_{k+1} decreases in 2-norm, although not necessarily monotonically, concluding the proof. ■

The above derivations can be summarized in the form of the following algorithm

Algorithm 2: Batch mode CG-CLF for Perceptron Training

Choose $\mathbf{x}_0, \mathbf{b}_0 > \mathbf{0}$

Calculate $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{A}\mathbf{x}_0, \mathbf{p}_{\mathbf{x}_0} = \mathbf{A}^T \mathbf{r}_0$

Calculate $\mathbf{p}_{\mathbf{b}_0} = |\mathbf{r}_0| - \mathbf{r}_0, \mathbf{p}_0 = \mathbf{A}\mathbf{p}_{\mathbf{x}_0} - \mathbf{p}_{\mathbf{b}_0}$

For $k = 0, 1, \dots$, until convergence with $\mathbf{b}_{final} > \mathbf{0}$

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_{\mathbf{x}_k}$$

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k \mathbf{p}_{\mathbf{b}_k}$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{p}_k$$

$$\mathbf{r}'_{k+1} = (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_{k+1} - |\mathbf{r}_{k+1}|$$

$$\beta_{k+1} = -\frac{\langle \mathbf{r}'_{k+1}, \mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{p}_k \rangle}$$

$$\mathbf{p}_{\mathbf{x}_{k+1}} = \beta_{k+1} \mathbf{p}_{\mathbf{x}_k} + \mathbf{A}^T \mathbf{r}_{k+1}$$

$$\mathbf{p}_{\mathbf{b}_{k+1}} = \beta_{k+1} \mathbf{p}_{\mathbf{b}_k} + |\mathbf{r}_{k+1}| - \mathbf{r}_{k+1}$$

$$\mathbf{p}_{k+1} = \beta_{k+1} \mathbf{p}_k + \mathbf{r}'_{k+1}$$

End

Observation 1: Notice that theorem 1 guarantees that algorithm 2 will converge to a solution of $\mathbf{A}\mathbf{x}_k = \mathbf{b}_k$, but

does not ensure that $\mathbf{b}_k > \mathbf{0}$. However, the experimental results below show that algorithm 2 converges to a solution \mathbf{x}_k such that $\mathbf{A}\mathbf{x}_k = \mathbf{b}_k > \mathbf{0}$, and a rigorous proof that convergence to \mathcal{S}^+ , instead of just to \mathcal{S} , occurs is being sought.

In [7], a steepest descent (SD) method for adaptive filters was proposed. The results in [7] show that it is possible for the SD method for adaptive filters to have a performance comparable with the CG and RLS methods. The same extension, made above for the CG method, can be carried out for the SD method in order to design, by CLFs, a SD method for batch mode perceptron training (SD-CLF). The system of equations representing the regulation problem, equivalent to the batch mode perceptron training problem by the SD-CLF method, is given by (24) below

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{r}_k \\ \mathbf{b}_{k+1} = \mathbf{b}_k + \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k) \\ \mathbf{r}_k = \mathbf{b}_k - \mathbf{A}\mathbf{x}_k \end{cases} \quad (24)$$

A convergence result can now be stated as follows:

Theorem 2: With the assumption (10), the choice

$$\alpha_k = \frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} \quad (25)$$

is optimal and ensures that the equilibrium of the system (24) is asymptotically stable (i.e. $\mathbf{r}_k \rightarrow \mathbf{0}$), where \mathbf{r}'_k is given by (11).

Proof: From (24), one can write

$$\begin{aligned} \mathbf{r}_{k+1} &= \mathbf{b}_{k+1} - \mathbf{A}\mathbf{x}_{k+1} \\ &= \mathbf{b}_k + \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k) - \mathbf{A}\mathbf{x}_k - \alpha_k \mathbf{A}\mathbf{A}^T \mathbf{r}_k \\ &= \mathbf{r}_k - \alpha_k ((\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k|) \\ &= \mathbf{r}_k - \alpha_k \mathbf{r}'_k, \end{aligned} \quad (26)$$

where \mathbf{r}'_k is given by (11). Once again choose the control Liapunov function $V_{\mathbf{r}}(k)$ given by (17), then

$$\begin{aligned} \Delta V_{\mathbf{r}} &= V_{\mathbf{r}}(k+1) - V_{\mathbf{r}}(k) \\ &= \langle \mathbf{r}_{k+1}, \mathbf{r}_{k+1} \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k - \alpha_k \mathbf{r}'_k, \mathbf{r}_k - \alpha_k \mathbf{r}'_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= \langle \mathbf{r}_k, \mathbf{r}_k \rangle - 2\alpha_k \langle \mathbf{r}_k, \mathbf{r}'_k \rangle + \alpha_k^2 \langle \mathbf{r}'_k, \mathbf{r}'_k \rangle - \langle \mathbf{r}_k, \mathbf{r}_k \rangle \\ &= -2\alpha_k \langle \mathbf{r}_k, \mathbf{r}'_k \rangle + \alpha_k^2 \langle \mathbf{r}'_k, \mathbf{r}'_k \rangle. \end{aligned} \quad (27)$$

The optimal value of α_k (which makes $\Delta V_{\mathbf{r}}$ as negative as possible), given by (25), is determined calculating $\frac{\partial \Delta V_{\mathbf{r}}}{\partial \alpha_k}$ and setting it to zero. Substituting (25) into (27) yields

$$\Delta V_{\mathbf{r}} = -\frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle^2}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} < \mathbf{0}, \quad (28)$$

which implies that $\|\mathbf{r}_k\|$ is a decreasing sequence. ■

The pseudocode of the extension of the SD method for batch mode perceptron training, based on theorem 2 and referenced in this paper as SD-CLF, is presented in algorithm 3.

