



**UNIVERSITATEA NAȚIONALĂ
DE ȘTIINȚĂ ȘI TEHNOLOGIE
POLITEHNICA BUCUREȘTI**



**Școala Doctorală de Electronică, Telecomunicații și
Tehnologia Informației**

REZUMAT TEZĂ DE DOCTORAT

Ing. Ionuț-Dorinel FÎCIU

**ALGORITMI ADAPTIVI EFICIENȚI BAZAȚI PE OPTIMIZAREA
ÎN SENSUL CELOR MAI MICI PĂTRATE PENTRU
IDENTIFICAREA SISTEMELOR**

**IMPROVED RECURSIVE LEAST-SQUARES ADAPTIVE
ALGORITHMS FOR SYSTEM IDENTIFICATION PROBLEMS**

Prof. Dr. Ing. Constantin PALEOLOGU
POLITEHNICA București

Conducător de doctorat

BUCUREȘTI 2025

Cuprins

1	Introducere	1
1.1	Prezentarea domeniului tezei de doctorat	1
1.2	Scopul tezei de doctorat	2
1.3	Conținutul tezei de doctorat	2
2	Introducere în teoria filtrării adaptive	3
2.1	Principiul filtrării adaptive	3
2.2	Configurațiile sistemelor adaptive	3
2.2.1	Identificarea sistemelor	4
2.2.2	Modelarea inversă	4
2.2.3	Predicția	4
2.2.4	Suprimarea interferențelor	4
2.3	Principalii algoritmi adaptivi	4
2.3.1	Algoritmi bazați pe minimizarea erorii pătratice medii	4
2.3.2	Algoritmi bazați pe optimizarea în sensul celor mai mici pătrate	5
2.4	Algoritmul RLS	5
2.4.1	Relații recursive	5
2.4.2	Relații de ortogonalitate	5
2.5	Algoritmul RLS-DCD cu ponderare exponențială	5
3	Algoritmi adaptivi de tip RLS bazați pe descompunerea tensorială	6
3.1	Modelul de sistem cu forme multiliniare	6
3.2	Algoritmul NLMS-T	7
3.3	Algoritmul RLS-T	7
3.4	Algoritmul RLS-NLMS-T	7
3.5	Algoritmul RLS-DCD-T	8
3.6	Algoritmul VR-RLS-DCD-T	9
3.7	Rezultatele simulărilor	9
3.7.1	Algoritmul RLS-NLMS-T	10
3.7.2	Algoritmul RLS-DCD-T	10
3.7.3	Algoritmul VR-RLS-DCD-T	11
3.8	Concluzii și perspective	11

4	Algoritmi adaptivi de tip RLS bazați pe abordarea reutilizării datelor	12
4.1	Algoritmul DR-RLS	12
4.2	Algoritmul DR-RLS-DCD	13
4.3	Algoritmul R-DR-RLS-DCD	14
4.4	Rezultatele simulărilor	14
4.5	Concluzii și perspective	15
5	Algoritmi adaptivi de tip RLS pentru compensarea ecoului acustic stereofonic	16
5.1	Compensarea ecoului acustic stereofonic pe baza modelului WL	16
5.2	Algoritmii WL-RLS-LSM	17
5.3	Versiunile WL-RLS-LSM	17
5.3.1	Algoritmul LSM cu gradient conjugat	17
5.3.2	Algoritmul LSM cu reducere coordonată	18
5.3.3	Algoritmul LSM cu reducere dihotomică coordonată	18
5.4	Algoritmii VR-WL-RLS-LSM	18
5.5	Algoritmul WL-DR-RLS-DCD	19
5.6	Rezultatele simulărilor	19
5.6.1	Algoritmii WL-RLS-LSM și VR-WL-RLS-LSM	19
5.6.2	Algoritmul WL-DR-RLS-DCD	20
5.7	Concluzii și perspective	21
6	Concluzii	22
6.1	Rezultate obținute	22
6.2	Contribuții originale	23
6.3	Lista lucrărilor originale	23
6.4	Perspective de dezvoltare ulterioară	25
	Bibliografie	26

Capitolul 1

Introducere

Teza explorează progresele în filtrarea adaptivă a semnalelor, determinată de nevoia de analiză eficientă a datelor în sisteme tehnologice complexe. Odată cu creșterea volumului de date și necesitatea unor tehnici robuste în medii nestaționare, filtrarea adaptivă a devenit crucială. Studiul se concentrează pe îmbunătățirea algoritmilor adaptivi de tip RLS (recursive least-squares) pentru identificarea sistemelor, un proces vital pentru crearea unor modele matematice precise ale sistemelor dinamice. Aceste modele au aplicații importante în controlul automat, problemele de predicție și prelucrarea adaptivă a semnalelor, în special unde zgomotul și nestaționaritatea reprezintă provocări. Acest capitol introduce concepte cheie și stabilește principiile fundamentale care susțin cercetarea prezentată în această lucrare.

1.1 Prezentarea domeniului tezei de doctorat

Prelucrarea semnalelor este un domeniu multidisciplinar axat pe analiza, manipularea și interpretarea semnalelor care reprezintă informații despre sisteme fizice. Scopul principal este de a extrage informații semnificative, minimizând zgomotul și distorsiunile. Filtrarea, un concept cheie, modifică semnalele pentru a obține rezultate dorite, filtrele liniare, cum ar fi cele cu răspuns finit (FIR) sau infinit (IIR) la impuls, fiind studiate pe larg.

Filtrarea adaptivă permite filtrelor să ajusteze dinamic parametrii în funcție de datele în schimbare, fiind valoroasă în medii cu caracteristici variabile ale semnalelor. Algoritmul RLS este o tehnică populară de filtrare adaptivă, cunoscut pentru convergența rapidă și calculul eficient, făcându-l potrivit pentru aplicații în timp real, cum ar fi telecomunicațiile și prelucrarea audio. Totuși, provocările precum sensibilitatea la zgomot și complexitatea computațională persistă, în special în identificarea sistemelor.

Cercetările își propun să abordeze aceste limitări prin dezvoltarea unor algoritmi îmbunătățiți care să sporească robustetea împotriva zgomotului și să îmbunătățească ratele de convergență. Astfel de progrese contribuie la soluții eficiente în aplicațiile care depind de prelucrarea semnalelor.

1.2 Scopul tezei de doctorat

Această teză doctorală își propune să îmbunătățească algoritmi adaptivi RLS în prelucrarea semnalelor și identificarea sistemelor prin trei obiective cheie. În primul rând, integrează descompunerea tensorială cu algoritmi RLS pentru a gestiona structuri de date complexe și a îmbunătăți eficiența identificării sistemelor. În al doilea rând, avansează algoritmi RLS prin implementarea unei abordări de reutilizare a datelor, sporind ratele de convergență și abordând provocările computaționale. În al treilea rând, aplică algoritmi RLS la compensarea ecoului acustic stereofonic (SAEC), îmbunătățind estimarea răspunsului la impuls și gestionând corelațiile în sistemele audio. Aceste obiective urmăresc să dezvolte algoritmi RLS inovatori care să îmbunătățească performanța și să ofere noi perspective în filtrarea adaptivă și prelucrarea semnalelor.

1.3 Conținutul tezei de doctorat

Această teză doctorală este structurată în patru capitole principale, fiecare abordând aspecte critice ale algoritmilor adaptivi RLS prelucrarea semnalelor. Capitolul 2 stabilește baza prin examinarea principiilor filtrării adaptive, concentrându-se pe metrici de performanță și comparând configurații și categorii de algoritmi adaptivi. De asemenea, discută un algoritm specific și versiunea sa îmbunătățită, evidențiind beneficiile acestora.

Capitolul 3 analizează integrarea descompunerii tensoriale cu algoritmi RLS pentru identificarea sistemelor, demonstrând cum gestionează structuri de date complexe și reformulează sarcinile în probleme gestionabile, reducând astfel complexitatea computațională. De asemenea, se evaluează eficiența diversilor algoritmi adaptivi în medii dinamice.

Capitolul 4 prezintă o strategie de reutilizare a datelor (DR) pentru a rafina algoritmi RLS, având ca scop creșterea ratelor de convergență și a performanței filtrelor prin actualizări multiple pe aceleași semnale de intrare și referință. Se evaluează diferite variații ale acestei strategii în diverse scenarii.

Capitolul 5 explorează aplicația algoritmilor RLS în contextul SAEC, abordând provocările cauzate de multiple difuzoare și microfoane. Se pune accent pe un model liniar extins și se detaliază tehnici adaptive care îmbunătățesc urmărirea și performanța în medii acustice complexe.

În final, capitolul 6 rezumă descoperirile cercetării, evidențiază contribuțiile originale prin listarea lucrărilor publicate și discută direcțiile viitoare de cercetare.

Împreună, aceste capitole vizează avansarea înțelegerii și aplicării algoritmilor adaptivi RLS, oferind perspective noi în filtrarea adaptivă și prelucrarea semnalelor.

Capitolul 2

Introducere în teoria filtrării adaptive

Acest capitol analizează principiile filtrării adaptive, metricile de performanță și aplicațiile, precum compensarea ecoului și suprimarea interferențelor. Se compară două categorii principale de algoritmi adaptivi, discutând avantajele și dezavantajele lor, și se explorează algoritmul RLS, inclusiv versiunea sa îmbunătățită cu iterații DCD (Dichotomous Coordinate Descent), evidențiind caracteristicile și provocările acestora.

2.1 Principiul filtrării adaptive

Un filtru adaptiv este un filtru digital care își ajustează dinamic parametrii pentru a optimiza performanța printr-un algoritm recursiv [1], fiind util în medii cu semnale nestaționare. Parametrii sunt controlați de un algoritm adaptiv, care determină coeficienții optimi.

Sistemul are două intrări principale: semnalul de intrare $x(n)$ și răspunsul dorit $d(n)$. Semnalul de ieșire $y(n)$ aproximează $d(n)$, iar semnalul de eroare este definit astfel:

$$e(n) = d(n) - y(n) \quad (2.1)$$

Semnalul de eroare $e(n)$ este esențial pentru actualizarea coeficienților filtrului prin minimizarea unei funcții de cost, cum ar fi eroarea pătratică medie. Algoritmii adaptivi sunt selectați pe baza robusteții, complexității, dezinclinerii normate, capacității de urmărire și vitezei de convergență.

2.2 Configurațiile sistemelor adaptive

Filtrarea adaptivă este esențială în telecomunicații pentru sarcini precum egalizarea canalelor, suprimarea interferențelor și compensarea ecoului, menținând integritatea semnalului. Este, de asemenea, importantă în AI și în învățarea profundă pentru adaptarea în timp real. Diverse configurații ale sistemelor adaptive abordează provocări specifice.

2.2.1 Identificarea sistemelor

Această configurație modelează un sistem necunoscut prin ajustarea filtrului adaptiv pentru a modela ieșirea sistemului, fiind utilizată frecvent pentru compensarea ecoului.

2.2.2 Modelarea inversă

În această configurație, filtrul adaptiv aproximează funcția de transfer inversă a unui sistem necunoscut, fiind util în ingineria audio pentru corectarea distorsiunilor.

2.2.3 Predicția

Această configurație estimează valorile „viitoare” ale semnalului prin compararea ieșirii filtrului adaptiv cu eșantioanele curente, fiind aplicată în prognoza financiară.

2.2.4 Suprimarea interferențelor

Această configurație reduce zgomotul prin utilizarea unui semnal de referință corelat cu perturbația, îmbunătățind claritatea semnalului în telecomunicații.

2.3 Principalii algoritmi adaptivi

Filtrarea adaptivă a evoluat semnificativ din anii 1940, cu două familii principale de algoritmi: cei bazați pe minimizarea erorii pătratice medii (MSE) și cei bazați pe optimizarea în sensul celor mai mici pătrate (LS). Alegerea depinde de cerințele aplicației.

