

**COMPETIȚIA DE PROIECTE DE CERCETARE A ACADEMIEI OAMENILOR
DE ȘTIINȚĂ DIN ROMÂNIA “AOȘR-TEAMS-IV”
EDIȚIA 2025-2026**

**O nouă aplicație hardware eficientă pentru compensarea ecoului acustic în sisteme de
teleconferință stereo folosind o schemă de descompunere inovatoare**

ETAPA 1 – RAPORT DE CERCETARE ȘTIINȚIFICĂ

Rezumatul etapei

Obiectivul specific corespunzător primei etape a proiectului (desfășurată în perioada aprilie 2025 – iulie 2025) este:
O1. Analiză de performanță a algoritmului VR-WL-RLS-NKP-DCD, în funcție de parametrul de descompunere P .

În cadrul acestei etape de cercetare au fost îndeplinite toate activitățile asociate acestui obiectiv specific, prevăzute în planul de realizare a proiectului, după cum urmează:

1. Analiză de convergență și de stare stabilă pentru algoritmul WL-RLS-NKP-DCD, relaționat la parametrul de descompunere P .
2. Proiectarea unei metode practice pentru setarea parametrului P în interiorul algoritmului.
3. Proiectarea algoritmului VR-WL-RLS-NKP-DCD cu parametru de descompunere variabil.
4. Evaluare performanțe folosind simulări în Matlab.
5. Publicare articol într-o revistă ISI Q1/Q2.
6. Elaborare documentație pentru analiză de performanță algoritmului VR-WL-RLS-NKP-DCD.

Rezultatele din cadrul acestei etape de cercetare au fost publicate în lucrările [1, 2] din lista de referințe.

1 Introducere

Datorită capacității lor de a opera în medii variabile în timp, filtrele adaptive sunt frecvent utilizate pentru a rezolva probleme reale de identificare a sistemelor necunoscute [3]. Cu toate acestea, apar provocări semnificative atunci când sunt folosite pentru identificarea răspunsurilor la impuls cu lungimi mari, în special în conjuncție cu algoritmi rapid convergenți (dar mai complecși), cum ar fi algoritmul recursiv al celor mai mici pătrate (recursive least-squares - RLS) [4]. Deși versiunile rapide ale algoritmului RLS reduc volumul de calcul [5, 6, 7], acestea se confruntă cu limitări atunci când operează cu filtre adaptive de lungime mare, în ceea ce privește convergența, capacitatea de urmărire (*tracking*) și acuratețea soluțiilor lor. Recent, o tehnică bazată pe descompunere a fost aplicată cu succes în astfel de scenarii dificile de identificare a sistemelor [8]. Aceasta exploatează cea mai apropiată descompunere a produsului Kronecker (nearest Kronecker decomposition - NKP) a răspunsului la impuls (de lungime mare) care urmează să fie identificat, în conjuncție cu aproximări de rang scăzut. Prin urmare, algoritmi bazați pe RLS dezvoltati folosind această abordare s-au dovedit a fi alegeți atractive în mai multe aplicații legate de mediul acustic [9, 10, 11]. În plus, aceștia ar putea fi utilizați în diverse alte cadre de lucru, deoarece descompunerea bazată pe NKP a fost aplicată recent în contextul multor aplicații importante, de exemplu, [12, 13, 14, 15, 16, 17, 18].

Alegerea parametrului de descompunere al algoritmului RLS bazat pe abordarea NKP, și anume RLS-NKP [9], influențează performanța generală a filtrului adaptiv. Acest parametru este legat de acuratețea aproximărilor de rang scăzut. Considerând un răspuns la impuls de lungime $L = L_1 L_2$ (cu $L_1 \geq L_2$), algoritmul RLS-NKP își obține estimarea prin combinarea coeficienților a două filtre adaptive de lungimi PL_1 și PL_2 , unde P este parametrul de descompunere, cu $P < L_2$. Soluția de rang complet” corespunde lui $P = L_2$, în timp ce în majoritatea aplicațiilor care implică sisteme de rang scăzut avem de-a face cu $P \ll L_2$. De exemplu, în anularea ecoului de rețea, putem opera cu o valoare foarte mică a lui P (de exemplu, în vecinătatea lui $L_2/8$), în timp ce răspunsurile la impulsuri acustice pot fi modelate folosind o valoare mai mare a parametrului de descompunere (de exemplu, aproape de $L/2$) [9]. Cu toate acestea, în locul setării a priori a valorii lui P pe baza acestor considerații empirice, ar fi util să se găsească o modalitate mai practică de a seta acest parametru.

În cadrul acestei etape a proiectului, propunem o soluție simplă care implică o etapă de preprocesare în primele NL iterații ale algoritmului, cu $N \geq 1$. În acest scop, algoritmul de proiecție afină (APA) [19] este utilizat pentru a obține o estimare preliminară a răspunsului la impuls, care este ulterior remodelat (prin vectorizare inversă)

într-o matrice corespunzătoare de dimensiunea $L_1 \times L_2$. Apoi, descompunerea în valori singulare (SVD) a acestei matrice oferă baza pentru alegerea parametrului de descompunere P al algoritmului RLS-NKP, care va fi utilizat în următoarele iterații ale filtrului adaptiv.

Ca și aplicație directă, anularea ecoului acustic stereofonic (stereophonic acoustic echo cancellation - SAEC) necesită identificarea a patru răspunsuri la impuls necunoscute corespunzătoare a patru perechi difuzor-microfon. Dezvoltările recente în domeniul filtrelor adaptive pentru configurațiile SAEC au permis gestionarea unui singur răspuns la impuls adaptiv cu valori complexe, în loc de cele patru filtre adaptive clasice cu valori reale. Având cadrul de lucru simplificat oferit de modelul liniar în sens larg (widely linear - WL), au fost utilizate versiuni mai avansate ale metodei recursive a celor mai mici pătrate (RLS) pentru a profita de vitezele lor superioare de urmărire atunci când se lucrează cu semnale de intrare puternic corelate (cum ar fi vorbirea). În ciuda performanțelor și stabilității numerice oferite de utilizarea versiunilor ponderate exponențial ale familiei RLS în combinație cu metodele de căutare a liniilor (line search method - LSM), configurațiile SAEC au capacități limitate de atenuare a efectelor negative cauzate de perturbațiile de nivel ridicat care afectează cele două semnale de microfon. Acesta este cazul scenariilor de dublă vorbire (double-talk - DT), care reduc considerabil precizia actualizării sistemului adaptiv. Acest proiect analizează o tehnică de regularizare pentru filtrele adaptive WL-RLS-LSM, prin ajustarea matricei de corelație asociată semnalelor de intrare și crearea unei reacții în procesul de actualizare. Metoda propusă este concepută pentru a încetini considerabil (sau chiar a îngheța) procesul de adaptare în timp ce se manifestă perturbația.