Algorithm 3: Batch mode SD-CLF for Perceptron Training

Choose $\mathbf{x}_0, \mathbf{b}_0 > \mathbf{0}$

Calculate $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{A}\mathbf{x}_0$

For $k = 0, 1, \dots$, until convergence

$$\begin{aligned} \mathbf{r}'_k &= (\mathbf{A}\mathbf{A}^T + \mathbf{I})\mathbf{r}_k - |\mathbf{r}_k| \\ \alpha_k &= \frac{\langle \mathbf{r}_k, \mathbf{r}'_k \rangle}{\langle \mathbf{r}'_k, \mathbf{r}'_k \rangle} \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \mathbf{A}^T \mathbf{r}_k \\ \mathbf{b}_{k+1} &= \mathbf{b}_k + \alpha_k (|\mathbf{r}_k| - \mathbf{r}_k) \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k \mathbf{r}'_k \end{aligned}$$

End

Notice that algorithm 3 has 4 steps less than algorithm 2 and correspondingly a lower computational cost, but slower convergence in general. Observation 1 does not apply, since $\mathbf{b}_0 > \mathbf{0}$ and $\alpha_k > 0$.

IV. NUMERICAL SIMULATIONS

In order to verify the convergence properties of the proposed batch mode algorithms, several simulations are performed. The performance of the proposed algorithms are compared with the Ho-Kashyap (HK) and its batch mode adaptive version (AHK) [4] and the conjugate gradient algorithm of [9] (called CGA). The experiment was performed for various linearly and non-linearly separable sets. The set chosen is the iris plants database from the UCI ML repository [14]. This is a frequently used benchmark dataset in the pattern recognition literature and contains three classes of 50 instances each, where each class refers to a type of iris plant. Each data point consists of four numeric, predictive attributes and the class. The attributes are sepal length, sepal width, petal length and petal width, all measured in cm. The three class labels, *iris setosa*, *iris versicolor* and *iris virginica* correspond to the three types of iris plant. It is known that *iris setosa* is linearly separable from the other two classes, but the latter are not linearly separable from each other even with respect to all four attributes. Though this a three-class problem, it can be converted to a two-class problem thus allowing the use of perceptron learning [9]. One way to do this is to convert it into 3 two class problems by making samples belonging to class w_i ($1 \leq w_i \leq 3$) positive and those not belonging to it, negative. In the first case, samples belonging to *iris setosa* are marked as class +1 and those belonging to *iris versicolor* are assigned a class label -1. In the second case, samples belonging to *iris setosa* are again taken as belonging to the class +1 and those belonging to *iris virginica* are assigned a class label -1. In the third and last case, the *iris versicolor* samples are labeled as class +1 while the *iris virginica* samples are class -1.

For computational purposes (but not for the graphs in the figures below), the input vectors are augmented by adding an additional input of +1 for the bias. The negative class patterns including the bias are negated so that the goal becomes: find the weight vector \mathbf{x}_k such that $\mathbf{A}\mathbf{x}_k > \mathbf{0}$.

Figures 3, 4 and 5 plot the separating hyperplanes obtained by using the algorithms 2, 3, HK, AHK and CGA. As the perceptron training problem can be solved by support vector machine (SVM), the obtained results are compared with the results of the least-square SVM (LS-SVM) of [15]. It can be noticed that, for the two linearly separable cases, all the considered algorithms classify correctly all the samples, but

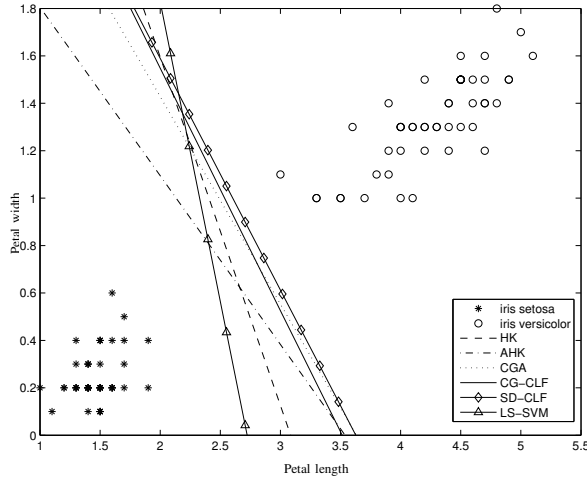


Fig. 3. Separating hyperplanes of classes *iris setosa* and *iris versicolor*. All the considered algorithms classify all the samples correctly. The distance margin M obtained after convergence is shown in table I. The stopping criterion is $\|\mathbf{r}_k\| \leq 0.001$ and the maximum number of iterations is 1000.

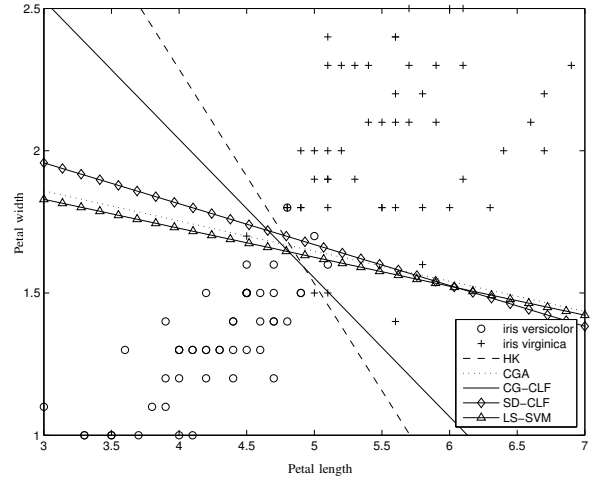


Fig. 5. Separating hyperplanes of classes *iris versicolor* and *iris virginica*. Table I shows the classification statistics and the distance margin M obtained after reaching the maximum number of iterations. The stopping criterion is $\|\mathbf{r}_k\| \leq 0.001$ and the maximum number of iterations is 1000.