2.3.1 Algoritmi bazați pe minimizarea erorii pătratice medii

Acești algoritmi utilizează eroarea pătratică medie ca funcție de cost și sunt exemplificați de algoritmul LMS (least mean square), care minimizează MSE, fiind robust și eficient pentru aplicații cu resurse computaționale limitate. Relația sa de actualizare este dată de:

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu \mathbf{x}(n) e^*(n), \quad (2.10)$$

unde $e(n)$ este semnalul de eroare. Stabilitatea este menținută prin controlul dimensiunii pasului μ , cu intervalul de stabilitate:

$$0 < \mu < \frac{2}{\lambda_{\max}}, \quad (2.11)$$

unde $\lambda_{\max}$ este valoarea proprie maximă a matricei de autocorelație a semnalului de intrare. Varianta normalizată a LMS (NLMS) ajustează dimensiunea pasului în funcție de puterea de intrare, sporind astfel robustețea.

2.3.2 Algoritmi bazați pe optimizarea în sensul celor mai mici pătrate

Acești algoritmi minimizează suma ponderată a erorilor pătrate, oferind o convergență mai rapidă, dar cu o complexitate mai mare. Algoritmul RLS este un exemplu cheie, oferind o viteză de convergență superioară, fiind ideal pentru medii dinamice. Ecuația normală pentru criteriul LS este:

$$\mathbf{R}(n)\hat{\mathbf{h}}(n) = \mathbf{p}(n), \quad (2.19)$$

unde $\mathbf{R}(n)$ este matricea de autocorelație și $\mathbf{p}(n)$ este vectorul de corelație. Algoritmul RLS actualizează recursiv aceste valori pentru a calcula coeficienții optimi ai filtrului, echilibrând complexitatea și adaptabilitatea.

2.4 Algoritmul RLS

Algoritmul RLS minimizează o sumă ponderată exponențial a erorilor pătrate utilizând o funcție de cost care include un factor de uitare, λ . Acest factor afectează memoria și reacția: valorile mai mici permit o adaptare mai rapidă, în timp ce valorile mai mari oferă o memorie pe termen lung benefică în medii staționare. Actualizarea este recursivă, permițând calculul eficient al matricei de autocorelație și al vectorului de corelație, facilitând estimarea în timp real a coeficienților filtrului.

2.4.1 Relații recursive

Algoritmul RLS utilizează relații recursive pentru actualizările funcției de cost, menținând complexitatea computațională scăzută și permițând actualizări rapide ale coeficienților filtrului, inițializați cu valori specifice pentru coeficienți și matricea de covarianță inversă.

2.4.2 Relații de ortogonalitate

Algoritmul respectă condițiile de ortogonalitate, minimizând eroarea apriori pentru coeficienții optimi și confirmând eficiența abordării RLS în medii dinamice.

2.5 Algoritmul RLS-DCD cu ponderare exponențială

Algoritmul RLS-DCD se bazează pe cadrul RLS pentru a reduce complexitatea computațională, păstrând în același timp performanța, concentrându-se pe vectorul rezidual pentru actualizări eficiente ale coeficienților filtrului.

Acesta include actualizări recursive pentru matricea de autocorelație și vectorul de corelație, esențiale pentru adaptarea răspunsului la impuls estimat. Metoda RLS-DCD prezintă o stabilitate și eficiență mai mari comparativ cu abordările RLS clasice, dovedindu-se eficientă în aplicații în timp real.

Capitolul 3

Algoritmi adaptivi de tip RLS bazați pe descompunerea tensorială

Acest capitol explorează îmbunătățirea algoritmilor adaptivi RLS folosind descompunerea tensorială pentru a optimiza identificarea sistemelor. Prin gestionarea eficientă a structurilor de date complexe, aceste metode reduc complexitatea computațională și îmbunătățesc performanța în domenii precum telecomunicațiile și procesarea semnalelor.

Cercetarea demonstrează că descompunerea tensorială permite divizarea sarcinilor de identificare a sistemelor în probleme mai mici, ușor gestionabile, reducând cerințele computaționale fără a compromite performanța. Acești algoritmi sunt promițători pentru aplicații în timp real, oferind robustețe și eficiență.

Tehnicile și concluziile discutate în acest capitol au fost publicate în [2–7].

3.1 Modelul de sistem cu forme multiliniare

Pentru identificarea sistemelor, descompunerea tensorială este utilizată pentru a gestiona eficient structuri de date complexe. Aceasta se aplică într-un sistem MISO (multiple-input/single-output), unde semnalul de ieșire $y(n)$ la indicele de timp discret n este definit ca agregarea semnalelor de intrare prin tensorul $\mathcal{X}(n) \in \mathbb{R}^{L_1 \times L_2 \times \dots \times L_N}$:

$$y(n) = \sum_{l_1=1}^{L_1} \sum_{l_2=1}^{L_2} \dots \sum_{l_N=1}^{L_N} x_{l_1 l_2 \dots l_N}(n) h_{1,l_1} h_{2,l_2} \dots h_{N,l_N}. \quad (3.2)$$

Semnalul dorit $d(n)$ include zgomotul necorelat $w(n)$:

$$d(n) = \mathbf{g}^T \mathbf{x}(n) + w(n). \quad (3.12)$$

3.2 Algoritmul NLMS-T

Algoritmul NLMS-T abordează problema identificării sistemului prin filtrare adaptivă, unde semnalul de eroare este definit ca:

$$e(n) = d(n) - \widehat{\mathbf{g}}^T(n-1)\mathbf{x}(n). \quad (3.15)$$

Inițializarea coeficienților filtrului este crucială, definită ca:

$$\widehat{\mathbf{h}}_1(0) = \begin{bmatrix} 1 \\ \mathbf{0}_{L_1-1} \end{bmatrix}, \quad \widehat{\mathbf{h}}_j(0) = \frac{1}{L_j} \mathbf{1}_{L_j}, \quad j = 2, 3, \dots, N, \quad (3.20)$$

Aceștia oferă estimări inițiale rezonabile pentru răspunsurile canalelor. Fiecare $\widehat{\mathbf{h}}_i(n)$ pentru $i = 1, 2, \dots, N$ este actualizat cu un algoritm NLMS, generând estimarea globală:

$$\widehat{\mathbf{g}}(n) = \widehat{\mathbf{h}}_N(n) \otimes \dots \otimes \widehat{\mathbf{h}}_1(n). \quad (3.21)$$

Produsul Kronecker facilitează combinarea eficientă a contribuțiilor individuale ale filtrului, îmbunătățind performanța în filtrarea adaptivă cu algoritmul NLMS-T.

3.3 Algoritmul RLS-T

Algoritmii RLS converg mai repede decât cei din familia LMS, fiind mult mai complecși computațional. Abordarea tensorială îmbunătățește eficiența RLS prin adaptarea criteriului de eroare LS [1] la un context tensorial, utilizând relațiile din (3.12) și (3.15). Algoritmul RLS-T folosește funcții de cost în N forme, fiecare corespunzând unui răspuns la impuls, aplicând optimizarea multiliniară [8] prin fixarea a $N - 1$ componente și optimizarea uneia singure, rezultând în ecuații normale rezolvate iterativ.

Pașii cheie implică actualizarea filtrelor cu vectori de câștig Kalman derivați din inversa matricelor de corelație, actualizate prin lema inversării matricelor [1]. Inițializarea urmează (3.20), iar răspunsul global la impuls este determinat similar cu (3.21). Abordarea bazată pe tensori este eficientă din punct de vedere computațional, având complexitate proporțională cu $\sum_{i=1}^N O(L_i^2)$, comparativ cu $O(L^2)$ pentru algoritmul RLS convențional, unde $L = \prod_{i=1}^N L_i$. Algoritmul RLS-T oferă o convergență și urmărire superioară, în special pentru filtre mai lungi.

3.4 Algoritmul RLS-NLMS-T

Această secțiune prezintă o soluție care utilizează algoritmii NLMS-T și RLS-T (descriși în Secțiunile 3.2 și 3.3) pentru filtrele individuale introduse în [2, 3]. Integrarea filtrelor adaptive în forme multiliniare prezintă un echilibru mai bun între caracteristicile de convergență și complexitatea computațională a abordării generale.

Algoritmii bazați pe tensori extind filtrele cu forme biliniare, iar performanța depinde de cel mai lung filtru. Pentru formele multiliniare, se propune o combinație de filtre adaptive: RLS-T pentru primele M filtre și NLMS-T pentru restul. Această metodă, denumită RLS_M -NLMS $_{N-M}$ -T, valorifică convergența rapidă a RLS-T și complexitatea mai mică a NLMS-T. Simulările demonstrează îmbunătățiri semnificative ale performanței, chiar și cu configurația care utilizează RLS pentru primul filtru. În plus, încorporarea iterațiilor DCD în algoritmul RLS-T sporește eficiența.

Tabel 3.3 Algoritmul RLS_M -NLMS $_{N-M}$ -T

Pas	Acțiune
Init.	Setează $\widehat{\mathbf{h}}_i(0)$ pe baza (3.20) $\mathbf{R}_m^{-1}(0) = \delta_m^{-1} \mathbf{I}_{L_i}$, $\delta_m > 0$, $m = 1, 2, \dots, M$, $M < N$ $\lambda_m = 1 - \frac{1}{K_m L_m}$, $K_m \geq 1$, $m = 1, 2, \dots, M$ $0 < \alpha_p < 1$, $\xi_p > 0$, $p = M+1, M+2, \dots, N$
Pentru $n = 1, 2, \dots$, număr iterații:	
1	Calculează $\mathbf{x}_{\widehat{\mathbf{h}}_i}(n)$
2	$e_{\widehat{\mathbf{h}}_i}(n) = d(n) - \widehat{\mathbf{h}}_i^T(n-1)\mathbf{x}_{\widehat{\mathbf{h}}_i}(n)$
3	Pentru $m = 1, 2, \dots, M$: $\mathbf{k}_m(n) = \frac{\mathbf{R}_m^{-1}(n-1)\mathbf{x}_{\widehat{\mathbf{h}}_m}(n)}{\lambda_m + \mathbf{x}_{\widehat{\mathbf{h}}_m}^T(n)\mathbf{R}_m^{-1}(n-1)\mathbf{x}_{\widehat{\mathbf{h}}_m}(n)}$ $\widehat{\mathbf{h}}_m(n) = \widehat{\mathbf{h}}_m(n-1) + \mathbf{k}_m(n)e_{\widehat{\mathbf{h}}_m}(n)$ $\mathbf{R}_m^{-1}(n) = \frac{1}{\lambda_m} \left[\mathbf{I}_{L_m} - \mathbf{k}_m(n)\mathbf{x}_{\widehat{\mathbf{h}}_m}^T(n) \right] \mathbf{R}_m^{-1}(n-1)$
4	Pentru $p = M+1, M+2, \dots, N$: $\widehat{\mathbf{h}}_p(n) = \widehat{\mathbf{h}}_p(n-1) + \frac{\alpha_p \mathbf{x}_{\widehat{\mathbf{h}}_p}(n)e_{\widehat{\mathbf{h}}_p}(n)}{\xi_p + \mathbf{x}_{\widehat{\mathbf{h}}_p}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}_p}(n)}$
5	$\widehat{\mathbf{g}}(n) = \widehat{\mathbf{h}}_N(n) \otimes \widehat{\mathbf{h}}_{N-1}(n) \otimes \dots \otimes \widehat{\mathbf{h}}_1(n)$