2 Analiza de performanță în funcție de parametrul de descompunere

Să luăm în considerare un filtru adaptiv acționat de semnalul de intrare cu medie zero $x(n)$, la indicele de timp discret n , și având disponibilă secvența de referință $d(n)$, funcționând în cadrul unui scenariu de identificare a unui sistem liniar cu o singură intrare și o singură ieșire (single input single output - SISO). Scopul principal este de a identifica un răspuns impulsional necunoscut de lungime L , având coeficienții grupați în vectorul \mathbf{h} . În acest context, semnalul de referință are ca rezultat $d(n) = \mathbf{h}^T \mathbf{x}(n) + w(n)$, unde T reprezintă operatorul de transpunere, $\mathbf{x}(n)$ este un vector care conține cele mai recente L eşantioane temporale ale semnalului de intrare, iar $w(n)$ este un zgomot aditiv cu medie zero, care este necorelat cu $x(n)$. Pentru simplitate, considerăm că toate secvențele sunt cu valori reale. Cu toate acestea, extinderea la cazul complex este simplă.

Semnalul de control al filtrului adaptiv este eroarea a priori, care este definită ca $e(n) = d(n) - \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n)$, unde $\hat{\mathbf{h}}(n-1)$ reprezintă o estimare a \mathbf{h} la indicele de timp $n-1$. În cazul algoritmului RLS convențional, urmând criteriul de optimizare prin metoda celor mai mici pătrate (LS) bazat pe funcția sa specifică de cost, regula de actualizare devine $\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mathbf{k}(n)e(n)$, unde $\mathbf{k}(n)$ este vectorul de amplificare Kalman de lungime L [4].

Pe de altă parte, algoritmul RLS-NKP [9] își obține estimarea pe baza unei combinații de seturi mai scurte de coeficienți, astfel încât $\hat{\mathbf{h}}(n) = \sum_{p=1}^P \hat{\mathbf{h}}_{2,p}(n) \otimes \hat{\mathbf{h}}_{1,p}(n)$, unde $\hat{\mathbf{h}}_{1,p}(n)$ și $\hat{\mathbf{h}}_{2,p}(n)$ au lungimile L_1 și respectiv L_2 , cu $L_1 L_2 = L$, $L_1 \geq L_2$ și $P < L_2$; operatorul \otimes denotă produsul Kronecker. Prin urmare, semnalul de eroare a priori poate fi exprimat în două moduri echivalente, vizând separarea celor două seturi de coeficienți:

$$e(n) = d(n) - \hat{\mathbf{h}}_1^T(n-1)\mathbf{x}_2(n) = d(n) - \hat{\mathbf{h}}_2^T(n-1)\mathbf{x}_1(n),$$

unde $\hat{\mathbf{h}}_1(n-1) = \left[\hat{\mathbf{h}}_{1,1}^T(n-1) \quad \dots \quad \hat{\mathbf{h}}_{1,P}^T(n-1) \right]^T$ și $\hat{\mathbf{h}}_2(n-1) = \left[\hat{\mathbf{h}}_{2,1}^T(n-1) \quad \dots \quad \hat{\mathbf{h}}_{2,P}^T(n-1) \right]^T$ au lungimile PL_1 și PL_2 , respectiv,

$$\mathbf{x}_1(n) = \begin{bmatrix} \left[\mathbf{I}_{L_2} \otimes \hat{\mathbf{h}}_{1,1}(n-1) \right]^T \mathbf{x}(n) \\ \vdots \\ \left[\mathbf{I}_{L_2} \otimes \hat{\mathbf{h}}_{1,P}(n-1) \right]^T \mathbf{x}(n) \end{bmatrix}, \quad \mathbf{x}_2(n) = \begin{bmatrix} \left[\hat{\mathbf{h}}_{2,1}(n-1) \otimes \mathbf{I}_{L_1} \right]^T \mathbf{x}(n) \\ \vdots \\ \left[\hat{\mathbf{h}}_{2,P}(n-1) \otimes \mathbf{I}_{L_1} \right]^T \mathbf{x}(n) \end{bmatrix},$$

și \mathbf{I}_{L_1} și \mathbf{I}_{L_2} reprezintă matricile identitate de dimensiuni $L_1 \times L_1$ și respectiv $L_2 \times L_2$. Pe baza expresiilor echivalente anterioare ale semnalului de eroare a priori, se pot construi două funcții de cost urmând criteriul LS și considerând o strategie de optimizare biliniară (adică, unul dintre sisteme este fixat în cadrul criteriului de optimizare al celuilalt). În final, actualizările algoritmului RLS-NKP au ca rezultat

$$\hat{\mathbf{h}}_1(n) = \hat{\mathbf{h}}_1(n-1) + \mathbf{k}_2(n)e(n), \quad (1)$$

$$\hat{\mathbf{h}}_2(n) = \hat{\mathbf{h}}_2(n-1) + \mathbf{k}_1(n)e(n), \quad (2)$$

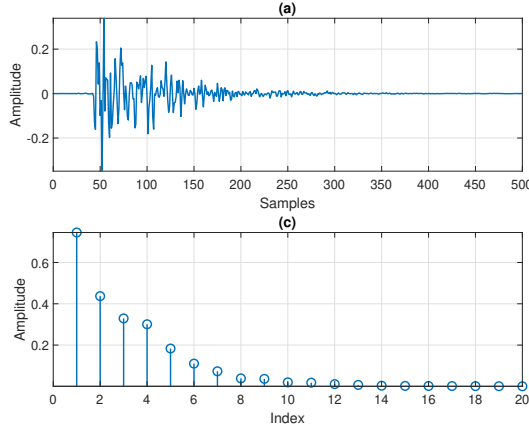


Figure 1: (a) Răspunsul la impuls acustic ($L = 500$) și (b) valorile singulare ale matricei \mathbf{H} (dimensiunea 25×20), corespunzătoare răspunsului la impuls din (a).

unde $\mathbf{k}_2(n)$ și $\mathbf{k}_1(n)$ sunt doi vectori Kalman de lungimi PL_1 și respectiv PL_2 , care pot fi calculați așa cum se arată în [9]. Practic, algoritmul RLS-NKP combină (prin intermediul produsului Kronecker) coeficienții filtrelor adaptive mai scurte din (1) și (2), pentru a obține estimarea răspunsului global la impuls. În consecință, ordinul său de complexitate este proporțional cu $\mathcal{O}[(PL_1)^2 + (PL_2)^2 + PL]$, unde ultimul termen corespunde produselor Kronecker. Pe de altă parte, algoritmul RLS convențional necesită o cantitate de calcul de $\mathcal{O}(L^2)$. Astfel, algoritmul RLS-NKP poate fi semnificativ mai puțin costisitor (pentru $P \ll L_2$) în comparație cu omologul său convențional.