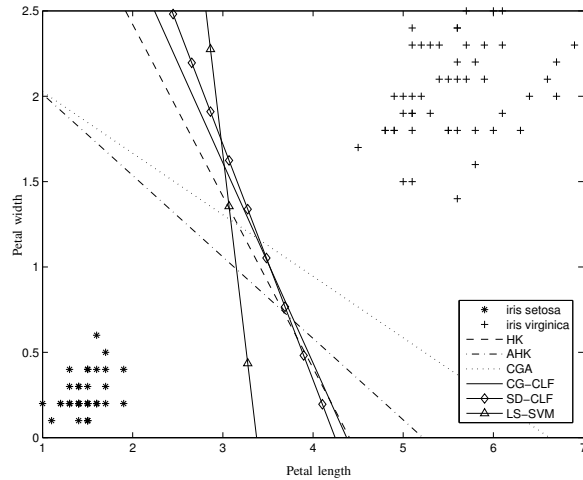


Fig. 4. Separating hyperplanes of classes *iris setosa* and *iris virginica*. All the considered algorithms classify all the samples correctly. The distance margin M obtained after convergence is shown in table I. The stopping criterion is $\|\mathbf{r}_k\| \leq 0.001$ and the maximum number of iterations is 1000.

<i>Iris</i>	Algorithm	%CC	M	$\ \mathbf{r}_k\ $
<i>Setosa</i> × <i>Versicolor</i>	HK	100	1.3050	9.96×10^{-4}
	AHK	100	1.3118	3.41×10^{-1}
	CGA	100	1.3105	7.30×10^{-3}
	CG-CLF	100	1.2989	8.01×10^{-4}
	SD-CLF	100	1.2893	9.51×10^{-1}
	LS-SVM	100	1.3133	×
<i>Setosa</i> × <i>Virginica</i>	HK	100	1.2561	9.98×10^{-4}
	AHK	100	1.2625	1.74×10^{-1}
	CGA	100	1.2554	2.70×10^{-3}
	CG-CLF	100	1.2504	8.42×10^{-4}
	SD-CLF	100	1.2463	6.96×10^{-1}
	LS-SVM	100	1.2664	×
<i>Versicolor</i> × <i>Virginica</i>	HK	95	3.3292	2.01
	AHK	50	0.8636	2.75×10^{-1}
	CGA	94	3.3297	3.59
	CG-CLF	95	3.3295	2.08
	SD-CLF	94	3.3318	3.52
	LS-SVM	95	2.9931	×

TABLE I
COMPARATIVE PERFORMANCE OF THE METHODS.

only the algorithms 2 and HK reach the stopping criterion set as $\|\mathbf{r}_k\| \leq 0.001$. The HK and AHK algorithms are very sensitive to the choice of the parameters ρ_1 and ρ_2 , i.e., improper choice of the parameter values ρ_1 and ρ_2 would lead to poor performance. Then the choices $\rho_1 = 1, 5$ and $\rho_2 = 1/\lambda_{max}$ (λ_{max} is the highest eigenvalue of $\mathbf{A}^T \mathbf{A}$), as in [9], result in a residual vector with relatively high norm, as shown in table I where

$$\%CC = \frac{\text{Number of samples correctly classified}}{\text{Number of samples}} \times 100, \tag{29}$$

and M is the margin distance defined as the minimum of the distances from the correctly classified samples to the separating hyperplane, the stopping criterion is $\|\mathbf{r}_k\| \leq 0.001$ and the maximum number of iterations is equal to 1000.

The results in table I show that algorithm 2 returns a

residual vector with the lowest norm, in linearly separable cases. The proposed algorithms present basically the same performance in the linearly non-separable case, except the AHK algorithm which fails, classifying all samples as belonging to the same class.

Notice that the computational cost of one iteration of algorithm 3 is lower than one of algorithm 2, but if the number of iterations of algorithm 3 is too high, its total computational cost and time can be higher than the total computational cost of algorithm 2. To illustrate this, we run an experiment in order to calculate the time necessary for the convergence of each algorithm. The experiment consists in running the same algorithm 100 times with the same initial conditions and determining the mean time t (in seconds) and the number of iterations. The stopping criterion is again set as $\|\mathbf{r}_k\| \leq 0.001$. The results of this experiment are shown in table II. As expected, although algorithm 3 is simpler than algorithm 2, in some cases algorithm 2 converges more rapidly and therefore spends less mean time than the

<i>Iris</i>	Algorithm	t	iter
<i>Setosa</i> × <i>Versicolor</i>	HK	0.0234	462
	AHK	0.0562	1000
	CGA	0.1094	1000
	CG-CLF	0.0327	79
	SD-CLF	0.3656	1000
<i>Setosa</i> × <i>Virginica</i>	HK	0.0172	295
	AHK	0.0563	1000
	CGA	0.1109	1000
	CG-CLF	0.0421	107
	SD-CLF	0.3609	1000
<i>Versicolor</i> × <i>Virginica</i>	HK	0.0407	1000
	AHK	0.0579	1000
	CGA	0.1125	1000
	CG-CLF	0.3547	1000
	SD-CLF	0.3548	1000

TABLE II
COMPARISON OF MEAN TIME AND ITERATION NUMBER.

<i>Set</i>	Algorithm	%CC	$\ r_k\ $
Ionosphere	HK	×	×
	AHK	89.7	8.07
	CGA	91.3	7.83
	CG-CLF	91.3	7.89
	SD-CLF	88.5	9.05
BUPA	HK	66.2	15.54
	AHK	×	×
	CGA	66.9	15.61
	CG-CLF	66.9	15.54
	SD-CLF	67.2	15.73
Diabetes	HK	73.7	19.05
	AHK	×	×
	CGA	74.3	19.79
	CG-CLF	73.9	19.02
	SD-CLF	66.4	21.85

TABLE III
COMPARATIVE PERFORMANCE OF THE METHODS. THE HK METHOD FAILS TO CALCULATE THE PSEUDO-INVERSE OF THE “IONOSPHERE” DATA MATRIX, AND THE AHK METHOD DIVERGES FOR THE “BUPA” AND “DIABETES” SETS.

algorithms 3, AHK and CGA.

Other numerical tests were performed using the sets “Ionosphere”, “BUPA Liver Disorders” and “Pima Indian Diabetes”, confirming Observation 1. The classification results are shown in table III. More details on these three sets can be found in [14].