3.5 Algoritmul RLS-DCD-T

Tabel 3.4 Algoritmul RLS-DCD-T pentru un canal, cu complexitățile asociate

Pas	Acțiuni	Complexitate '×' & '+'
Init.	Setează $\widehat{\mathbf{h}}_i(0) = \mathbf{0}_{L_i \times 1}$, $\mathbf{r}_i(0) = \mathbf{0}_{L_i \times 1}$ $\mathbf{R}_i^{-1}(0) = \delta_i \mathbf{I}_{L_i}$, $\delta_i > 0$ $\lambda_i = 1 - \frac{1}{K_i L_i}$, $K_i \geq 1$	
Pentru $n = 1, 2, \dots$, număr iterații:		
1	Calculează $\mathbf{x}_{\widehat{\mathbf{h}}_i}(n)$	$L + \sum_{i=1}^{N-1} \prod_{j=i}^N L_j$ & $L-1$
2	$\mathbf{R}_i^{(0)}(n) = \lambda_i \mathbf{R}_i^{(0)}(n-1) + \mathbf{x}_{\widehat{\mathbf{h}}_i}(n)\mathbf{x}_{\widehat{\mathbf{h}}_i}^{(0)T}(n)$	$2L_i$ & L_i
3	$y_{\widehat{\mathbf{h}}_i}(n) = \widehat{\mathbf{h}}_i^T(n-1)\mathbf{x}_{\widehat{\mathbf{h}}_i}(n)$	L_i & L_i-1
4	$e_{\widehat{\mathbf{h}}_i}(n) = d(n) - y_{\widehat{\mathbf{h}}_i}(n)$	0 & 1
5	$\mathbf{p}_{0,i}(n) = \lambda_i \mathbf{r}_i(n-1) + e_{\widehat{\mathbf{h}}_i}(n)\mathbf{x}_{\widehat{\mathbf{h}}_i}(n)$	$2L_i$ & L_i
6	$\mathbf{R}_i(n)\Delta\widehat{\mathbf{h}}_i(n) = \mathbf{p}_{0,i}(n) \xrightarrow{\text{DCD}} \Delta\widehat{\mathbf{h}}_i(n), \mathbf{r}_i(n)$	0 & $N_{u,i}(2L_i+1) + M_b$
7	$\widehat{\mathbf{h}}_i(n) = \widehat{\mathbf{h}}_i(n-1) + \Delta\widehat{\mathbf{h}}_i(n)$	0 & L_i

Algoritmul RLS-DCD-T combină iterațiile DCD cu metoda RLS din Secțiunea 2.5 pentru a rezolva eficient sistemele de ecuații [4, 5]. Aceasta îmbunătățește eficiența

computațională și stabilitatea, valorificând rezultatele anterioare ale fiecăruia dintre cele N filtre pentru a calcula incrementările coeficientului. Coeficienții filtrului sunt estimați folosind metode DCD, reducând efortul de actualizare a vectorilor reziduali. Pașii și complexitatea sunt detaliate în Tabelul 3.4. Iterațiile DCD scad complexitatea aritmetică, permițând implementarea hardware fără înmulțiri, făcându-l o alternativă mai eficientă la metodele RLS tradiționale pentru filtre mari și sisteme adaptive.

3.6 Algoritmul VR-RLS-DCD-T

Algoritmul VR-RLS-DCD-T [6, 7], prezentat în Tabelul 3.6, îmbunătățește RLS-DCD-T prin ajustarea dinamică a lui δ_i , pentru a optimiza performanța la raport semnal-zgomot (RSZ) scăzut și a aborda problemele de condiționare. De asemenea, detaliază pașii cheie precum calculul varianței și estimarea RSZ, oferind robustețe cu cost minim, fiind potrivit pentru implementări FPGA.

Tabel 3.6 Algoritmul VR-RLS-DCD-T

Pas	Acțiuni
Init.	Setează $\hat{\mathbf{h}}_i(0) = \mathbf{0}_{L_i \times 1}$; $\mathbf{r}_i(0) = \mathbf{0}_{L_i \times 1}$; $\mathbf{R}_i^{-1}(0) = \delta_i \mathbf{I}_{L_i}$, $\delta_i > 0$ $\lambda_i = 1 - \frac{1}{K_i L_i}$, $K_i \geq 1$; $\gamma_i = 1 - \frac{1}{\epsilon_i L_i}$, $\epsilon_i > 0$ $\hat{\sigma}_y^2(0) = 0$; $\hat{\sigma}_d^2(0) = 0$; $\hat{\sigma}_x^2(0) = 0$
For $n = 1, 2, \dots$, număr iterații:	
1	Calculează $\mathbf{x}_{\hat{\mathbf{h}}_i}(n)$
2	$\mathbf{R}_i^{(0)}(n) = \lambda_i \mathbf{R}_i^{(0)}(n-1) + \mathbf{x}_{\hat{\mathbf{h}}_i}(n) \mathbf{x}_{\hat{\mathbf{h}}_i}^{(0)T}(n)$
3	$\hat{\mathbf{y}}_{\hat{\mathbf{h}}_i}(n) = \hat{\mathbf{h}}_i^T(n-1) \mathbf{x}_{\hat{\mathbf{h}}_i}(n)$
4	$e_{\hat{\mathbf{h}}_i}(n) = d(n) - \hat{\mathbf{y}}_{\hat{\mathbf{h}}_i}(n)$
5 a)	$\hat{\sigma}_u^2(n) = \gamma_i \hat{\sigma}_u^2(n-1) + (1 - \gamma_i) u^2(n)$, for $u(n) \triangleq x(n), d(n), \hat{\mathbf{y}}(n)$
5 b)	Calculează $\alpha_i(n)$
5 c)	$\widehat{\text{SNR}}(n) = \frac{\hat{\sigma}_y^2(n)}{\hat{\sigma}_d^2(n) - \hat{\sigma}_y^2(n)}$
5 d)	$\delta_i(n) = \alpha_i(n) \frac{1 + \sqrt{1 + \widehat{\text{SNR}}(n)}}{\widehat{\text{SNR}}(n)}$
6	$[\mathbf{R}_i(n) + \delta_i(n) \mathbf{I}_{L_i}] \Delta \hat{\mathbf{h}}_i(n) = \mathbf{p}_{0,i}(n) \xrightarrow{\text{DCD}} \Delta \hat{\mathbf{h}}_i(n), \mathbf{r}_i(n)$
7	$\hat{\mathbf{h}}_i(n) = \hat{\mathbf{h}}_i(n-1) + \Delta \hat{\mathbf{h}}_i(n)$

3.7 Rezultatele simulărilor

Simulările pentru identificarea sistemului MISO utilizează intrări Gaussiene pentru a evalua convergența și performanța. Zgomotul alb Gaussian este comparat cu intrări corelate, care încetinesc convergența. Algoritmul RLS este mai puțin afectat de corelația intrărilor decât LMS [1]. Metricii de performanță, inclusiv nealinierea normalizată (NM) și nealinierea proiecției normalizate (NPM) [9], evaluează acuratețea răspunsurilor la impuls estimate.

3.7.1 Algoritmul RLS-NLMS-T

Figurile 3.1 și 3.2 arată că algoritmul $\text{RLS}_M\text{-NLMS}_{N-M}\text{-T}$ atinge convergența și urmărirea algoritmului RLS-T, dar are o nealiniere ușor mai mare, similară cu NLMS-T, mai ales pentru valori mari ale lui M . Algoritmii $\text{RLS}_1\text{-NLMS}_3\text{-T}$ și RLS-T au performanțe comparabile, indicând că actualizarea RLS este optimă pentru cel mai lung filtru.

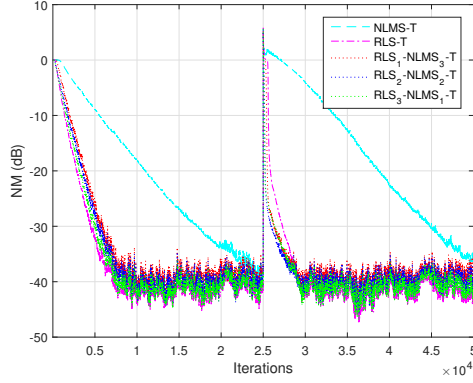


Figura 3.1 NM pentru algoritmii NLMS-T, RLS-T și $\text{RLS}_M\text{-NLMS}_{N-M}\text{-T}$ într-un scenariu de urmărire, pentru diferite valori ale lui $M < N$.

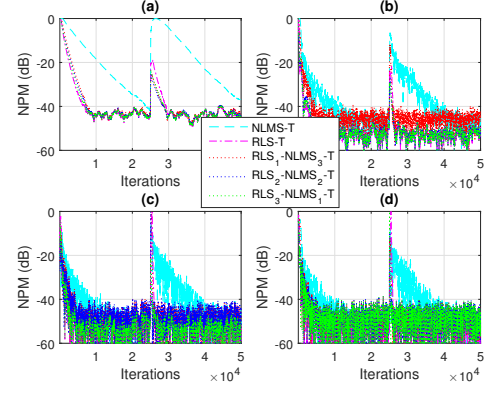


Figura 3.2 NPM pentru algoritmii NLMS-T, RLS-T și $\text{RLS}_M\text{-NLMS}_{N-M}\text{-T}$, pentru diferite valori ale lui $M < N$: (a) $\mathbf{h}_1$, (b) $\mathbf{h}_2$, (c) $\mathbf{h}_3$, (d) $\mathbf{h}_4$

3.7.2 Algoritmul RLS-DCD-T

Pentru $N = 4$ filtre și $L = 2048$ ($L_1 = 16$, $L_2 = 8$, $L_3 = L_4 = 4$), metodele bazate pe tensori depășesc RLS cu intrare AR(1) (Figura 3.8), iar reducerea memoriei RLS îmbunătățește urmărirea, dar afectează convergența. În condiții de RSZ scăzut cu $L = 4096$ și $N = 4$ ($L_i = 8$), RLS-DCD-T menține robustețea similară cu RLS-T (Figura 3.9).

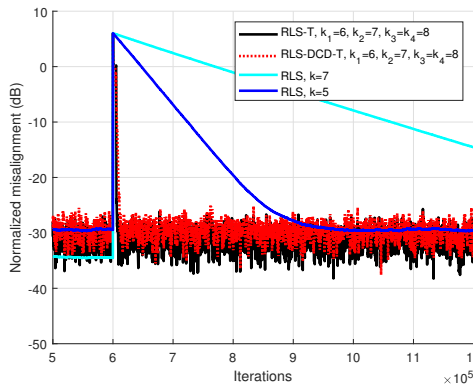


Figura 3.8 Performanța algoritmilor RLS-T, RLS-DCD-T și RLS într-un scenariu de urmărire, pentru o secvență AR(1) ca intrare, cu $N = 4$ și $L = 2048$

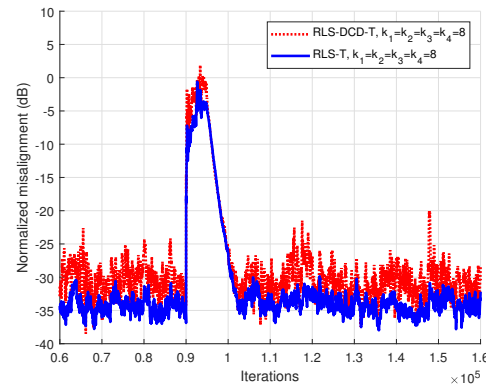


Figura 3.9 Performanța algoritmilor RLS-T și RLS-DCD-T, pentru o secvență AR(1) ca intrare, cu $N = 4$ și $L = 4096$, RSZ scăzând la -15 dB pentru 5000 de iterații

3.7.3 Algoritmul VR-RLS-DCD-T

Algoritmul VR-RLS-DCD-T, testat în condiții de RSZ scăzut, cu $N = 4$ sisteme, depășește RLS din punctul de vedere al vitezei de convergență după schimbările bruște (Figura 3.11). Acesta se aliniază la performanța RLS-DCD-T și RLS-T cu un efort computațional mai mic. Figura 3.13 arată reziliența VR-RLS-DCD-T la scăderile de RSZ, recuperându-se mai repede decât variantele RLS.