Parametrul de descompunere al algoritmului RLS-NKP este legat de acuratețea aproximării de rang inferior a răspunsului la impuls $\mathbf{h} = [h_1 \ h_2 \ \dots \ h_L]^T$, cu $L = L_1 L_2$ ($L_1 \geq L_2$). Să luăm în considerare matricea asociată $\mathbf{H} = \text{ivec}(\mathbf{h}) = [\mathbf{h}_1 \ \mathbf{h}_2 \ \dots \ \mathbf{h}_{L_2}]$ de dimensiunea $L_1 \times L_2$, unde $\text{ivec}(\bullet)$ denotă operația de vectorizare inversă și $\mathbf{h}_l = [h_{(l-1)L_1+1} \ h_{(l-1)L_1+2} \ \dots \ h_{lL_1}]^T$, $l = 1, 2, \dots, L_2$ sunt vectori de lungime L_1 . Evident, rangul maxim al acestei matrice este L_2 , însă acest lucru este rar valabil în practică, unde majoritatea sistemelor (acustice) au rang scăzut, adică $\text{rang}(\mathbf{H}) = P < L_2$. Această caracteristică poate fi evaluată pe baza descompunerii SVD a \mathbf{H} [8]. Cu cât valorile singulare ale lui \mathbf{H} scad mai repede spre zero, cu atât sistemul este mai departe de rangul complet. Un exemplu în acest sens este furnizat în Fig. 1, unde este reprezentat un răspuns la impuls acustic măsurat (de lungime $L = 500$), împreună cu valorile singulare ale matricei sale corespunzătoare (de dimensiune $L_1 \times L_2 = 25 \times 20$). Pe baza acestui grafic, valoarea lui P ar putea fi aleasă în vecinătatea lui $L_2/2$. Cu toate acestea, deoarece răspunsul la impuls real nu este disponibil în aplicații din lumea reală, valoarea lui P ar putea fi stabilită a priori pe baza unor considerații empirice sau ar putea fi estimată într-un mod mai practic, așa cum se propune în cele ce urmează. Mai întâi, să considerăm că o estimare a răspunsului la impuls este disponibilă la momentul NL , cu $N \geq 1$. Prin urmare, așa cum s-a explicat anterior, coeficienții lui $\hat{\mathbf{h}}(NL)$ pot fi remodelați în matricea corespunzătoare $\hat{\mathbf{H}}(NL) = \text{ivec}[\hat{\mathbf{h}}(NL)]$ de dimensiune $L_1 \times L_2$. Valorile singulare ale acestei matrice sunt σ_l , $l = 1, 2, \dots, L_2$, cu $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{L_2} \geq 0$. Astfel, valoarea (minimă) a lui P poate fi selectată folosind un prag predefinit, astfel încât

$$\sum_{p=1}^P \sigma_p \geq \zeta \sum_{l=1}^{L_2} \sigma_l, \quad (3)$$

unde $0 \ll \zeta < 1$. Condiția anterioară poate fi interpretată și în termenii normei ℓ_1 , care este de obicei legată de raritate. Astfel, (3) poate fi rescris ca $\|\boldsymbol{\sigma}_P\|_1 \geq \zeta \|\boldsymbol{\sigma}\|_1$, unde $\boldsymbol{\sigma}_P = [\sigma_1 \ \sigma_2 \ \dots \ \sigma_P]^T$ și $\boldsymbol{\sigma} = [\sigma_1 \ \sigma_2 \ \dots \ \sigma_{L_2}]^T$. Cu alte cuvinte, cu cât vectorul $\boldsymbol{\sigma}$ este mai rar, cu atât valoarea lui P poate fi mai mică. Pentru a obține o estimare a răspunsului la impuls care poate fi utilizată pentru a evalua (3), propunem să folosim standardul APA [19] în primele iterații NL . Actualizarea sa este definită de relația:

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mu \mathbf{X}(n) [\delta + \mathbf{X}^T(n) \mathbf{X}(n)]^{-1} \mathbf{e}(n), \quad (4)$$

unde μ parametrul de mărime a pasului (cu $0 < \mu \leq 1$), $\mathbf{X}(n) = [\mathbf{x}(n) \ \mathbf{x}(n-1) \ \dots \ \mathbf{x}(n-M+1)]$ este o matrice de dimensiunea $L \times M$, unde M reprezintă ordinea de proiectie, $\delta > 0$ este o constantă de regularizare, iar $\mathbf{e}(n) = \mathbf{d}(n) - \mathbf{X}^T(n) \hat{\mathbf{h}}(n-1)$, unde $\mathbf{d}(n) = [d(n) \ d(n-1) \ \dots \ d(n-M+1)]^T$. Motivul pentru alegerea APA

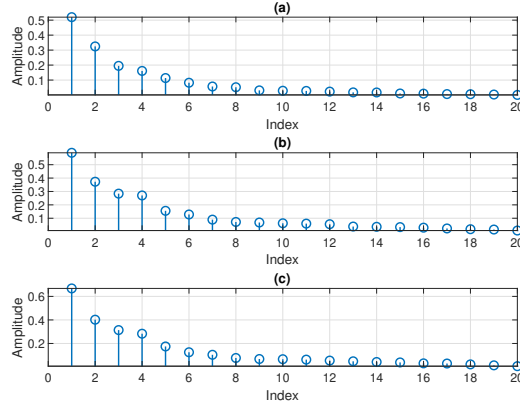


Figure 2: Valorile singulare ale matricei $\hat{\mathbf{H}}(NL)$ corespunzătoare estimării $\hat{\mathbf{h}}(NL)$ obținută prin APA (folosind $\mu = 1$ și $M = 8$) după NL iterații, cu $L = 500$ și (a) $N = 1$, (b) $N = 5$ și (c) $N = 10$. Semnalul de intrare este vorbirea și $\text{ENR} = 20$ dB.