The numerical results obtained show that the proposed batch mode conjugate gradient algorithm presents equivalent or better performance than the algorithms HK, AHK and CGN without using any empirical learning parameter and any heuristic reinitialization of the direction vector.

V. CONCLUSIONS

We have presented conjugate gradient and steepest descent algorithms for perceptron training in batch mode and proved the asymptotic stability of the algorithms in theorems 1 and 2. Their performance have been compared with the algorithms CGA [9], HK, AHK [4] and LS-SVM [15]. The HK algorithm uses the pseudo-inverse of the pattern matrix in order to determine the solution vector. However the determination of the pseudo-inverse become unstable as the matrix $A^T A$ becomes close to singular, on the other hand the computational cost of this process increases highly for large scale problems, i.e., separation of classes with large

number of attributes. The AHK algorithm does not calculate the pseudo-inverse of the pattern matrix, but the empirical choice of the learning parameters ρ_1 and ρ_2 is critical for the convergence of the algorithm. Numerical results in [9] and this paper show that improper choice of the learning parameters values ρ_1 and ρ_2 lead to poor performance of the AHK algorithm. Nagaraja and Bose [9] have presented a conjugate gradient algorithm for the perceptron training, however they use an heuristic technique based on conjugate gradient reinitialization [10] in order to ensure the convergence of the algorithm. This is due to the fact that the standard conjugate algorithm uses orthogonality of the residues to simplify the formulas of the parameters necessary for convergence [6]. In an finite precision implementation the numerical errors could cause a loss of the orthogonality of the residues, leading to a non-convergence of the method. To solve this problem, this paper does not use any heuristic technique, in contrast to [9], and proposes an analysis using the linear structure of system (3) and the bilinear structure of the dynamical system solvers (9) and (24) considered, and the asymptotic stability of the algorithms is proved in theorems 1 and 2. The proof of the positivity of the margin vector b_k found by algorithm 2 is a topic under investigation.

REFERENCES

- [1] N. J. Nilsson, *The mathematical foundations of learning machines*. San Mateo, CA: Morgan Kaufmann, 1965.
- [2] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: John Wiley and Sons, 1973.
- [3] Y. C. Ho and R. L. Kashyap, “An algorithm for linear inequalities and its applications,” *IEEE Trans. on Electron. Comput.*, vol. EC-14, pp. 683–688, 1965.
- [4] M. H. Hassoun and J. Song, “Adaptive ho-kashyap rules for perceptron training,” *IEEE Trans. on Neural Networks*, vol. 3, no. 1, pp. 51–61, 1992.
- [5] Y. Saad, *Iterative methods for sparse linear systems*. Boston: PWS Publishing Company, 1996.
- [6] M. R. Hestenes and E. Stiefel, “Methods of conjugate gradients for solving linear systems,” *Journal of Research of the National Bureau of Standards*, vol. 49, pp. 409–436, 1952.
- [7] O. Diene and A. Bhaya, “Adaptive filtering algorithms designed using control liapunov functions,” *IEEE Signal Processing Letters*, vol. 13, no. 4, pp. 224–227, 2006.
- [8] G. Nagaraja and G. Krishna, “Solution of linear inequalities: a conjugate gradient algorithm for linear constraints,” in *Proceeding of Joint Automatic Control Conference*, vol. 19-21, June 1974, pp. 662–666. [Online]. Available: http://www.cse.iitb.ac.in/~gn/CGA_Constraints.pdf
- [9] G. Nagaraja and R. P. J. C. Bose, “Adaptive conjugate gradient algorithm for perceptron training,” *Neurocomputing*, vol. 69, pp. 368–386, 2006.
- [10] R. Fletcher and C. M. Reeves, “Function minimization by conjugate gradients,” *Comput. J.*, vol. 7, pp. 149–154, 1964.
- [11] A. Bhaya and E. Kaszkurewicz, “Iterative methods as dynamical systems with feedback control,” in *Proc. 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, USA, December 2003, pp. 2374–2380.
- [12] —, *Control Perspectives on Numerical Algorithms and Matrix Problems*. PA: SIAM, 2006.
- [13] O. Diene, “Linear iterative methods: Design and analysis using control theory tools,” Master’s thesis.
- [14] D. J. Newman, S. Hettich, C. L. Blake, and C. Merz, “UCI repository of machine learning databases,” 1998. [Online]. Available: <http://www.ics.uci.edu/~mllearn/MLRepository.html>
- [15] L. V. Ferreira, E. Kaszkurewicz, and A. Bhaya, “Support vector classifiers via gradient systems with discontinuous righthand sides,” *Neural Networks*, vol. 19, no. 10, pp. 1612–1623, Dec 2006.

Referências Bibliográficas

- [1] YOUNG, D. M., *Iterative Solution of Large Linear Systems*. 111 Fifth Avenue, New York 10003, Academic Press, 1971.
- [2] VARGA, R. S., *Matrix Iterative Analysis*. Englewood Cliffs, NJ, Prentice Hall, 1962.
- [3] HESTENES, M. R., STIEFEL, E., “Methods of conjugate gradients for solving linear systems”. *Journal of Research of the National Bureau of Standards* v. 49, pp. 409–436, 1952.
- [4] KELLEY, C. T., *Iterative Methods for Linear and Nonlinear Equations*. Philadelphia, SIAM, 1995.
- [5] SAAD, Y., *Iterative methods for sparse linear systems*. Boston, PWS Publishing Company, 1996.
- [6] SAAD, Y., VAN DER VORST, H. A., “Iterative solution of linear systems in the 20th century”. *J. Computational and Applied Mathematics* v. 122, pp. 1–33, 2000.
- [7] VAN DER VORST, H. A., “Krylov subspace iteration”. *Computing in Science and Engineering* v. 2, pp. 32–37, 2000.
- [8] SCHAEERER, C., KASZKUREWICZ, E., “The shooting method for the solution of ordinary differential equations: a control-theoretical perspective”. *International Journal of Systems Science* v. 32, pp. 1047–1053, 2001.
- [9] BHAYA, A., KASZKUREWICZ, E., “Iterative methods as dynamical systems with feedback control”. In: *Proc. 42nd IEEE Conference on Decision and Control*, pp. 2374–2380, Maui, Hawaii, USA, December 2003.
- [10] BHAYA, A., KASZKUREWICZ, E., *Control Perspectives on Numerical Algorithms and Matrix Problems*. PA, SIAM, 2006.
- [11] DIENE, O., *Métodos Iterativos Lineares: Análise e Projeto Através de Ferramentas da Teoria de Controle*. Dissertação de mestrado, COPPE/UFRJ, Abril 2004.
- [12] DIENE, O., BHAYA, A., “Métodos iterativos lineares: Análise e projeto através de ferramentas da teoria de controle”. In: *Proc. 15th Congresso Brasileiro de Automática*, 2004.
- [13] FERREIRA, L. V., *Utilização de Sistemas Gradientes para Resolução de Problemas de Otimização em Computadores Paralelos*. Tese de doutorado, COPPE/UFRJ, Agosto 2006.