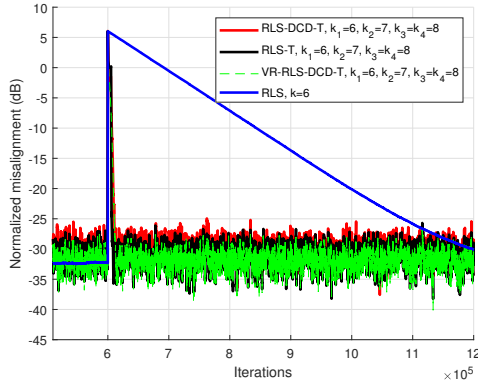


Figura 3.11 Performanța algoritmilor RLS-DCD-T, RLS-T, VR-RLS-DCD-T și RLS în scenariul de urmărire, pentru o secvență AR(1) ca intrare, cu $N = 4$ și $L = 2048$

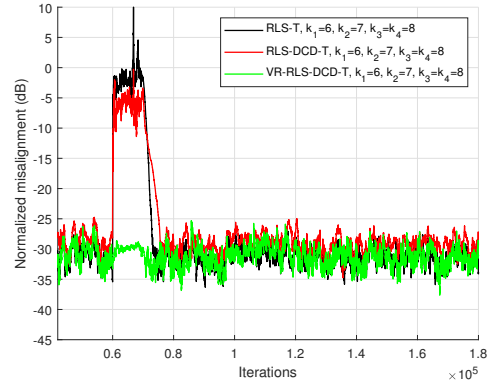


Figura 3.13 Performanța algoritmilor RLS-T, RLS-DCD-T și VR-RLS-DCD-T, pentru o secvență AR(1) ca intrare, cu $N = 4$ și $L = 2048$ când RSZ scade la -15 dB

3.8 Concluzii și perspective

Algoritmii adaptivi pentru identificarea sistemelor se confruntă cu provocări în medii nestationare și în timp real. Metodele LMS oferă un echilibru între performanță și sarcina computațională, în timp ce filtrele RLS asigură o urmărire și o convergență superioare, dar se confruntă adesea cu probleme de stabilitate și complexitate. Abordările bazate pe tensori simplifică sarcinile de dimensiuni mari folosind filtre mai scurte. Algoritmii $RLS_M-NLMS_{N-M}-T$ și $RLS-DCD-T$ reduc complexitatea, păstrând performanța. $VR-RLS-DCD-T$ excelează în condiții de RSZ scăzut, demonstrând robustețe și eficiență chiar și la niveluri ridicate de zgomot. Lucrările viitoare ar putea fi axate pe tehnici de fereastră glisantă și proiectarea FPGA pentru configurații MISO, îmbunătățind aplicațiile practice.

Capitolul 4

Algoritmi adaptivi de tip RLS bazați pe abordarea reutilizării datelor

Acest capitol îmbunătățește algoritmi adaptivi RLS printr-o abordare de reutilizare a datelor (DR), permițând multiple actualizări pe aceleași intrări pentru a spori convergența și performanța filtrului, reducând în același timp complexitatea computațională. Algoritmi DR-RLS și DR-RLS-DCD îmbunătățesc filtrarea adaptivă și urmărirea, cu acesta din urmă menținând o complexitate gestionabilă. Algoritmul R-DR-RLS-DCD adaugă robustețe în condiții de RSZ scăzut. Simulările confirmă eficacitatea acestor metode, subliniind rolul strategiilor DR în optimizarea algoritmilor RLS pentru aplicații reale. Rezultatele sunt publicate în [10–12].

4.1 Algoritmul DR-RLS

Tabel 4.1 Algoritmul DR-RLS

Pas	Acțiuni
Init.	Setează $\widehat{\mathbf{h}}(0) = \mathbf{0}_{L \times 1}$, $\mathbf{R}(0) = \delta \mathbf{I}_L$, $\delta > 0$ $\lambda = 1 - \frac{1}{KL}$, $K \geq 1$
Pentru $n = 1, 2, \dots$, număr iterații:	
1	Calculează $\mathbf{x}(n)$
2	$\mathbf{k}(n) = \frac{\mathbf{R}^{-1}(n-1)\mathbf{x}(n)}{\lambda + \mathbf{x}^T(n)\mathbf{R}^{-1}(n-1)\mathbf{x}(n)}$
3	$e(n) = d(n) - \widehat{\mathbf{h}}^T(n-1)\mathbf{x}(n)$
4	$\mathbf{R}^{-1}(n) = \frac{1}{\lambda} [\mathbf{I}_L - \mathbf{k}(n)\mathbf{x}^T(n)] \mathbf{R}^{-1}(n-1)$
5	$\alpha(n) = 1 - \mathbf{x}^T(n)\mathbf{k}(n)$
6	$\beta(n) = \frac{1 - \alpha^{N_{it}(n)}}{1 - \alpha(n)}$
7	$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \beta(n)\mathbf{k}(n)e(n)$

Această secțiune introduce algoritmul DR-RLS [13] (prezentat în Tabelul 4.1), care îmbunătățește performanța filtrului prin permiterea multiplelor actualizări pe aceleași semnale de intrare și de referință. Această abordare sporește ratele de convergență, gestionând în același timp complexitatea computațională. Vectorul câștig Kalman $\mathbf{k}(n)$ este folosit pentru a actualiza matricea inversă de covarianță și coeficienții filtrului, valorificând principiul reutilizării datelor pentru a efectua N_{it} actualizări pe seturile de date. Algoritmul consolidează eficient actualizările, reducând costul computațional la $O(L)$ operații, echilibrând urmărirea, acuratețea și nealinierea.

4.2 Algoritmul DR-RLS-DCD

Algoritmul DR-RLS-DCD [10, 11] integrează principiul DR în algoritmul adaptiv RLS-DCD pentru a îmbunătăți capacitățile de urmărire, cu o creștere ușoară a complexității aritmetice. Algoritmul aplică multiple actualizări ale coeficienților filtrului la fiecare index de timp, folosind N_{it} iterații. Semnalul de eroare și vectorul rezidual sunt ajustați iterativ, menținând calculele eficiente. Complexitatea algoritmului DR-RLS-DCD rămâne proporțională cu lungimea filtrului, oferind o convergență mai rapidă și o urmărire îmbunătățită, esențiale pentru aplicațiile în timp real. Algoritmul este rezumat în Tabelul 4.2, unde

$$e_k(n) = \begin{cases} d(n) - \widehat{\mathbf{h}}^T(n-1)\mathbf{x}(n) = e_0(n), & k = 0 \\ e_{k-1}(n) - \Delta \widehat{\mathbf{h}}_{k-1}^T(n)\mathbf{x}(n), & k \geq 1, \end{cases} \quad (4.18)$$

$$\mathbf{p}_k(n) = \begin{cases} \lambda \mathbf{r}_{N_{it}-1}(n-1) + \mathbf{r}_{e,x,0}(n), & k = 0 \\ \mathbf{r}_{k-1}(n) + \mathbf{r}_{e,x,k}(n), & k \geq 1. \end{cases} \quad (4.21)$$

Tabel 4.2 Algoritmul DR-RLS-DCD cu ponderare exponențială

Pas	Acțiuni
Init.	Setează $\widehat{\mathbf{h}}(0) = \mathbf{0}_{L \times 1}$, $\mathbf{r}(0) = \mathbf{0}_{L \times 1}$ $\mathbf{R}(0) = \delta \mathbf{I}_L$, $\delta > 0$ $\lambda = 1 - \frac{1}{KL}$, $K \geq 1$
Pentru $n = 1, 2, \dots$, număr iterații:	
1	Calculează $\mathbf{x}(n)$
2	Calculează $\mathbf{R}^{(0)}(n) = \lambda \mathbf{R}^{(0)}(n-1) + \mathbf{x}(n)\mathbf{x}^{(0)}(n) \rightarrow \mathbf{R}(n)$
3	Pentru $k = 0, 1, 2, \dots, N_{it} - 1$:
3.a	Calculează $e_k(n)$ folosind (4.18)
3.b	Calculează $\mathbf{p}_k(n)$ folosind (4.21)
3.c	$\mathbf{R}(n)\Delta \mathbf{h}_k(n) = \mathbf{p}_k(n) \xrightarrow{\text{DCD}} \Delta \widehat{\mathbf{h}}_k(n), \mathbf{r}_k(n)$
4	$\widehat{\mathbf{h}}_k(n) = \widehat{\mathbf{h}}_k(n-1) + \Delta \widehat{\mathbf{h}}_k(n)$

4.3 Algoritmul R-DR-RLS-DCD

Algoritmul R-DR-RLS-DCD [12] integrează regularizarea și DR în cadrul RLS-DCD pentru a îmbunătăți robustețea la zgomot, în special în condiții de RSZ scăzut. Parametrul de regularizare δ ajustează frecvența actualizărilor în componenta DCD, valori mai mici permițând actualizări mai frecvente, iar valori mai mari încetinind acest proces, fiind avantajos în medii zgomotoase [6, 14]. δ este determinat pe baza lungimii filtrului adaptiv, varianței semnalului de intrare și RSZ. Această integrare îmbunătățește ratele de convergență și robustețea, menținând eficiența computațională, așa cum este prezentat în Tabelul 4.3.

Tabel 4.3 Algoritmul R-DR-RLS-DCD

Init.	Setează $\widehat{\mathbf{h}}(0) = \mathbf{0}_{L \times 1}$ $\mathbf{r}(0) = \mathbf{0}_{L \times 1}$ $\lambda = 1 - \frac{1}{KL}, K \geq 1$
Pentru $n = 1, 2, \dots$, număr iterații:	
1	Calculează $\mathbf{x}(n)$
2	Calculează $\mathbf{R}^{(0)}(n) = \lambda \mathbf{R}^{(0)}(n-1) + \mathbf{x}(n)\mathbf{x}^{(0)}(n) \rightarrow \mathbf{R}(n)$
3	Pentru $k = 1, 2, \dots, N_{it} - 1$:
3.a	Calculează $e_k(n)$ folosind (4.18)
3.b	Calculează $\mathbf{p}_k(n)$ folosind (4.21)
3.c	$[\mathbf{R}(n) + \delta \mathbf{I}_L] \Delta \mathbf{h}_k(n) = \mathbf{p}_k(n) \xrightarrow{\text{DCD}} \Delta \widehat{\mathbf{h}}_k(n), \mathbf{r}_k(n)$
4	$\widehat{\mathbf{h}}_k(n) = \widehat{\mathbf{h}}_k(n-1) + \Delta \widehat{\mathbf{h}}_k(n)$

4.4 Rezultatele simulărilor

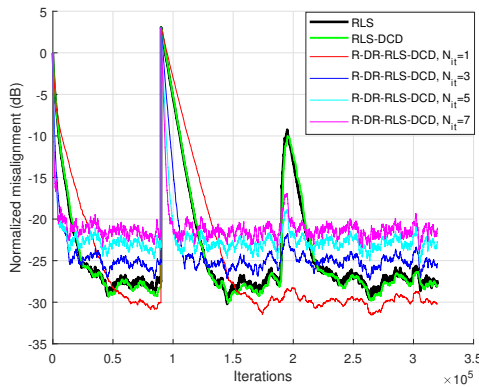


Figura 4.5 Performanța algoritmilor RLS, RLS-DCD și R-DR-RLS-DCD pentru valori diferite ale N_{it} și $N_u = 4$ fix, folosind o secvență AR(1) ca intrare.