este dublu. În primul rând, acest algoritm are o complexitate computațională moderată, care este proporțională cu $\mathcal{O}(ML)$; de asemenea, inversiunea matricei necesară în (4) poate fi evitată prin utilizarea diferitelor tehnici alternative [20]. În al doilea rând, posedă caracteristici bune de convergență, în special pentru semnale de intrare corelate (de exemplu, cum ar fi vorbirea). Rezumând, algoritmul rezultat folosește APA din (4) în primele iterații NL (cu $N \geq 1$), pentru a oferi estimarea $\hat{\mathbf{h}}(NL)$. Apoi, calculăm valorile singulare ale matricei corespunzătoare, $\hat{\mathbf{H}}(NL)$, și selectăm valoarea lui P pe baza (3). În final, această valoare a parametrului de descompunere este utilizată în cadrul algoritmului RLS-NKP. De asemenea, condiția din (4) ar putea fi verificată periodic, pentru a ajusta valoarea lui P . Cu toate acestea, această procedură ar putea afecta și configurația de descompunere a algoritmului RLS-NKP și dimensiunea variabilelor sale, astfel încât nu ar trebui utilizată frecvent.

Experimentele sunt efectuate într-un scenariu de anulare acustică a ecoului, cu scopul de a identifica răspunsul la impuls din Fig. 1(a), cu $L = 500$. În acest caz, descompunerea este efectuată folosind $L_1 = 25$ și $L_2 = 20$. Semnalul de intrare este o secvență de vorbire înregistrată, cu o rată de eșantionare de 8 kHz. Ieșirea căii de ecou (adică semnalul de ecou) este coruptă de un zgomot gaussian alb, astfel încât raportul ecou-zgomot (echo-to-noise ratio - ENR) este de 20 dB. Măsura performanței este dezalinierea normalizată (normalized misalignment), care este evaluată (în dB) ca $20\log_{10} \left[\frac{\|\mathbf{h} - \hat{\mathbf{h}}(n)\|_2}{\|\mathbf{h}\|_2} \right]$, unde $\|\cdot\|_2$ este norma ℓ_2 . Parametrii specifici ai APA sunt setați la $\mu = 1$ (adică cel mai rapid mod de convergență) și $M = 8$, vizând o estimare destul de fiabilă după primele NL iterații; de asemenea, constanta de regularizare este setată la $\delta = 20M\sigma_x^2$, unde σ_x^2 este varianța semnalului de intrare [21]. În general, N ar trebui limitat la valori rezonabil de mici. Influența sa este analizată în Fig. 2, unde valorile singulare ale $\hat{\mathbf{H}}(NL)$ sunt reprezentate pentru diferite valori ale N ; semnalul de intrare este vorbirea și $\text{ENR} = 20$ dB. După cum putem observa, chiar și pentru valori mici ale N , putem obține un set fiabil de valori singulare. De exemplu, folosind o rată de eșantionare de 8 kHz, valoarea $N = 5$ înseamnă că estimarea furnizată de APA este obținută în aproximativ 0,3 secunde.

Algoritmul RLS-NKP care utilizează APA (pentru setarea parametrului de descompunere) va fi denumit RLS-NKP-A. Performanța acestui algoritm (folosind $N = 5$) este analizată în Fig. 3 pentru diferite valori ale parametrului prag ζ din condiția (3), care duc la diferite valori ale P . Factorii de uitare ai algoritmului bazat pe RLS sunt setați la valorile lor maxime (adică egali cu unu) [9], concentrându-se doar pe nivelul de nealiniere, adică pe acuratețea estimării. Trebuie menționat că într-un scenariu de urmărire, aceste valori ar trebui reduse în mod corespunzător. APA convențional care utilizează $\mu = 1$ și $M = 8$ este, de asemenea, introdus în comparații, ca algoritm de referință.

După cum putem observa în Fig. 3, RLS-NKP-A depășește performanța algoritmului de referință, atingând o rată de convergență similară cu APA utilizând cea mai mare dimensiune a pasului ($\mu = 1$), dar atingând niveluri de nealiniere semnificativ mai mici. În partea inițială de convergență a RLS-NKP-A, putem observa momentul de "comutare", când algoritmul comută de la APA la actualizarea RLS-NKP. Pentru a îmbunătăți acuratețea APA de referință (adică pentru a reduce nealinierea acesteia), dimensiunea pasului μ sau ordinul de proiecție M ar trebui reduse, dar cu rate de convergență mai lente. Valorile lui P generate pe baza estimării APA (după primele iterații NL) sunt rezonabil mai mici în comparație cu $L_2 = 20$, adică în vecinătatea lui $L/2$.

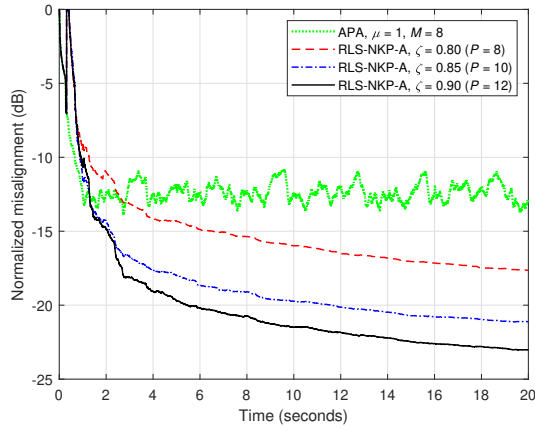


Figure 3: Dezalinierea normalizată a APA (folosind $\mu = 1$ și $M = 8$) și RLS-NKP-A (cu $L_1 = 25$, $L_2 = 20$ și $N = 5$), pentru diferite valori ale lui ζ din (3). Semnalul de intrare este vorbirea, $L = 500$ și $\text{ENR} = 20$ dB.