- [14] PAZOS, F. A., *Projeto de Algoritmos para Resolução de Problemas de Otimização Utilizando Funções de Liapunov com Controle*. Tese de doutorado, COPPE/UFRJ, Setembro 2007.
- [15] HAYKIN, S., *Adaptive Filter Theory*. Englewood Cliffs, New Jersey, Prentice Hall, 1991.
- [16] DINIZ, P. S. R., *Adaptive Filtering: Algorithms and Practical Implementation*. 101 Philip Drive, Assinippi Park, Norwell, Massachusetts 02061 USA, Kluwer Academic Publishers, 2002.
- [17] SAYED, A. H., *Fundamentals of Adaptive Filtering*. New Jersey, John Wiley, 2003.
- [18] BORAY, G. K., SRINATH, M. D., “Conjugate gradient techniques for adaptive filtering”. *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications* v. 39, pp. 1–10, 1992.
- [19] CHANG, P. S., WILLSON, A. N., “Analysis of conjugate gradient algorithms for adaptive filtering”. *IEEE Transactions on Signal Processing* v. 48, pp. 409–418, 2000.
- [20] APOLINÁRIO JR, J. A., DE CAMPOS, M. L. R., BERNAL, C. P. O., “The constrained conjugate gradient algorithm”. *IEEE Signal Processing Letters* v. 7, pp. 351–354, 2000.
- [21] DIENE, O., BHAYA, A., “Adaptive filtering algorithms designed using control Liapunov functions”. *IEEE Signal Processing Letters* v. 4, pp. 224–227, 2006. Correction in vol. 14 (12) (2007) 1047.
- [22] DIENE, O., BHAYA, A., “Linearly constrained adaptive filtering algorithms designed using control Liapunov functions”. In: *Proc. 16th Congresso Brasileiro de Autômática*, Salvador, BA, Brazil, October 2006.
- [23] GRAY, S. D., “Multipath reduction using constant modulus conjugate gradient techniques”. *IEEE Journal on Selected Areas in Communications* v. 10, n. 8, pp. 1300–1305, 1992.
- [24] UENG, F.-B., CHEN, J.-D., TSAI, S.-C., CHEN, P.-Y., “Blind adaptive DS/CDMA receivers for multipath and multiuser environments”. *Circuits and Systems I: Regular Papers, IEEE Transactions on [Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on]* v. 53, n. 2, pp. 440–453, Feb. 2006.
- [25] PAPADIAS, C. B., PAULRAJ, A. J., “A constant modulus algorithm for multiuser signal separation in presence of delay spread using antenna arrays”. *Signal Processing Letters, IEEE* v. 4, n. 6, pp. 178–181, Jun 1997.
- [26] MIGUEZ, J., CASTEDO, L., “A linearly constrained constant modulus approach to blind adaptive multiuser interference suppression”. *IEEE Communications Letters* v. 2, n. 8, , 1998.
- [27] XU, C., FENG, G., “Comments on “a linearly constrained constant modulus approach to blind adaptive multiuser interference suppression””. *Communications Letters, IEEE* v. 4, n. 9, pp. 280–282, Sep 2000.

- [28] XU, C., FENG, G., KWAK, K. S., “A modified constrained constant modulus approach to blind adaptive multiuser detection”. *Trans. on Communications, IEEE* v. 49, n. 9, pp. 1642–1648, Sep 2001.
- [29] JOHNSON, R. C., SCHNITER, P., ENDRES, T. J., BEHM, J. D., BROWN, D. R., CASAS, R. A., “Blind equalization using the constant modulus criterion: a review”. *Proceedings of the IEEE* v. 86, n. 10, pp. 1927–1950, Oct 1998.
- [30] GODARD, D. N., “Self-recovering equalization and carrier tracking in two-dimensional data communication systems”. *Communications, IEEE Transactions on [legacy, pre - 1988]* v. 28, n. 11, pp. 1867–1875, Nov 1980.
- [31] HO, Y. C., KASHYAP, R. L., “An algorithm for linear inequalities and its applications”. *IEEE Trans. on Electron. Comput.* v. EC-14, pp. 683–688, 1965.
- [32] HO, Y. C., KASHYAP, R. L., “A class of iterative procedures for linear inequalities”. *Journal of SIAM Control* v. 4, n. 1, pp. 112–115, 1966.
- [33] DUDA, R. O., HART, P. E., *Pattern Classification and Scene Analysis*. New York, John Wiley and Sons, 1973.
- [34] NAGARAJA, G., KRISHNA, G., “Solution of linear inequalities: a conjugate gradient algorithm for linear constraints”. In: *Proceeding of Joint Automatic Control Conference*, v. 19-21, pp. 662–666, University of Texas, Austin, June 1974.
- [35] HASSOUN, M. H., SONG, J., “Adaptive Ho-Kashyap rules for perceptron training”. *IEEE Trans. on Neural Networks* v. 3, n. 1, pp. 51–61, 1992.
- [36] NAGARAJA, G., BOSE, R. P. J. C., “Adaptive conjugate gradient algorithm for perceptron training”. *Neurocomputing* v. 69, pp. 368–386, 2006.
- [37] DIENE, O., BHAYA, A., “Perceptron training algorithms designed using discrete-time control Liapunov functions”. In: *IEEE Multi-conference on Systems and Control (MSC) 2007*, pp. 608–613, Singapore, October 2007.
- [38] BJÖRCK, A., ELFVING, T., STRAKOS, Z., “Stability of conjugate gradient and Lanczos methods for linear least squares problems”. *SIAM J. Matrix Anal. Appl.* v. 19, n. 3, pp. 720–736, 1998.
- [39] PAIGE, C. C., ROZLOZNÍK, M., STRAKOS, Z., “Modified Gram-Schmidt (MGS), least squares, and backward stability of MGS-GMRES”. *SIAM J. Matrix Anal. Appl.* v. 28, n. 1, pp. 264–284, 2006.
- [40] GIRAUD, L., GRATTON, S., LANGOU, J., “Convergence in backward error of relaxed GMRES”. *SIAM J. Sci. Comput.* v. 29, n. 2, pp. 710–728, 2007.
- [41] STRAKOS, Z., TICHÝ, P., “On error estimation in the conjugate gradient method and why it works in finite precision computations”. *Electronic Transactions on Numerical Analysis* v. 13, pp. 56–80, 2002.
- [42] MEURANT, G., STRAKOS, Z., “The Lanczos and conjugate gradient algorithms in finite precision arithmetic”. *Acta Numerica* v. , pp. 471–542, 2006.