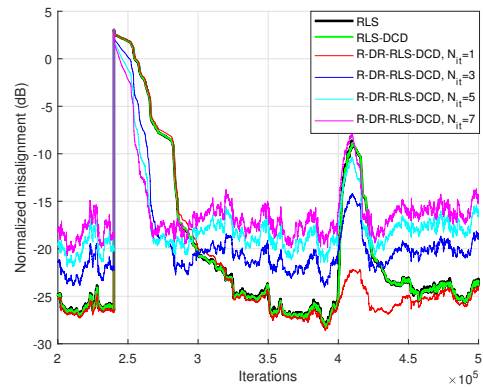


Figura 4.6 Performanța algoritmilor RLS, RLS-DCD și R-DR-RLS-DCD pentru valori diferite ale N_{it} , folosind o secvență vocală ca intrare.

Simulările pentru identificarea sistemului necunoscut au utilizat o cale de ecou din recomandarea G.168 a ITU-T, cu o lungime a răspunsului la impuls de $L = 128$. Două semnale de intrare, o secvență AR(1) și vorbire de înaltă calitate, au fost testate la un RSZ de 10 dB. Factorul de uitare a fost $\lambda = 1 - 1/(KL)$, cu parametrii DCD $N_u = 4$, $H = 1$ și $M_b = 16$.

Scenariile au implicat urmărirea (modificarea coeficienților răspunsului la impuls) și condiții de RSZ scăzut, comparând R-DR-RLS-DCD, RLS și RLS-DCD, folosind NM. Pentru cazul AR, modificările pentru urmărire au avut loc la indexul 90001, iar începând cu indexul 190001, RSZ a scăzut timp de 5000 de iterații, așa cum se arată în Figura 4.5. Pentru eșantioanele de vorbire, sistemul s-a schimbat la indexul 240001, cu o scădere de 20 dB a RSZ între 400001 și 410000, așa cum este ilustrat în Figura 4.6. Creșterea N_{it} a îmbunătățit urmărirea, dar a redus acuratețea în regimul staționar. R-DR-RLS-DCD a depășit ceilalți algoritmi în condiții de RSZ scăzut. Rezultatele au evidențiat un compromis între viteza de urmărire și acuratețea în regimul staționar, algoritmi regularizați arătând fluctuații mai mici în condiții de RSZ scăzut.

4.5 Concluzii și perspective

Acest capitol a explorat îmbunătățirile algoritmilor RLS cu complexitate redusă, începând cu metoda DR-RLS și extinzându-se la DR-RLS-DCD și R-DR-RLS-DCD.

Algoritmii DR-RLS-DCD și R-DR-RLS-DCD îmbunătățesc RLS prin integrarea tehnicii DCD, care rezolvă eficient sistemele de ecuații auxiliare folosind operații aritmetice simple, precum adunări și deplasări de biți.

Algoritmul DR-RLS oferă baza pentru dezvoltările ulterioare, demonstrând potențialul reutilizării datelor în filtrarea adaptivă. Pe această bază, DR-RLS-DCD introduce o abordare nouă de reutilizare a datelor, echilibrând viteza de urmărire, complexitatea aritmetică și acuratețea estimării. Complexitatea sa este proporțională cu lungimea filtrului, făcându-l potrivit pentru implementări hardware. Simulările confirmă performanța sa superioară față de metodele bazate pe identitatea lui Woodbury, în special în scenarii cu zgomot și intrări corelate.

R-DR-RLS-DCD îmbunătățește capacitățile prin combinarea reutilizării datelor și a tehnicilor de regularizare, sporind viteza de urmărire și robustețea în condiții de RSZ scăzut. Testele extinse demonstrează eficiența sa pe diverse semnale de intrare cu niveluri crescute de corelație, subliniind performanța sa în scenarii de urmărire și RSZ scăzut. Eficiența sa hardware îl face ideal pentru aplicații ce necesită mulți coeficienți ai filtrelor.

Direcțiile viitoare includ îmbunătățirea DR-RLS-DCD cu iterații variabile de reutilizare a datelor pentru o acuratețe mai bună și optimizarea R-DR-RLS-DCD pentru o robustețe mai mare în medii dificile. O altă perspectivă promițătoare poate fi exploatarea aplicațiilor în timp real ale acestor algoritmi.

Capitolul 5

Algoritmi adaptivi de tip RLS pentru compensarea ecoului acustic stereofonic

Acest capitol explorează sistemele SAEC cu multiple difuzoare și microfoane. Metodele tradiționale, precum LMS, întâmpină dificultăți pentru intrările corelate, în timp ce algoritmi RLS sunt mai complecși. Modelul WL (widely linear) îmbunătățește performanța în scenarii cu RSZ scăzut și vorbire simultană (double talk - DT), utilizând un filtru complex.

Algoritmii WL-RLS-LSM (WL-RLS with Line Search Method) și versiunile lor cu regularizare variabilă (VR) îmbunătățesc viteza de urmărire și acuratețea, în timp ce WL-DR-RLS-DCD permite multiple actualizări pe iterație. Simulările confirmă echilibrul între complexitate și performanță. Rezultatele sunt publicate în [15–22].

5.1 Compensarea ecoului acustic stereofonic pe baza modelului WL

Modelul SAEC folosește o configurație stereofonică cu două microfoane și două difuzoare per terminal pentru a sintetiza auzul binauricular. Fiecare difuzor se conectează la un microfon printr-un răspuns la impuls necunoscut, generând ecouri. Modelul WL simplifică problema prin combinarea semnalelor reale într-o formă complexă:

$$x(n) = x_{L_c}(n) + jx_{R_c}(n) \quad (5.1)$$

Aceasta permite encapsularea semnalelor de intrare într-un vector complex $\tilde{\mathbf{x}}(n)$, folosind un singur filtru adaptiv pentru a estima căile de ecou. Ieșirea este $y(n) = \tilde{\mathbf{h}}_t^H \tilde{\mathbf{x}}(n)$, cu $\tilde{\mathbf{h}}_t$ ca răspuns complex la impuls. Eroarea a posteriori este $e(n) = d(n) - \tilde{\mathbf{h}}^H(n) \tilde{\mathbf{x}}(n)$. Această abordare reduce complexitatea și îmbunătățește eficiența filtrării în SAEC.

5.2 Algoritmii WL-RLS-LSM

Modelul WL gestionează eficient interacțiunile semnalelor acustice complexe în configurațiile SAEC. Algoritmii WL-RLS-LSM îmbunătățesc acest lucru prin utilizarea metodei RLS cu ponderi exponențiale, unde factorul de uitare λ echilibrează urmărirea și acuratețea. Actualizările recursive pentru matricea de autocorelație $\mathbf{R}(n)$ și vectorul de corelație $\mathbf{p}(n)$ asigură o calculare eficientă, evitând inversarea directă a matricei. Aceste metode valorifică proprietățile statistice ale semnalelor difuzoarelor pentru a rezolva ecuații auxiliare, combinând RLS cu metodele de căutare liniară (LSM) pentru a menține performanța și stabilitatea, așa cum este detaliat în Tabelul 5.1. Inițializarea setează coeficienții filtrului și vectorii reziduali la zero, iar $\mathbf{R}(n)$ este inițializată pentru a evita singularitatea. Algoritmul iterează prin actualizarea vectorului de intrare:

$$\tilde{\mathbf{x}}(n) = \begin{bmatrix} x(n) & x^*(n) & \dots & x(n-L+1) & x^*(n-L+1) \end{bmatrix}^T. \quad (5.11)$$

Tabel 5.1 Algoritmii WL-RLS-LSM, cu complexitățile asociate

Pas	Acțiuni	'×'
Init.	$\tilde{\mathbf{h}}(0) = \mathbf{0}_{2L \times 1}; \quad \mathbf{r}(0) = \mathbf{0}_{2L \times 1}$ $\mathbf{R}(0) = \Phi \mathbf{I}_{2L}, \Phi > 0$	0 0
Pentru $n = 1, 2, \dots$, număr iterații:		
1	Actualizează $\tilde{\mathbf{x}}(n)$ folosind (5.1) și (5.11)	0
2	Actualizează $\mathbf{R}(n)$ folosind proprietatea deplasării temporale: $\mathbf{R}_{:,1}(n) = \lambda \mathbf{R}_{:,1}(n-1) + x^*(n) \tilde{\mathbf{x}}(n)$	$4L$
3	$\tilde{y}(n) = \tilde{\mathbf{h}}^H(n-1) \tilde{\mathbf{x}}(n)$	$8L$
4	$e(n) = d(n) - \tilde{y}(n)$	0
5	$\mathbf{p}_0(n) = \lambda \mathbf{r}(n-1) + e^*(n) \tilde{\mathbf{x}}(n)$	$4L$
6	$\mathbf{R}(n) \Delta \mathbf{h}(n) = \mathbf{p}_0(n) \xrightarrow{\text{LSM}} \Delta \tilde{\mathbf{h}}(n), \mathbf{r}(n)$	...
7	$\tilde{\mathbf{h}}(n) = \tilde{\mathbf{h}}(n-1) + \Delta \tilde{\mathbf{h}}(n)$	0

5.3 Versiunile WL-RLS-LSM

Algoritmii WL-RLS-LSM rezolvă un sistem auxiliar de ecuații prin minimizarea unei funcții de cost ce reflectă eroarea între răspunsurile dorite și cele estimate. Vectorul soluției $\Delta \tilde{\mathbf{h}}(n)$ este optimizat iterativ, îmbunătățind acuratețea răspunsului sistemului, folosind statisticile datelor de intrare, începând cu un vector zero și actualizându-se până la convergența soluției optime.

5.3.1 Algoritmul LSM cu gradient conjugat

Metoda gradientului conjugat (CG) determină iterativ un vector de direcție $\mathbf{d}(k)$ pentru a minimiza funcția de cost. Deși are o complexitate de $O(4L^2)$, este popular datorită performanței și vitezei de urmărire.

5.3.2 Algoritmul LSM cu reducere coordonată

Metoda reducerii coordonate (CD) valorifică proprietățile statistice ale matricei de autocorelație, actualizând o singură componentă a vectorului soluție în fiecare iterație, având o complexitate de $O(2LN_u)$.

5.3.3 Algoritmul LSM cu reducere dihotomică coordonată

Metoda DCD utilizează o abordare de tip „lacom” (greedy) și deplasări de biți în loc de înmulțiri, oferind actualizări eficiente. Este cea mai eficientă metodă din punctul de vedere al operațiilor aritmetice, necesitând doar adunări.