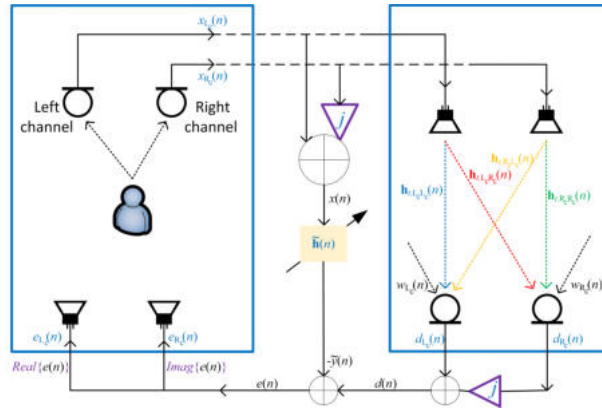


Figure 4: Modelul WL pentru sisteme SAEC.

3 Algoritmi de tip VR-WL-RLS pentru SAEC

Pentru configurația SAEC (Fig. 4), semnalele generate de cele două difuzoare reprezintă informațiile de intrare. Notăm indicele de timp discret cu n și scriem semnalele acustice corespunzătoare prin $x_{Lc}(n)$ (pentru *canalul stâng*), respectiv prin $x_{Rc}(n)$ (pentru *canalul drept*). Modelul WL combină cele două semnale cu valori reale într-o singură notăție cu valori complexe și se scrie:

$$x(n) = x_{Lc}(n) + jx_{Rc}(n), \quad (5)$$

unde $j = \sqrt{-1}$ [22, 23]. În mod corespunzător, putem exprima cei doi vectori $N \times 1$ care cuprind ultimele eșantioane asociate cu cele două canale acustice sub formele

$$\mathbf{x}_{Lc}(n) = [x_{Lc}(n) \quad x_{Lc}(n-1) \quad \cdots \quad x_{Lc}(n-N+1)]^T, \quad (6)$$

$$\mathbf{x}_{Rc}(n) = [x_{Rc}(n) \quad x_{Rc}(n-1) \quad \cdots \quad x_{Rc}(n-N+1)]^T, \quad (7)$$

unde exponentul T reprezintă operatorul de transpunere. De asemenea, putem cumula toate informațiile de intrare într-un singur vector cu $N \times 1$ valori complexe, astfel:

$$\mathbf{x}(n) = \mathbf{x}_{Lc}(n) + j\mathbf{x}_{Rc}(n). \quad (8)$$

În plus, notăm prin $\mathbf{h}_{t,LcLc}$, $\mathbf{h}_{t,LcRc}$, $\mathbf{h}_{t,RcLc}$, respectiv $\mathbf{h}_{t,RcRc}$, cele patru răspunsuri la impuls de dimensiuni $N \times 1$ asociate cu căile de ecou difuzor-microfon. Eșantioanele de ieșire corespunzătoare canalelor stâng și drept sunt obținute prin utilizarea intrărilor din (6), respectiv (7) [23, 24] și se pot scrie

$$y_{Lc}(n) = \mathbf{h}_{t,LcLc}^T \mathbf{x}_{Lc}(n) + \mathbf{h}_{t,RcLc}^T \mathbf{x}_{Rc}(n), \quad (9)$$

$$y_{Rc}(n) = \mathbf{h}_{t,LcRc}^T \mathbf{x}_{Lc}(n) + \mathbf{h}_{t,RcRc}^T \mathbf{x}_{Rc}(n), \quad (10)$$

unde se poate exprima și un semnal de ieșire cu valori complexe sub forma

$$y(n) = y_{L_c}(n) + jy_{R_c}(n). \quad (11)$$

Cele patru răspunsuri la impuls acustice cu valori reale pot fi combinate în două sisteme complexe cu valori $N \times 1$

$$\mathbf{h}_\alpha = \frac{\mathbf{h}_{t,L_c L_c} + \mathbf{h}_{t,R_c R_c}}{2} + j \frac{\mathbf{h}_{t,R_c L_c} - \mathbf{h}_{t,L_c R_c}}{2}, \quad (12)$$

$$\mathbf{h}_\beta = \frac{\mathbf{h}_{t,L_c L_c} - \mathbf{h}_{t,R_c R_c}}{2} - j \frac{\mathbf{h}_{t,R_c L_c} + \mathbf{h}_{t,L_c R_c}}{2}, \quad (13)$$

și se poate introduce o singură notație pentru gestionarea tuturor căilor de ecou cu valori reale de la difuzor la microfon, sub forma unui răspuns la impuls complex cu $2N \times 1$ valori, utilizând operația de întrețesere:

$$\mathbf{h}_t = [h_{\alpha,0} \ h_{\beta,0} \ \dots \ h_{\alpha,N-1} \ h_{\beta,N-1}]^T, \quad (14)$$

unde $h_{\alpha,l}$ și $h_{\beta,l}$, cu $l = 0, 1, \dots, N-1$, sunt elementele vectorilor \mathbf{h}_α și \mathbf{h}_β [22]. Informația de intrare poate fi, de asemenea, aranjată prin utilizarea operației de intercalare. Scriem vectorul complex cu $2N \times 1$ valori

$$\mathbf{x}_{in}(n) = [x(n) \ x^*(n) \ \dots \ x(n-N+1) \ x^*(n-N+1)]^T, \quad (15)$$

și putem obține o expresie mai simplă pentru rezultatul complex, astfel:

$$y(n) = \mathbf{h}_t^H \mathbf{x}_{in}(n). \quad (16)$$

unde H reprezintă operatorul hermitic [22]. Cu ajutorul gestionării îmbunătățite oferite de modelul WL, scriem și semnalul de referință complex ca

$$d(n) = y(n) + w(n), \quad (17)$$

unde $w(n) = w_{L_c}(n) + jw_{R_c}(n)$ reprezintă zgomotul complex care afectează canalele microfonului. Componentele sale corespunzătoare [adică $w_{L_c}(n)$, respectiv $w_{R_c}(n)$] nu sunt corelate cu semnalele difuzorului. Mai mult, având în vedere estimarea complexă $\tilde{y}(n)$ generată de sistemul SAEC pentru ieșirea $y(n)$, prin utilizarea unui singur filtru adaptiv $\tilde{\mathbf{h}}(n)$ cu $2N \times 1$ coeficienți complecși, eroarea a posteriori poate fi definită ca

$$e(n) = d(n) - \tilde{y}(n) = d(n) - \tilde{\mathbf{h}}^H(n) \mathbf{x}_{in}(n). \quad (18)$$

Modelul WL descris simplifică gestionarea configurației SAEC prin utilizarea numărului minim posibil de notații pentru toate semnalele implicate. În plus, patru răspunsuri la impuls acustice necunoscute, care în mod normal ar necesita patru algoritmi adaptivi, sunt acum combinate într-un singur sistem cu valori complexe, estimat folosind un singur filtru adaptiv. Complexitatea aritmetică generală și performanțele asociate nu se modifică [22, 24].