- [43] BUNCH, J. R., BORNE, R. C. L., PROUDLER, I. K., “A conceptual framework for consistency, conditioning, and stability issues in signal processing”. *IEEE Transactions on Circuits and Systems* v. 49, pp. 1971–1981, September 2001.
- [44] BUNCH, J. R., BORNE, R. C. L., PROUDLER, I. K., “Applying numerical linear algebra techniques to analyzing algorithms in signal processing”. *Linear Algebra and its Applications* v. 361, pp. 133–146, 2003.
- [45] GRÜNE, L., “Input-to-state dynamical stability and its Lyapunov function characterization”. *IEEE Transactions on Automatic Control* v. 47, n. 9, pp. 1499–1504, 2002.
- [46] DOYLE, J. C., STEIN, G., “Multivariable feedback design: Concepts for a classical/modern synthesis”. *IEEE Transactions on Automatic Control* v. 26, n. 1, pp. 4–16, 1981.
- [47] BARRETT, R., BERRY, M., CHAN, T. F., DEMMEL, J., DONATO, J., DONGARRA, J., EIJKHOUT, V., POZO, R., ROMINE, C., DER VORST, H. V., *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. Philadelphia, PA, SIAM, 1994.
- [48] GREENBAUM, A., *Iterative Methods for Solving Linear Systems*. in Applied Mathematics. 3600 University City Science Center, Philadelphia, PA 19104-2688, SIAM, 1997.
- [49] HESTENES, M. R., *Conjugate Direction Methods in Optimization*. New York, Springer-Verlag, 1980.
- [50] WIDROW, B., HOFF, J. M. E., “Adaptive switching circuits”. *IRE WESCON Convention Record* v. , pp. 96–104, 1960.
- [51] WOODBURY, M., “Inverting modified matrices”. In: *Memorandum Rep. 42*, New Jersey, 1950. Princeton University.
- [52] BOTTOMLEY, G., ALEXANDER, S., “A novel approach for stabilizing recursive least squares filters”. *Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on]* v. 39, n. 8, pp. 1770–1779, Aug 1991.
- [53] DE CAMPOS, M. L. R., *Development and Analysis of Fast and Robust Newton-type Adaptation Algorithm*. Ph.d. thesis, University of Victoria, 1995.
- [54] MACCHI, O., EWEDA, E., “Convergence analysis of self-adaptive equalizers”. *Information Theory, IEEE Transactions on* v. 30, n. 2, pp. 161–176, Mar 1984.
- [55] EWEDA, E., MACCHI, O., “Convergence of an adaptive linear estimation algorithm”. *Automatic Control, IEEE Transactions on* v. 29, n. 2, pp. 119–127, Feb 1984.
- [56] DIENE, O., BHAYA, A., “Métodos iterativos lineares projetados através de ferramentas da teoria de controle e suas aplicações”. *Revista de Controle e Automação* v. 17, n. 3, pp. 265–277, 2006.

- [57] BOYD, S., VANDENBERGHE, L., *Convex Optimization*. Edinburgh, Cambridge, Cambridge University Press, 2004.
- [58] STRANG, G., *Linear Algebra and its Applications*. Orlando, Florida 32887, Harcourt Brace Jovanovich, 1988.
- [59] FROST III, O. L., “An algorithm for linearly-constrained adaptive array processing”. *Proc. IEEE* v. 60, pp. 926–935, Aug 1972.
- [60] JIM, C. W., “A comparison of two LMS constrained optimal structures”. *Proc. IEEE* v. 65, pp. 1730–1731, Dec 1977.
- [61] GRIFFITHS, L. J., JIM, C. W., “An alternative approach to linearly constrained adaptive beamforming”. *IEEE Transactions on Antennas Propagation* v. AP-30, pp. 27–34, 1982.
- [62] RESENDE, L. S., ROMANO, J. M. T., BELLANGER, M. G., “A fast least-squares algorithm for linearly constrained adaptive filtering”. *IEEE Transactions on Signal Processing* v. 44, pp. 1168–1174, May 1996.
- [63] JOHNSON, D. H., DUDGEON, D. E., *Array Signal Processing*. Englewood Cliffs, New Jersey, Prentice-Hall, 1993.
- [64] BREED, B. R., STRAUSS, J., “A short proof of the equivalence of LCMV and GSC beamforming”. *IEEE Signal Processing Letters* v. 9, pp. 168–169, June 2002.
- [65] WERNER, S., JOSÉ A. APOLINÁRIO, J., DE CAMPOS, M. L. R., “On the equivalence of RLS implementations of LCMV and GSC processors”. *IEEE Signal Processing Letters* v. 10, n. 12, pp. 356–359, Dec 2003.
- [66] TSENG, C. Y., GRIFFITHS, L. J., “A systematic procedure for implementing the blocking matrix in decomposed form”. In: *Proc. ACSSC*, pp. 808–812, Pacific Grove, CA, October 1988.
- [67] LIBERTI, J. C., RAPPAPORT, T. S., *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*. New Jersey, Prentice Hall, 1999.
- [68] REGALIA, P. A., MBOUP, M., “Properties of some blind equalization criteria in noisy multiuser environments”. *Signal Processing, IEEE Transactions on* v. 49, n. 12, pp. 3112–3122, December 2001.
- [69] MAI, J., SAYED, A. H., “A feedback approach to the steady-state performance of fractionally spaced blind adaptive equalizers”. *IEEE Transactions on Signal Processing* v. 48, n. 1, pp. 80–91, 2000.
- [70] CHEN, Y., LE-NGOC, T., CHAMPAGNE, B., XU, C., “Recursive least squares constant modulus algorithm for blind adaptive array”. *IEEE Transactions on Signal Processing* v. 52, n. 5, pp. 1452–1456, 2004.
- [71] WHITEHEAD, J. B., TAKAWIRA, F., “Performance analysis of the linearly constrained constant modulus algorithm-based multiuser detector”. *Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on]* v. 53, n. 2, pp. 643–653, Feb 2005.