5.4 Algoritmii VR-WL-RLS-LSM

Această secțiune prezintă o îmbunătățire a RLS cu ponderi exponențiale printr-un parametru de regularizare dinamic Φ , determinat la fiecare index de timp n în funcție de raportul ecou-zgomot (ENR). Această ajustare sporește robustețea în scenarii cu ENR scăzut, modificând funcția de cost pentru a include Φ . Algoritmul VR-WL-RLS-LSM implementează această strategie, detaliată în Tabelul 5.5, incluzând inițializarea și actualizările iterative ce implică împărțiri și o operație de rădăcină pătrată pentru a calcula Φ . Variantele semnalelor $x(n)$, $d(n)$ și $\tilde{y}(n)$, notate $c(n)$, sunt estimate folosind o fereastră exponențială pentru a îmbunătăți performanța în condiții acustice dificile:

$$\tilde{\sigma}_c^2(n) = \gamma \tilde{\sigma}_c^2(n-1) + (1-\gamma) |c(n)|^2. \quad (5.45)$$

Tabel 5.5 Algoritmii VR-WL-RLS-LSM, cu complexitățile asociate

Pas	Acțiuni	'×'	'+'
Init.	$\mathbf{h}(0) = \mathbf{0}_{2L \times 1}; \quad \mathbf{r}(0) = \mathbf{0}_{2L \times 1}$	0	0
	$\mathbf{R}(0) = \tilde{\Phi}(0) \mathbf{I}_{2L}, \quad \tilde{\Phi}(0) > 0$	0	0
	$\tilde{\sigma}_x^2(n) = 0; \quad \tilde{\sigma}_d^2(n) = 0; \quad \tilde{\sigma}_y^2(n) = 0; \quad \tilde{\sigma}_w^2(n) = 0$	0	0
Pentru $n = 1, 2, \dots$, număr iterații:			
1	Actualizează $\tilde{\mathbf{x}}(n)$ folosind (5.1) și (5.11)	0	0
2	Actualizează $\mathbf{R}(n)$ folosind proprietatea deplasării temporale: $\mathbf{R}_{:,1}(n) = \lambda \mathbf{R}_{:,1}(n-1) + x^*(n) \tilde{\mathbf{x}}(n)$	4L	0
3	Actualizează $\tilde{\sigma}_x^2(n)$, $\tilde{\sigma}_d^2(n)$ folosind (5.45); $\alpha = 2L \tilde{\sigma}_x^2(n)$	8	0
4	$\tilde{y}(n) = \mathbf{h}^H(n-1) \tilde{\mathbf{x}}(n)$	8L	0
5	Actualizează $\tilde{\sigma}_y^2(n)$ folosind (5.45)	4	0
6	$\tilde{\sigma}_w^2(n) = \tilde{\sigma}_d^2(n) - \tilde{\sigma}_y^2(n) $; $\overline{\text{ENR}} = \frac{\tilde{\sigma}_y^2(n)}{\tilde{\sigma}_w^2(n)}$	0	1
7	$\tilde{\Phi}(n) = \left(1 + \sqrt{1 + \overline{\text{ENR}}}\right) \alpha / \overline{\text{ENR}}$	1	1
8	$e(n) = d(n) - \tilde{y}(n)$	0	0
9	$\mathbf{p}_0(n) = \lambda \mathbf{r}(n-1) + e^*(n) \tilde{\mathbf{x}}(n)$	4L	0
10	$[\mathbf{R}(n) + \tilde{\Phi}(n) \mathbf{I}_{2L}] \Delta \mathbf{h}(n) = \mathbf{p}_0(n) \xrightarrow{\text{LSM}} \Delta \tilde{\mathbf{h}}(n), \mathbf{r}(n)$	...	...
11	$\mathbf{h}(n) = \mathbf{h}(n-1) + \Delta \mathbf{h}(n)$	0	0

5.5 Algoritmul WL-DR-RLS-DCD

În Secțiunea 4.2, abordarea DR îmbunătățește algoritmi RLS prin actualizarea coeficienților filtrului $\tilde{\mathbf{h}}(n)$ de mai multe ori pentru același index n , folosind DCD. Prin repetarea pașilor 3-7 din Tabelul 5.1 pentru N_{it} iterații, WL-DR-RLS-DCD se aliniază cu WL-RLS-DCD când $N_{it} = 1$. Fiecare iterație actualizează estimarea filtrului $\Delta\tilde{\mathbf{h}}_q(n)$, semnalul de eroare $e_q(n)$ și vectorul rezidual $\mathbf{p}_{0,q}(n)$.

Algoritmul propus, detaliat în Tabelul 5.6, menține o complexitate proporțională cu lungimea filtrului înmulțită cu un număr mic întreg, cu N_{it} de obicei mai mic decât 10, ducând la echilibrarea vitezei de urmărire și a efortului computațional.

Tabel 5.6 Algoritmul WL-DR-RLS-DCD, cu complexitățile asociate

Pas	Acțiuni	'×'	'+'
Init.	$\tilde{\mathbf{h}}(0) = \mathbf{0}_{2L \times 1}; \quad \mathbf{r}(0) = \mathbf{0}_{2L \times 1}$ $\mathbf{R}(0) = \Phi \mathbf{I}_{2L}, \Phi > 0; \quad q = 0$	0 0	0 0
Pentru $n = 1, 2, \dots$, număr iterații:			
1	Actualizează $\tilde{\mathbf{x}}(n)$	0	0
2	Actualizează $\mathbf{R}(n)$ (deplasarea temporală): $\mathbf{R}_{:,1}(n) = \lambda \mathbf{R}_{:,1}(n-1) + x^*(n)\tilde{\mathbf{x}}(n)$	$4L$	$8L$
3	$q = q + 1$	0	0
4	Determină $e_q(n)$	$\leq 8L + 4(N_{it} - 1)N_u$	$\leq 6L - 2 + 2N_{it} + 4N_u(N_{it} - 1)$
5	Determină $\mathbf{p}_{0,q}(n)$	$4L(2N_{it} + 1)$	$4LN_{it}$
6	$\mathbf{R}(n)\Delta\mathbf{h}_q(n) = \mathbf{p}_{0,q}(n) \xrightarrow{\text{DCD}} \Delta\tilde{\mathbf{h}}_q(n), \mathbf{r}_q(n)$	0	$\leq M_b + N_{it}[(8L + 1)N_u]$
7	Determină $\tilde{\mathbf{h}}_q(n)$	0	$\leq N_{it}N_u$
8	Dacă $q < N_{it} \xrightarrow{\text{sari}} \text{Pas 3}$	0	0

5.6 Rezultatele simulărilor

În configurarea SAEC, algoritmi bazați pe WL-RLS procesează semnale de intrare cu diverse proprietăți de corelație, cum ar fi zgomotul Gaussian și vorbirea. Un bloc de predistorsiune reduce corelația între canalele de intrare, ceea ce poate îmbunătăți performanța, așa cum este detaliat în [23]. Căile de ecou sunt definite prin răspunsuri la impuls din lumea reală, iar performanța a fost măsurată folosind NM.

5.6.1 Algoritmi WL-RLS-LSM și VR-WL-RLS-LSM

Scenariile de urmărire și DT au fost simulate cu ENR de 10 dB, utilizând o secvență de vorbire ca semnal de intrare, predistorsionată cu $\alpha_r = 0.35$. Răspunsurile la impuls au avut o lungime de $L = 256$, cu $\lambda = 1 - 1/(64L)$ și, pentru algoritmul VR, $\gamma = 0.999$. Căile de ecou s-au schimbat la $t_0 = 120$, iar o situație de DT a avut loc între [230, 234] secunde, performanța fiind măsurată cu NM. Figurile 5.9, 5.10, și 5.11 arată că algoritmi

VR au demonstrat o reducere mai mică a performanței în timpul episoadelor de DT, depășind variantele non-VR cu peste 7 dB, 4.5 dB și, respectiv, 6 dB. Regularizarea a îmbunătățit acuratețea în regimul staționar, iar actualizările au rămas minim afectate de DT. Figura 5.12 indică faptul că versiunile VR prezintă curbe de aliniere similare, cu VR-WL-RLS-DCD fiind cea mai atractivă opțiune pe măsură ce ENR scade.

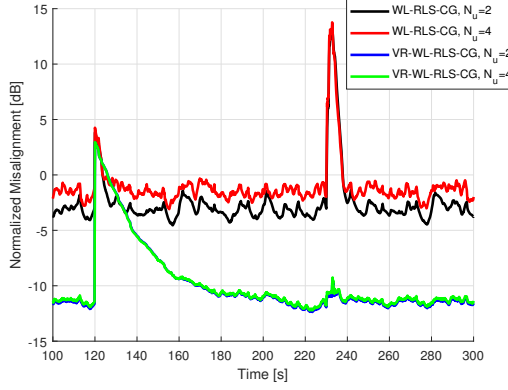


Figura 5.9 Performanța WL-RLS-CG și VR-WL-RLS-CG pentru diferite valori ale lui N_u , cu semnal vocal ca intrare și ENR = 10 dB.

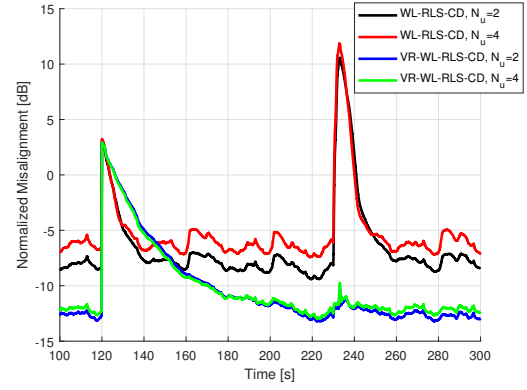


Figura 5.10 Performanța WL-RLS-CD și VR-WL-RLS-CD pentru diferite valori ale lui N_u , cu semnal vocal ca intrare și ENR = 10 dB.

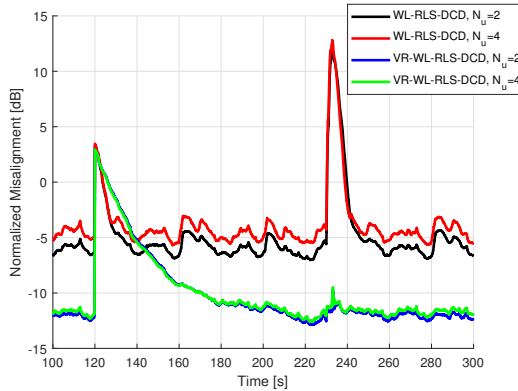


Figura 5.11 Performanța WL-RLS-DCD și VR-WL-RLS-DCD pentru diferite valori ale lui N_u , cu semnal vocal ca intrare și ENR = 10 dB.

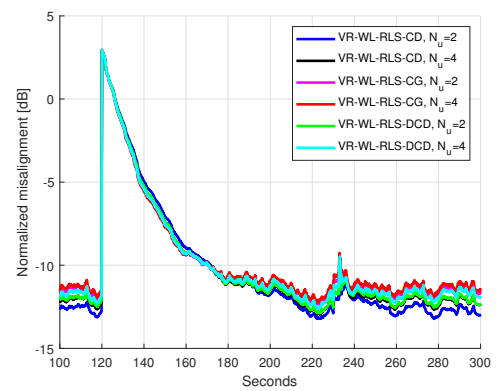


Figura 5.12 Performanța VR-WL-RLS-CG, VR-WL-RLS-CD și VR-WL-RLS-DCD pentru diferite valori ale lui N_u , cu semnal vocal ca intrare și ENR = 10 dB.

5.6.2 Algoritmul WL-DR-RLS-DCD

Performanța algoritmului WL-DR-RLS-DCD a fost validată pentru identificarea răspunsului la impuls $\tilde{\mathbf{h}}(n)$, folosind parametrii DCD $H = 1$, $N_u = 4$ și $M_b = 16$ cu, RSZ de 25 dB și 30 dB.

În figura 5.16, creșterea lui N_{it} a îmbunătățit viteza de urmărire, cu $N_{it} = 3$ fiind suficient pentru performanța în regim staționar.

Figura 5.17 arată că schimbarea căilor de ecou și a pozițiilor microfoanelor în regimul staționar a îmbunătățit urmărirea pentru valori mai mari ale lui N_{it} , până la o limită de performanță. Lipsa preprocesării a dus la un decalaj semnificativ de performanță între $\alpha_r = 0$ și $\alpha_r = 0.4$.