Versiunile ponderate exponențial ale metodelor adaptive RLS sunt dezvoltate pe baza funcției de cost

$$\mathcal{J}_{LS}[\tilde{\mathbf{h}}(n)] = \sum_{i=1}^n \lambda^{n-i} [d(i) - \tilde{\mathbf{h}}^H(n) \mathbf{x}_{in}(i)]^2, \quad (19)$$

unde $0 < \lambda \leq 1$ reprezintă factorul de uitare [4, 22]. Valoarea lui λ este asociată cu un compromis făcut între capacitățile de urmărire ale algoritmului, pe de o parte, respectiv precizia pe care o are în starea staționară, pe de altă parte. Pentru valori mai mici ale lui λ , memoria algoritmului este diminuată, iar contribuțiile asociate cu informațiile mai vechi devin neglijabile mai rapid. În astfel de cazuri, algoritmul se adaptează mai rapid la orice modificări care ar putea apărea în sistemul necunoscut care urmează să fie identificat. Când valoarea lui λ este mai aproape de 1 (adică, la memorie infinită), urmărirea modificărilor este mai lentă. Cu toate acestea, sunt stocate mai multe informații despre semnalele difuzorului, respectiv microfonului, simultan. Astfel, precizia RLS este mai bună în starea staționară, iar neajustarea scade [4]. Folosim factorul de uitare și exprimăm notațiile recursive

$$\mathbf{R}(n) = \lambda \mathbf{R}(n-1) + \mathbf{x}_{in}(n) \mathbf{x}_{in}^H(n), \quad (20)$$

$$\mathbf{p}(n) = \lambda \mathbf{p}(n-1) + \mathbf{x}_{in}(n) d(n), \quad (21)$$

unde $\mathbf{R}(n)$ este estimatul de $2N \times 2N$ al matricei de corelație asociată semnalului de intrare, respectiv $\mathbf{p}(n)$ reprezintă vectorul cu $2N \times 1$ valori de corelație dintre semnalul de intrare și referința complexă [4, 22]. În consecință, putem folosi notațiile menționate mai sus, iar minimizarea funcției de cost în (19) conduce la sistemul de ecuații

$$\mathbf{R}(n) \tilde{\mathbf{h}}(n) = \mathbf{p}(n). \quad (22)$$

Table 1 Algoritmul WL-RLS-LSM

Pas	Acțiune	×
	Inițializare :	
	$\tilde{\mathbf{h}}(0) = \mathbf{0}_{2N \times 1}$;	0
	$\mathbf{r}(0) = \mathbf{0}_{2N \times 1}$;	0
	$\mathbf{R}(0) = \Phi \mathbf{I}_{2N}$, $\Phi > 0$	0
	Pentru $n = 1, 2, 3 \dots$	
1	Actualizare $\mathbf{x}_{in}(n)$ folosind (5) și (15)	0
2	Actualizare $\mathbf{R}(n)$ folosind deplasare În timp $\mathbf{R}_{:,1}(n) = \lambda \mathbf{R}_{:,1}(n-1) + x^*(n) \mathbf{x}_{in}(n)$	$4N$
3	$\tilde{y}(n) = \tilde{\mathbf{h}}^H(n-1) \mathbf{x}_{in}(n)$	$8N$
4	$e(n) = d(n) - \tilde{y}(n)$	0
5	$\mathbf{p}_0(n) = \lambda \mathbf{r}(n-1) + e^*(n) \mathbf{x}_{in}(n)$	$4N$
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

Soluția directă pentru (22) necesită calcularea inversei matricei $\mathbf{R}(n)$ la fiecare indice de timp n . O astfel de abordare este prohibitivă pentru cipurile moderne, deoarece numărul obișnuit de coeficienți de filtrare pentru aplicațiile SAEC este de ordinul miilor. O alternativă populară la soluția directă este metoda RLS cu valori complexe bazată pe lema inversiunii matriceale (cunoscută și sub numele de identitatea lui Woodbury), care este adesea utilizată ca referință pentru lucrările de cercetare [4]. În restul lucrării, vom numi algoritmul menționat metoda WL-RLS. Aceasta necesită o cantitate de înmulțiri complexe proporțională cu $4N^2$ și, de asemenea, $4N$ diviziuni cu valori reale pentru fiecare iterație a filtrului. În ciuda reducerii efortului de calcul cu un ordin de mărime, complexitatea generală este încă considerată dificil de abordat pentru aplicațiile SAEC, care ar putea fi afectate și de problemele de stabilitate numerică asociate metodei menționate.

Cercetările fundamentale au condus la reinterpretarea sistemului din (22), pentru a exploata proprietățile statistice ale semnalelor difuzoarelor așa cum sunt reflectate în matricea de corelație $\mathbf{R}(n)$ [5, 6, 7]. În consecință, metodele RLS ponderate exponențial pot fi combinate cu diferite variante LSM pentru a rezolva un sistem auxiliar de ecuații. Algoritmii WL-RLS-LSM rezultați pot oferi performanțe comparabile cu WL-RLS, fără a manifesta probleme de stabilitate numerică [5, 24]. Mai multe versiuni au, de asemenea, complexități aritmetice mai mici în raport cu WL-RLS. Pașii metodei generale WL-RLS-LSM sunt prezentați în Tabelul 1, unde sunt afișate și cantitățile corespunzătoare ale multiplicărilor cu valori reale necesare. Etapa de inițializare rulează o singură dată și folosește vectorul $\mathbf{0}_{2N \times 1}$ cu $2N \times 1$ cu valori zero pentru a seta valorile coeficienților de filtrare $\tilde{\mathbf{h}}(0)$, respectiv ale *vectorului rezidual* $\mathbf{r}(0)$ cu $2N \times 1$ valori. Mai mult, matricea identitate $\mathbf{I}_{2N \times 2N}$ este utilizată pentru a completa estimarea matricei de corelație $\mathbf{R}(0)$ cu zerouri pe toate pozițiile, cu excepția diagonalei principale, unde constanta pozitivă cu valoare reală mică Φ este utilizată pentru a evita proprietatea de matrice nesingulară în etapele inițiale ale algoritmului. În forma din Tabelul 1, efectul lui Φ devine neglijabil după ce WL-RLS-LSM rulează pentru un număr de iterații, datorită lui λ .