- [72] TREICHLER, J. R., AGEE, B. C., “A new approach to multipath correction of constant modulus signals”. *Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing]*, *IEEE Transactions on* v. 31, n. 2, pp. 459–472, Apr 1983.
- [73] AGEE, B. C., “The least-squares CMA: A new technique for rapid correction of constant modulus signals”. *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '86*. v. 11, pp. 953–956, Apr 1986.
- [74] PICKHOLTZ, R., ELBARBARY, K., “The recursive constant modulus algorithm; a new approach for real-time array processing”. *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on* v. 1, n. 3, pp. 627–632, Nov 1993.
- [75] WANG, Z., DOWLING, E. M., “Stochastic conjugate gradient constant modulus blind equalizer for wireless communications”. In: *Proc. 1996 IEEE International Conference on Communications*.
- [76] WANG, Z., DOWLING, E. M., “Block shanno constant modulus algorithm for wireless equalizations”. In: *ICASSP '96: Proceedings of the Acoustics, Speech, and Signal Processing, 1996. on Conference Proceedings., 1996 IEEE International Conference*, pp. 2678–2681, Washington, DC, USA, 1996. IEEE Computer Society. ISBN 0-7803-3192-3.
- [77] SHIN, E. S., CHOI, S. W., SHIM, D. H., KYEONG, M. G., CHANG, K. H., PARK, Y. O., HAN, K. C., LEE, C. K., “A new blind beamforming procedure based on the conjugate gradient method for CDMA mobile communications”. *ETRI Journal* v. 20, n. 2, , 1998.
- [78] ARASARATNAM, P., ZHU, S., CONSTANTINIDES, A. G., “Stochastic conjugate gradient based multi-user constant modulus algorithm for use in multiuser DS-CDMA environment”. In: *Proc. IEEE GLOBECOM 2002*, v. 1, pp. 458–462, Taipei, Taiwan, November 2002.
- [79] PAIK, D., GOLDEN, R. M., TORLAK, M., DOWLING, E. M., “Blind adaptive CDMA processing for smart antennas using the block Shanno constant modulus algorithm”. *IEEE Transactions on Signal Processing* v. 54, n. 5, pp. 1956–1959, 2006.
- [80] DIENE, O., BHAYA, A., “Liapunov analysis and design of conjugate gradient constant modulus algorithm applied to blind adaptive array”. *IEEE Transactions on Signal Processing* v. Submitted, 2008.
- [81] CHERN, S.-J., SUN, C.-H., “Constrained adaptive constant modulus RLS algorithm for blind DS-CDMA multiuser receiver under time-varying channels”. *IEICE Trans Fundamentals* v. E90-A, n. 7, pp. 1452–1461, 2007.
- [82] LIU, T.-H., “Linearly constrained minimum variance filters for blind multiuser detection”. *Communications, IEEE Transactions on* v. 51, n. 10, pp. 1649–1652, Oct. 2003.

- [83] YU, J.-L., HUNG, C.-L., LEE, I.-T., “A two-stage partially adaptive linear receiver for CDMA MIMO systems with alamouti’s space–time block codes”. *Digit. Signal Process.* v. 17, n. 1, pp. 244–260, 2007.
- [84] CENSOR, Y., ELFVING, T., “New methods for linear inequalities”. *Linear Algebra and its Applications* v. 42, pp. 199–211, 1982.
- [85] PIERRO, A. R. D., IUSEM, A. N., “A simultaneous projections method for linear inequalities”. *Linear Algebra and its Applications* v. 64, pp. 243–253, 1985.
- [86] BRAMLEY, R., WINNICKA, B., “Solving linear inequalities in a least squares sense”. *SIAM Journal on Scientific Computing* v. 17, n. 1, pp. 275–286, 1996.
- [87] LABONTE, G., “On solving systems of linear inequalities with artificial neural networks”. *IEEE Transactions on Neural Networks* v. 8, n. 3, pp. 590–600, May 1997.
- [88] PINAR, M. ., “Newton’s method for linear inequality systems”. *European Journal of Operational Research* v. 107, n. 3, pp. 710–719, June 1998.
- [89] ROSENBLATT, F., *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Washington DC, Spartan Books, 1962.
- [90] JACOBS, R. A., “Increased rates of convergence through learning rate adaptation”. *Neural Networks* v. 1, pp. 295–307, 1988.
- [91] ROSENBLATT, F., *The perceptron-a perceiving and recognizing automaton*. Cornell Aeronautical Laboratory, report 85-460-1, Ithaca, NY, January 1957.
- [92] BLOCK, H. D., “The perceptron: a model for brain functioning.I”. *Rev. Mod. Phys.* v. 34, pp. 123–135, 1962.
- [93] PENROSE, R., “A generalized inverse for matrices”. *Proc. Cambridge Philosoph. Soc.* v. 51, pp. 406–413, 1955.
- [94] FLETCHER, R., REEVES, C. M., “Function minimization by conjugate gradients”. *Comput. J.* v. 7, pp. 149–154, 1964.
- [95] NEWMAN, D. J., HETTICH, S., BLAKE, C. L., MERZ, C. UCI repository of machine learning databases, 1998.
- [96] FERREIRA, L. V., KASZKUREWICZ, E., BHAYA, A., “Support vector classifiers via gradient systems with discontinuous righthand sides”. In: *Proc. of the International Joint Conference on Neural Networks*, v. 4, pp. 2997–3002, Budapest, Hungary, July 2004.
- [97] DATTA, B. N., *Numerical Linear Algebra and Applications*. 511 Forest Lodge Road Pacific Grove, CA 93950, USA, Brooks/Cole Publishing Company, 1995.
- [98] DEMMEL, J. W., *Applied Numerical Linear Algebra*. Philadelphia, SIAM, 1997.
- [99] TREFETHEN, L. N., BAU, D., *Numerical Linear Algebra*. Philadelphia, SIAM, 1997.