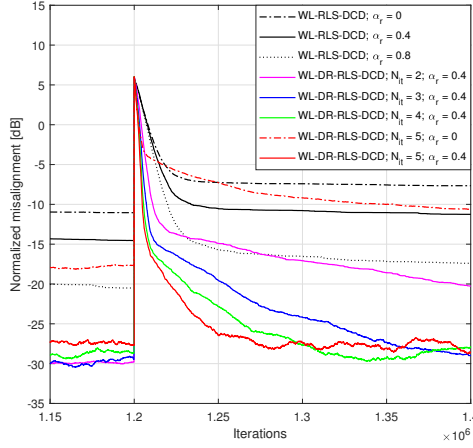


Figura 5.16 Performanța algoritmilor WL-RLS-DCD și WL-DR-RLS-DCD pentru diferite valori ale lui N_{it} și α_r . Semnalul de intrare este o secvență AR(1) cu polul 0.9, iar lungimea celor patru răspunsuri impuls necunoscute este $L = 128$ și $\lambda = 1 - 1/(64L)$. Sistemul necunoscut se schimbă la indicele de timp 1200000.

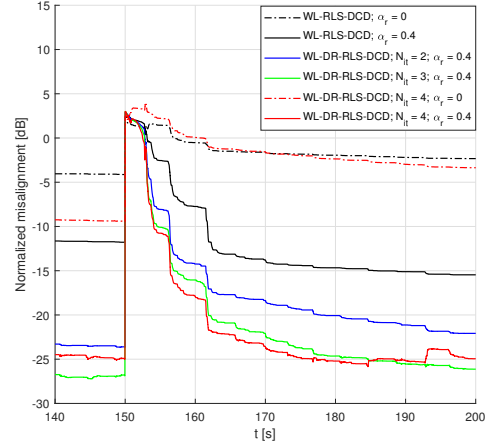


Figura 5.17 Performanța algoritmilor WL-RLS-DCD și WL-DR-RLS-DCD pentru diferite valori ale lui N_{it} și α_r . Semnalul de intrare este o secvență vocală, iar lungimea celor patru răspunsuri la impuls necunoscute este $L = 256$ și $\lambda = 1 - 1/(96L)$. Sistemul necunoscut se schimbă la $t = 150$ s (pozițiile microfoanelor sunt interschimbate).

5.7 Concluzii și perspective

Acest capitol a explorat algoritmi RLS de complexitate redusă pentru SAEC, subliniind eficiența cadrului WL în gestionarea variabilelor cu valori complexe. Dintre algoritmi WL-RLS-LSM, VR-WL-RLS-DCD se dovedește a fi deosebit de eficient, utilizând doar adunări și deplasări de biți, ceea ce îl face ideal pentru aplicații practice. Simulările au arătat că acești algoritmi reduc semnificativ NM în intervalele de DT (cu aproximativ 25 dB), variantele VR depășind versiunile non-VR, în special la niveluri mai scăzute de ENR.

Algoritmul WL-DR-RLS-DCD îmbunătățește urmărirea, menținând o complexitate redusă și adaptându-se rapid la schimbări. Acesta îmbunătățește eficient urmărirea în scenarii cu intrări puternic corelate, în ciuda unui compromis în ceea ce privește acuratețea în regimul staționar. În ansamblu, acești algoritmi echilibrează viteza de urmărire și performanța în regimul staționar, WL-DR-RLS-DCD ieșind în evidență pentru implementările hardware în sistemele SAEC. Lucrările viitoare ar putea viza optimizarea parametrilor și explorarea extensiilor multicanal.

Capitolul 6

Concluzii

Acest capitol rezumă principalele concluzii ale tezei, care avansează tehnicile de filtrare adaptivă cu aplicații practice. Cercetarea a abordat provocările legate de performanță și complexitate prin explorarea diversilor algoritmi adaptivi, îmbunătățind atât înțelegerea teoretică, cât și implementarea practică. S-au demonstrat îmbunătățiri semnificative în viteza de urmărire, ratele de convergență și eficiența computațională, facilitând aplicații robuste din lumea reală. Teza a adus contribuții inovatoare în trei domenii: îmbunătățirea identificării sistemelor cu algoritmi adaptivi bazați pe tensori, creșterea eficienței prin algoritmi RLS cu complexitate redusă folosind o abordare DR și abordarea provocărilor SAEC cu algoritmi RLS bazați pe modelul WL. Aceste progrese contribuie la dezvoltările teoretice și practice ale filtrării adaptive, oferind noi perspective și soluții.

6.1 Rezultate obținute

Cercetarea a analizat procesarea adaptivă a semnalelor, subliniind importanța și aplicațiile sale, și a definit domeniul de cercetare și obiectivele.

Au fost realizate progrese semnificative în filtrarea adaptivă, concentrându-se pe algoritmi RLS și RLS-DCD. Limitările metodelor tradiționale în medii nestaționare au fost abordate prin introducerea algoritmilor adaptivi bazați pe tensori, RLS_M-NLMS_{N-M-T} având o convergență rapidă și cerințe computaționale reduse. Algoritmii RLS-DCD-T și VR-RLS-DCD-T au menținut performanța în condiții de RSZ scăzut, având complexitate redusă. Metoda DR-RLS a condus la dezvoltarea DR-RLS-DCD și R-DR-RLS-DCD, integrând tehnica DCD pentru soluții eficiente, echilibrând viteza de urmărire, complexitatea și acuratețea, excelând în medii zgomotoase cu intrări corelate.

Pentru aplicațiile SAEC, cadrul WL a gestionat eficient variabilele cu valori complexe. Algoritmii bazați pe LSM, în special variantele VR, au arătat îmbunătățiri notabile, cu VR-WL-RLS-CD și VR-WL-RLS-DCD excelând în scenarii DT și menținând capacitățile de urmărire cu o complexitate aritmetică minimă, făcându-i ideali pentru implementări hardware.

6.2 Contribuții originale

Această secțiune oferă o listă a contribuțiilor originale. Fiecare contribuție va specifica lucrările originale de la care provine, cu referințele incluse în secțiunea următoare.

[1, 2] Contribuții la dezvoltarea unui algoritm pentru identificarea sistemelor MISO, care combină tehnicile RLS și NLMS, utilizând metode bazate pe tensori.

[3, 4] Contribuții la un nou algoritm cu complexitate redusă pentru identificarea sistemelor MISO, care integrează algoritmul RLS cu ponderi exponențiale și iterații DCD, utilizând metode bazate pe tensori.

[5, 6] Contribuții la dezvoltarea unei versiuni cu regularizare variabilă a algoritmului pentru identificarea sistemelor MISO bazat pe tensori, care integrează algoritmul RLS cu ponderi exponențiale și iterații DCD, încorporând termeni de regularizare în funcțiile de cost pentru a îmbunătăți performanța.

[7, 8] Contribuții la dezvoltarea unui algoritm RLS cu complexitate redusă, care utilizează o metodologie de reutilizare a datelor bazată pe iterații DCD, îmbunătățind performanța de urmărire și viteza de convergență în scenarii cu raport semnal-zgomot scăzut, minimizând costurile pentru implementările hardware.

[9] Contribuții la dezvoltarea unei versiuni regularizate a algoritmului RLS cu complexitate redusă din [7, 8], care integrează metoda DCD și utilizează o metodologie de reutilizare a datelor pentru a îmbunătăți viteza de urmărire și robustețea în condiții dificile de RSZ.

[10] Contribuții la dezvoltarea unei abordări bazate pe descompunere pentru SAEC, care utilizează cel mai apropiat produs Kronecker, combinând filtre mai scurte implementate cu algoritmul RLS, îmbunătățind viteza de urmărire și reducând complexitatea computațională.

[11, 12, 13] Contribuții la integrarea algoritmului RLS cu valori complexe cu diverse LSM în cadrul SAEC, oferind o analiză teoretică a modelului WL și a tehnicilor iterative (CG, CD și DCD).

[14, 15] Contribuții la dezvoltarea unei versiuni cu regularizare variabilă a familiei de filtre adaptive LSM pentru scenarii DT, concentrându-se pe ajustarea matricei de corelație pentru a încetini sau opri adaptarea în timpul perturbațiilor.

[16, 17] Contribuții la îmbinarea unei abordări de reutilizare a datelor cu algoritmul RLS și iterațiile DCD, îmbunătățind eficiența computațională și performanța de urmărire în setările SAEC pentru semnale de intrare puternic corelate.

6.3 Lista lucrărilor originale

Următoarele publicații au fost susținute de două granturi de la Ministerul Cercetării, Inovării și Digitalizării, CNCS–UEFISCDI: PN-III-P1-1.1-TE-2019-0529 și PN-III-P4-PCE-2021-0438, în cadrul PNCDI III. Suport suplimentar a fost oferit prin Programul

Național de Cercetare al Asociației Naționale a Universităților Tehnice—GNAC ARUT 2023 (Grant 36/09.10.2023, cod 133) și un grant de la Academia Oamenilor de Știință din România (AOȘR) în cadrul AOȘR-TEAMS-III (cod de înregistrare al proiectului 131/2024). Publicațiile au primit finanțare din diferite combinații ale acestor granturi.

- [1] I.-D. Fîciu, C. Stanciu, C. Anghel, C. Paleologu, and L. Stanciu, "Combinations of Adaptive Filters within the Multilinear Forms," in *Proceedings of the 2021 International Symposium on Signals, Circuits and Systems (ISSCS)*, Iași, Romania, pp. 1–4, 2021 [\[BDI, IEEE Xplore\]](#).
- [2] I.-D. Fîciu, L.-M. Dogariu, C.-L. Stanciu, and C. Paleologu, "Identification of Multilinear Forms Using Combinations of Adaptive Algorithms," in *Proceedings of the Sixth International Conference on Advances in Signal, Image and Video Processing (SIGNAL 2021)*, Valencia, Spain, pp. 11–12, IARIA, 2021.
- [3] I.-D. Fîciu, C.-L. Stanciu, C. Anghel, and C. Elisei-Iliescu, "Low-Complexity Recursive Least-Squares Adaptive Algorithm Based on Tensorial Forms," *Applied Sciences*, vol. 11, no. 18, 2021, WOS:000699216700001 [\[ISI Q1, IF 2.5\]](#).
- [4] I.-D. Fîciu, C.-L. Stanciu, L.-M. Dogariu, C. Elisei-Iliescu, and C. Paleologu, "Tensor-Based Recursive Least-Squares Algorithm with Low Computational Complexity," in *Proceedings of the Fifteenth International Conference on Sensor Technologies and Applications (SENSORCOMM 2021)*, Athens, Greece, pp. 32–33, IARIA, 2021.
- [5] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, and C. Anghel, "Tensor-Based Recursive Least-Squares Adaptive Algorithms with Low-Complexity and High Robustness Features," *Electronics*, vol. 11, no. 2, 2022, WOS:000746966000001 [\[ISI Q2, IF 2.6\]](#).
- [6] I.-D. Fîciu, C. Elisei-Iliescu, C.-L. Stanciu, and C. Paleologu, "Variable-Regularized Low-Complexity RLS Adaptive Algorithms for Bilinear Forms," in *Proceedings of the Thirteenth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE 2021)*, Porto, Portugal, pp. 9–12, IARIA, 2021.
- [7] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, C. Anghel, and R.-M. Udrea, "Low-Complexity Implementation of a Data-Reuse RLS Algorithm," in *Proceedings of the 2022 45th International Conference on Telecommunications and Signal Processing (TSP)*, Prague, Czech Republic, pp. 289–293, 2022, WOS:001070846300060 [\[ISI Proceedings\]](#).
- [8] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, C. Anghel, and C. Paleologu, "A Data-Reuse Approach for the RLS-DCD Algorithm," in *Proceedings of the 2022 IARIA Annual Congress on Frontiers in Science, Technology, Services, and Applications (IARIA Congress 2022)*, Nice, France, pp. 85–86, IARIA, 2022.
- [9] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, C. Anghel, C. Paleologu, and J. Benesty, "Low-Complexity Data-Reuse RLS Algorithm with Increased Robustness Features," in *Advanced Topics in Optoelectronics, Microelectronics, and Nanotechnologies XI*, Constanța, Romania, vol. 12493, p. 124931V, International Society for Optics and Photonics, SPIE, 2023 [\[BDI, Scopus\]](#).
- [10] I.-D. Fîciu, C. Paleologu, J. Benesty, C. Elisei-Iliescu, C.-L. Stanciu, and C. Anghel, "A Decomposition-Based RLS Algorithm for Stereophonic Acoustic Echo Cancellation," in *Proceedings of the 2022 International Symposium on Electronics and Telecommunications (ISETC)*, Timișoara, Romania, pp. 1–4, 2022 [\[BDI, IEEE Xplore\]](#).
- [11] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, C. Anghel, L. Stanciu, and R.-M. Udrea, "Efficient RLS Algorithms with Line Search Methods for Stereophonic Acoustic Echo Cancellation," in *Proceedings of the 2023 International Symposium on Signals, Circuits and Systems (ISSCS)*, Iași, Romania, pp. 1–4, 2023 [\[BDI, IEEE Xplore\]](#).