Pașii 1-7 se execută de fiecare dată pentru fiecare indice de timp n , iar primele patru etape sunt aceleași ca și celelalte metode RLS ponderate exponențial care lucrează cu valori complexe în cadrul SAEC [22, 23]. Pașii 1 și 2 efectuează actualizarea vectorului de intrare $\mathbf{x}_{in}(n)$, respectiv a matricei de corelație $\mathbf{R}(n)$. Aparent, calculul acestuia din urmă folosind (20) necesită un număr de multiplicări proporțional cu pătratul lui $2N$. Cu toate acestea, vectorul de intrare are proprietatea *time-shift* (deplasare în timp) și poate fi exprimat ca

$$\mathbf{x}_{in}(n) = [x(n) \ x^*(n) \ \mathbf{x}_{in,1:2N-2}^T(n-1)]^T, \quad (23)$$

unde $\mathbf{x}_{in,1:2N-2}(n-1)$ este vectorul $(2N-2) \times 1$ care cuprinde ultimele $2N-2$ eșantioane de intrare cu valori complexe la indicele de timp $n-1$. În mod corespunzător, $\mathbf{R}(n)$ poate fi actualizat prin exploatarea structurii sale redundante. Matricea este hermitică și fiecare două coloane consecutive, începând de la prima, conțin aceleași informații, fiecare pereche de valori fiind conjugată și având pozițiile schimbate în coloana vecină [5, 24, 26]. Aceleași reguli se aplică și pentru compoziția numerică a rândurilor. Astfel, prin copierea submatricei $(2N-2) \times (2N-2)$ din stânga sus a $\mathbf{R}(n-1)$ în submatricea $(2N-2) \times (2N-2)$ din dreapta jos a $\mathbf{R}(n)$ și calcularea primei coloane $\mathbf{R}_{:,1}(n)$ ca

$$\mathbf{R}_{:,1}(n) = \lambda \mathbf{R}_{:,1}(n-1) + x^*(n) \mathbf{x}_{in}(n), \quad (24)$$

atunci a doua coloană și primele două rânduri pot fi determinate prin copierea valorilor din prima coloană, luând în considerare proprietățile enumerate. Am considerat că λ poate fi ales folosind forma $\lambda = 1 - 1/(2NK)$, unde N și K

Table 2 Algoritm VR-WL-RLS-LSM

Step	Actions	×	/
	Inițializare :		
	$\tilde{\mathbf{h}}(0) = \mathbf{0}_{2N \times 1}$;	0	0
	$\mathbf{r}(0) = \mathbf{0}_{2N \times 1}$;	0	0
	$\mathbf{R}(0) = \tilde{\Phi}(0)\mathbf{I}_{2N}$, $\tilde{\Phi}(0) > 0$	0	0
	$\tilde{\sigma}_x^2(n) = 0$; $\tilde{\sigma}_d^2(n) = 0$; $\tilde{\sigma}_y^2(n) = 0$; $\tilde{\sigma}_w^2(n) = 0$	0	0
	Pentru $n = 1, 2, 3, \dots$		
1	Actualizare $\mathbf{x}_{in}(n)$ folosind (5) și (15)	0	0
2	Actualizare $\mathbf{R}(n)$ folosind time-shift $\mathbf{R}_{:,1}(n) = \lambda \mathbf{R}_{:,1}(n-1) + x^*(n)\mathbf{x}_{in}(n)$	4N	0
3	Actualizare $\tilde{\sigma}_{x_{in}}^2(n)$, $\tilde{\sigma}_d^2(n)$; $\alpha = 2N\tilde{\sigma}_{x_{in}}^2(n)$	8	0
4	$\tilde{y}(n) = \tilde{\mathbf{h}}^H(n-1)\mathbf{x}_{in}(n)$	8N	0
5	Actualizare $\tilde{\sigma}_y^2(n)$	4	0
6	$\tilde{\sigma}_w^2(n) = \left \tilde{\sigma}_d^2(n) - \tilde{\sigma}_y^2(n) \right $; $\widetilde{\text{ENR}} = \frac{\tilde{\sigma}_y^2(n)}{\tilde{\sigma}_w^2(n)}$	0	1
7	$\tilde{\Phi}(n) = \left(1 + \sqrt{1 + \widetilde{\text{ENR}}} \right) \alpha / \widetilde{\text{ENR}}$	1	1
8	$e(n) = d(n) - \tilde{y}(n)$	0	0
9	$\mathbf{p}_0(n) = \lambda \mathbf{r}(n-1) + e^*(n)\mathbf{x}_{in}(n)$	4N	0
10	$\left[\mathbf{R}(n) + \tilde{\Phi}(n)\mathbf{I}_{2N} \right] \Delta \mathbf{h}(n) = \mathbf{p}_0(n) \xrightarrow{\text{LSM}} \Delta \tilde{\mathbf{h}}(n), \mathbf{r}(n)$
11	$\tilde{\mathbf{h}}(n) = \tilde{\mathbf{h}}(n-1) + \Delta \tilde{\mathbf{h}}(n)$	0	0

pot fi exprimate ca puteri ale lui 2. Astfel, orice înmulțire efectuată cu λ poate fi înlocuită cu o deplasare binară și o scădere. Mai mult, se poate deduce că aproximativ un sfert din valorile complexe care compun $\mathbf{R}(n)$ trebuie stocate în memorie. Complexitatea pasului 2 este, prin urmare, proporțională cu N în termeni de înmulțiri.

Deoarece calculul rezultatului filtrului din pasul 3 necesită $8N$ înmulțiri, iar pasul 4 (determinarea erorii a priori) nu utilizează operații aritmetice mai complexe decât adunările, primii patru pași necesită doar adunări și înmulțiri. Efortul total corespunzător este proporțional cu lungimea filtrului adaptiv $\mathcal{O}(N)$. Discuția privind restul complexității aritmetice este mai elaborată atunci când ne referim la pașii 5, 6 și 7, care implementează soluția auxiliară asociată sistemului din (22). Sistemul auxiliar de ecuații

$$\mathbf{R}(n)\Delta \mathbf{h}(n) = \mathbf{p}_0(n), \quad (25)$$

care trebuie rezolvată în pasul 6, oferă o estimare pentru *vectorul soluție* $\Delta \tilde{\mathbf{h}}(n)$ și necesită calcularea *componentei reziduale* $\mathbf{p}_0(n)$ (pasul 5), prin utilizarea *vectorului rezidual* $\mathbf{r}(n-1)$ generat la iterația anterioară a filtrării (ponderat cu factorul de uitare), cumulat cu produsul dintre eroare și vectorul semnalului de intrare [5, 24]. Considerând că λ poate avea forma menționată la pasul 2 și structura redundantă a lui $\mathbf{x}_{in}(n)$, singurele înmulțiri necesare pentru pasul 5 corespund înmulțirii dintre forma conjugată a lui $e(n)$ și jumătatea valorilor care compun $\mathbf{x}_{in}(n)$.