- [100] HIGHAM, N. J., *Accuracy and stability of numerical algorithms*. Philadelphia, PA, SIAM, 1996.
- [101] DESOER, C. A., VIDYASAGAR, M., *Feedback Systems: Input-Output Properties*. New York, New York, Academic Press, Inc., 1975.
- [102] DOYLE, J. C., FRANCIS, B. A., TANNENBAUM, A. R., *Feedback Control Theory*. New York, New York, Macmillan Publishing Company, 1992.
- [103] ZHOU, K., DOYLE, J. C., GLOVER, K., *Robust and Optimal Control*. Englewood Cliffs, New Jersey, Prentice Hall, 1995.
- [104] ZHOU, K., DOYLE, J. C., *Essentials of Robust Control*. Upper Saddle River, New Jersey, Prentice Hall, 1998.
- [105] DE BRITO MOREIRA, M. V., *Uma Contribuição por Meio de Pré-compensação Normalizante e Parametrização para Utilização Efetiva do Método do Lugar Característico*. Tese de doutorado, COPPE/UFRJ, Março 2006.
- [106] SHAMMA, J. S., “The necessity of the small-gain theorem for time-varying and nonlinear systems”. *Automatic Control, IEEE Transactions on* v. 36, n. 10, pp. 1138–1147, Oct 1991.
- [107] SONTAG, E. D., “On the input-to-state stability property”. *European Journal of Control* v. 1, pp. 24–36, 1995.
- [108] SONTAG, E. D., “Comments on integral variants of ISS”. *Systems and Control Letters* v. 34, pp. 93–100, 1998.
- [109] SONTAG, E. D., “The ISS philosophy as a unifying framework for stability-like behavior”. v. 2, *Nonlinear Control in the Year 2000*, pp. 443–468. London, UK, Springer-Verlag, 2000.
- [110] JIANG, Z.-P., WANG, Y., “Input-to-state stability for discrete-time nonlinear systems”. *Automatica* v. 37, n. 6, pp. 857–869, 2001.
- [111] JIANG, Z.-P., LIN, Y., WANG, Y., “Nonlinear small-gain theorems for discrete-time feedback systems and applications”. *Automatica* v. 40, pp. 2129–2136, 2004.
- [112] LAILA, D. S., ASTOLFI, A., “Input-to-state stability for discrete-time time-varying systems with applications to robust stabilization of systems in power form”. *Automatica* v. 41, pp. 1891–1903, 2005.
- [113] HUANG, S., JAMES, M. R., NESIC, D., DOWER, P. M., “Analysis of input-to-state stability for discrete time nonlinear systems via dynamic programming”. *Automatica* v. 41, pp. 2055–2065, 2005.
- [114] CHEN, Z., HUANG, J., “A simplified small gain theorem for time-varying nonlinear systems”. *IEEE Transactions on Automatic Control* v. 50, n. 11, pp. 1904–1908, Nov. 2005.
- [115] CAI, C., TEEL, A. R., “Asymptotic characterizations of input-output-to-state stability for discrete-time systems”. *Systems and Control Letters* v. 56, pp. 408–415, 2007.

- [116] CAI, C., TEEL, A. R., “Input-output-to-state stability for discrete-time systems”. *Automatica* v. 44, n. 2, pp. 326–336, 2008.
- [117] CONCUS, P., SAYLOR, P. E., “Preconditioned iterative methods for indefinite symmetric Toeplitz systems”. In: *Iterative Methods for Large Linear Systems*, pp. 65–79, 1250 Sixth Avenue, San Diego, CA 92101, 1990. Academic Press.
- [118] LIM, J. S., UN, C. K., “Block conjugate gradient algorithms for adaptive filtering”. *Signal Process.* v. 55, n. 1, pp. 65–77, 1996.
- [119] LALOS, A. S., BERBERIDIS, K., “An efficient conjugate gradient method in the frequency domain: Application to channel equalization”. *Signal Process.* v. 88, n. 1, pp. 99–116, 2008.
- [120] BUCHNER, H., AICHNER, R., KELLERMANN, W., “Trinicon: a versatile framework for multichannel blind signal processing”. *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on* v. 3, pp. iii–889–92, May 2004.
- [121] AHMAVAARA, K., HAVERINEN, H., PICHNA, R., “Interworking architecture between 3GPP and WLAN systems”. *Communications Magazine, IEEE* v. 41, n. 11, pp. 74–81, Nov. 2003.
- [122] MOHANTY, S., “A new architecture for 3G and WLAN integration and inter-system handover management”. *Wirel. Netw.* v. 12, n. 6, pp. 733–745, 2006.
- [123] MANDIC, D. P., CHAMBERS, J. A., *Recurrent Neural Networks for Prediction*. New Jersey, John Wiley, 2001.