- [12] I.-D. Fîciu and C.-L. Stanciu, "Stereophonic Acoustic Echo Cancellation Adaptive Algorithms using Line Search Methods," in *SD-ETTI 2023: 1st Doctoral Symposium on Electronics, Telecommunications, and Information Technology*, Bucharest, Romania, 2023.
- [13] I.-D. Fîciu, C.-L. Stanciu, C. Paleologu, J. Benesty, C. Elisei-Iliescu, C. Anghel, and S. Ciochină, "Stereophonic Acoustic Echo Cancellation with the RLS Algorithm using the Conjugate Gradient Method," in *Advances in Signal Processing and Artificial Intelligence*, Tenerife (Canary Islands), Spain, pp. 55–57, IFSA Publishing, S. L., 2023.
- [14] C.-L. Stanciu, C. Anghel, I.-D. Fîciu, C. Elisei-Iliescu, M.-R. Udrea, and L. Stanciu, "On the Regularization of Recursive Least-Squares Adaptive Algorithms Using Line Search Methods," *Electronics*, vol. 13, no. 8, 2024, WOS:001210704600001 **[ISI Q2, IF 2.6]**.
- [15] C.-L. Stanciu, C. Anghel, C. Elisei-Iliescu, L.-M. Dogariu, I.-D. Fîciu, and C. Paleologu, "On the Regularization of a Low-Complexity Recursive Least-Squares Adaptive Algorithm," in *Proceedings of the Sixteenth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE 2024)*, Venice, Italy, pp. 26–28, 2024.
- [16] I.-D. Fîciu, C.-L. Stanciu, C. Paleologu, and J. Benesty, "Low-Complexity Data-Reuse RLS Algorithm for Stereophonic Acoustic Echo Cancellation," *Applied Sciences*, vol. 13, no. 4, 2023, WOS:000938788600001 **[ISI Q1, IF 2.5]**.
- [17] I.-D. Fîciu, C.-L. Stanciu, C. Paleologu, and F. Albu, "On the Performance of a Low-Complexity Data-Reuse RLS Algorithm for Stereophonic Acoustic Echo Cancellation," in *Proceedings of the Eighteenth International Conference on Digital Telecommunications (ICDT 2023)*, Venice, Italy, pp. 6–9, IARIA, 2023.

6.4 Perspective de dezvoltare ulterioară

Această teză evidențiază direcții promițătoare pentru cercetări viitoare, inclusiv explorarea tehnicilor de fereastră glisantă pentru a îmbunătăți adaptabilitatea în medii dinamice utilizând algoritmi adaptivi bazați pe tensori. De asemenea, implementarea designurilor FPGA cu virgulă fixă pentru configurații MISO ar putea oferi soluții eficiente pentru aplicații din lumea reală.

Pentru algoritmi RLS cu complexitate redusă care utilizează strategii de reutilizare a datelor, îmbunătățirea algoritmului DR-RLS-DCD prin variația numărului de iterații de reutilizare a datelor ar putea spori acuratețea în regimul staționar. Optimizarea R-DR-RLS-DCD pentru o robustețe mai mare în condiții dificile, cum ar fi medii cu SNR scăzut, este o altă direcție promițătoare. Implementarea acestor algoritmi în sisteme în timp real ar putea oferi perspective și progrese valoroase.

Pentru compensarea ecoului acustic stereofonic, algoritmul WL-DR-RLS-DCD prezintă oportunități pentru implementările hardware. Lucrările viitoare ar putea viza optimizarea acestor algoritmi cu parametri variabili, precum numărul de iterații de reutilizare a datelor și factorul de uitare, pentru a îmbunătăți performanța în medii acustice dificile. Explorarea descompunerii răspunsului la impuls cu aproximări de rang scăzut și extinderea la scenarii multicanal ar putea spori și mai mult aplicabilitatea lor.

Aceste perspective subliniază potențialul de inovație continuă, abordând atât avansurile teoretice, cât și provocările practice în domeniul filtrării adaptive.

Bibliografie

- [1] S. Haykin, *Adaptive Filter Theory*. Prentice-Hall information and system sciences series, Prentice Hall, 2002.
- [2] I.-D. Fîciu, C. Stanciu, C. Anghel, C. Paleologu, and L. Stanciu, "Combinations of Adaptive Filters within the Multilinear Forms," in *Proceedings of the 2021 International Symposium on Signals, Circuits and Systems (ISSCS)*, pp. 1–4, 2021.
- [3] I.-D. Fîciu, L.-M. Dogariu, C.-L. Stanciu, and C. Paleologu, "Identification of Multilinear Forms Using Combinations of Adaptive Algorithms," in *Proceedings of the Sixth International Conference on Advances in Signal, Image and Video Processing (SIGNAL 2021)*, pp. 11–12, IARIA, 2021.
- [4] I.-D. Fîciu, C.-L. Stanciu, C. Anghel, and C. Elisei-Iliescu, "Low-Complexity Recursive Least-Squares Adaptive Algorithm Based on Tensorial Forms," *Applied Sciences*, vol. 11, no. 18, 2021.
- [5] I.-D. Fîciu, C.-L. Stanciu, L.-M. Dogariu, C. Elisei-Iliescu, and C. Paleologu, "Tensor-Based Recursive Least-Squares Algorithm with Low Computational Complexity," in *Proceedings of the Fifteenth International Conference on Sensor Technologies and Applications (SENSORCOMM 2021)*, pp. 32–33, IARIA, 2021.
- [6] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, and C. Anghel, "Tensor-Based Recursive Least-Squares Adaptive Algorithms with Low-Complexity and High Robustness Features," *Electronics*, vol. 11, no. 2, 2022.
- [7] I.-D. Fîciu, C. Elisei-Iliescu, C.-L. Stanciu, and C. Paleologu, "Variable-Regularized Low-Complexity RLS Adaptive Algorithms for Bilinear Forms," in *Proceedings of the Thirteenth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE 2021)*, pp. 9–12, IARIA, 2021.
- [8] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.
- [9] D. Morgan, J. Benesty, and M. Sondhi, "On the evaluation of estimated impulse responses," *IEEE Signal Processing Letters*, vol. 5, no. 7, pp. 174–176, 1998.
- [10] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, C. Anghel, and R.-M. Udrea, "Low-Complexity Implementation of a Data-Reuse RLS Algorithm," in *Proceedings of the 2022 45th International Conference on Telecommunications and Signal Processing (TSP)*, pp. 289–293, 2022.
- [11] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, C. Anghel, and C. Paleologu, "A Data-Reuse Approach for the RLS-DCD Algorithm," in *Proceedings of the 2022 IARIA Annual Congress on Frontiers in Science, Technology, Services, and Applications (IARIA Congress 2022)*, pp. 85–86, IARIA, 2022.

- [12] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, C. Anghel, C. Paleologu, and J. Benesty, "Low-Complexity Data-Reuse RLS Algorithm with Increased Robustness Features," in *Advanced Topics in Optoelectronics, Microelectronics, and Nanotechnologies XI* (M. Vladescu, R. D. Tamas, and I. Cristea, eds.), vol. 12493, p. 124931V, International Society for Optics and Photonics, SPIE, 2023.
- [13] C. Paleologu, J. Benesty, and S. Ciochină, "Data-Reuse Recursive Least-Squares Algorithms," *IEEE Signal Processing Letters*, vol. 29, pp. 752–756, 2022.
- [14] C. Elisei-Iliescu, C. Paleologu, J. Benesty, C. Stanciu, C. Anghel, and S. Ciochină, "Recursive Least-Squares Algorithms for the Identification of Low-Rank Systems," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 5, pp. 903–918, 2019.
- [15] I.-D. Fîciu, C. Paleologu, J. Benesty, C. Elisei-Iliescu, C.-L. Stanciu, and C. Anghel, "A Decomposition-Based RLS Algorithm for Stereophonic Acoustic Echo Cancellation," in *Proceedings of the 2022 International Symposium on Electronics and Telecommunications (ISETC)*, pp. 1–4, 2022.
- [16] I.-D. Fîciu, C.-L. Stanciu, C. Elisei-Iliescu, C. Anghel, L. Stanciu, and R.-M. Udrea, "Efficient RLS Algorithms with Line Search Methods for Stereophonic Acoustic Echo Cancellation," in *Proceedings of the 2023 International Symposium on Signals, Circuits and Systems (ISSCS)*, pp. 1–4, 2023.
- [17] I.-D. Fîciu and C.-L. Stanciu, "Stereophonic Acoustic Echo Cancellation Adaptive Algorithms using Line Search Methods," in *SD-ETTI 2023: 1st Doctoral Symposium on Electronics, Telecommunications, and Information Technology*, 2023.
- [18] I.-D. Fîciu, C.-L. Stanciu, C. Paleologu, J. Benesty, C. Elisei-Iliescu, C. Anghel, and S. Ciochină, "Stereophonic Acoustic Echo Cancellation with the RLS Algorithm using the Conjugate Gradient Method," in *Advances in Signal Processing and Artificial Intelligence* (S. Y. Y. Prof., Dr., ed.), pp. 55–57, IFSA Publishing, S. L., 2023.
- [19] C.-L. Stanciu, C. Anghel, I.-D. Fîciu, C. Elisei-Iliescu, M.-R. Udrea, and L. Stanciu, "On the Regularization of Recursive Least-Squares Adaptive Algorithms Using Line Search Methods," *Electronics*, vol. 13, no. 8, 2024.
- [20] C.-L. Stanciu, C. Anghel, C. Elisei-Iliescu, L.-M. Dogariu, I.-D. Fîciu, and C. Paleologu, "On the Regularization of a Low-Complexity Recursive Least-Squares Adaptive Algorithm," in *Proceedings of the Sixteenth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE 2024)*, pp. 26–28, 2024.
- [21] I.-D. Fîciu, C.-L. Stanciu, C. Paleologu, and J. Benesty, "Low-Complexity Data-Reuse RLS Algorithm for Stereophonic Acoustic Echo Cancellation," *Applied Sciences*, vol. 13, no. 4, 2023.
- [22] I.-D. Fîciu, C.-L. Stanciu, C. Paleologu, and F. Albu, "On the Performance of a Low-Complexity Data-Reuse RLS Algorithm for Stereophonic Acoustic Echo Cancellation," in *Proceedings of the Eighteenth International Conference on Digital Telecommunications (ICDT 2023)*, pp. 6–9, IARIA, 2023.
- [23] J. Benesty, C. Paleologu, T. Gaensler, and S. Ciochină, *A Perspective on Stereophonic Acoustic Echo Cancellation*, vol. 4. Springer, 01 2011.