Așa cum a fost deja menționat, sistemul rezolvat în etapa 6 generează vectorul soluție estimat $\Delta \tilde{\mathbf{h}}(n)$ și o versiune actualizată a vectorului rezidual. Acesta din urmă va fi utilizat din nou în următoarea iterație a filtrului adaptiv (în etapa 5). Mai mult, considerând sistemul complex cu valori necunoscute ca fiind fix pentru un număr dat de iterații, pe măsură ce filtrul adaptiv converge, valorile absolute ale părților reale și imaginare care compun $\mathbf{r}(n)$ tind să scadă [26]. Sarcina din etapa 6 este potrivită pentru LSM-uri, care pot calcula ambii vectori menționați, $\Delta \tilde{\mathbf{h}}(n)$ și $\mathbf{r}(n)$. Există mai multe metode care pot fi utilizate în această etapă a WL-RLS-LSM [6]. Deoarece se bazează mai mult pe proprietăți specifice ale semnalului de intrare, ale matricei de corelație $\mathbf{R}(n)$, respectiv ale $\mathbf{r}(n)$, complexitatea aritmetică tinde să scadă. Cu toate acestea, proprietățile de corelație ale unor semnale, cum ar fi vorbirea, tind să fie staționare pe perioade limitate de timp, cum ar fi variațiile corespunzătoare determinate în $\mathbf{R}(n)$. Validarea unor variante LSM cu complexitate redusă este necesară, în special atunci când sunt combinate cu alte caracteristici. În final (pasul 7) vectorul soluție este adunat la setul anterior de coeficienți pentru a genera o nouă estimare.

În [25], funcția de cost din (19), corespunzătoare metodelor RLS ponderate exponențial, a fost îmbunătățită pentru a lua în considerare o adăugare permanentă la diagonala principală a matricei de corelație. Modificarea își propune să schimbe scopul parametrului de regularizare Φ , care este introdus în mod tradițional în etapa de inițializare pentru a evita proprietatea nesingulară a lui $\mathbf{R}(n)$. În consecință, valoarea lui Φ nu va fi aleasă arbitrar și va fi determinată la fiecare indice de timp n în funcție de raportul ecou-zgomot (*echo-to-noise* - ENR) estimat.

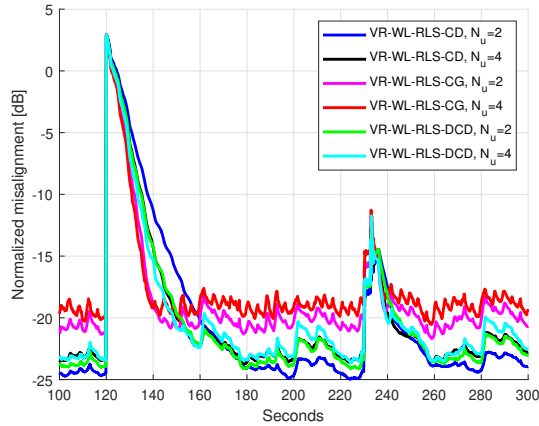


Figure 5: Dezalinare pentru VR-WL-RLS-CG, VR-WL-RLS-CD și VR-WL-RLS-DCD, cu valori diferite ale lui N_u . Semnalul de intrare este o secvență de vorbire. Lungimea celor patru căi de ecou necunoscute este $N = 256$, $\lambda = 1 - 1/(64N)$ și $\gamma = 0.999$. Căile de ecou se modifică la indicele de timp $t_0 = 120$, iar o situație de dublă vorbire apare în intervalul de timp $[230, 234]$ secunde. ENR = 25 dB.

Abordarea propusă vizează atenuarea efectelor scenariilor DT, atunci când valoarea ENR este de obicei foarte mică. O versiune regularizată variabilă (VR) a WL-RLS-LSM este prezentată în Tabelul 2, care estimează la fiecare indice de timp n valoarea ENR, respectiv varianța $\sigma_{x_{in}}^2$. Am considerat lungimea filtrului $2N$ ca o putere de doi și am exclus multiplicările corespunzătoare (acestea sunt efectuate folosind deplasări de biți, la fel ca în cazul Tabelului 1).

Trei tipuri de LSM-uri sunt combinate cu metoda VR aplicată RLS-ului cu valori complexe, și anume gradientul conjugat (conjugate gradient - CG), descendența coordonatelor (coordinate descent - CD) și descendența dihotomică a coordonatelor (dichotomous coordinate descent - DCD). Acestea generează estimări pentru vectorul soluție $\tilde{\mathbf{h}}(n)$, respectiv un vector rezidual $\mathbf{r}(n)$. LSM-urile prezentate au complexități diferite și exploatează (până la anumite niveluri) proprietățile statistice ale datelor de intrare. CG este cea mai costisitoare metodă, ceea ce duce la o complexitate aritmetică generală proporțională cu pătratul lungimii filtrului adaptiv VR-WL-RLS-CG. Metodele CD și DCD sunt progresiv mai simple în ceea ce privește volumul aritmetic și sunt concepute pentru a exploata proprietățile statistice ale semnalelor de intrare. VR-WL-RLS-CD și VR-WL-RLS-DCD necesită înmulțiri $\mathcal{O}(2N)$, cu avantajul celei de-a doua, care rezolvă sistemul auxiliar de ecuații folosind doar adunări și deplasări de biți.

Rezultatele simulării (Fig. 5) demonstrează că variantele VR ale WL-RLS-LMS-urilor scad NM cu aproximativ 25 dB în timpul intervalelor DT, cu compromisul reducerii vitezelor de urmărire ale algoritmilor corespunzători. Mai mult, VR-WL-RLS-LMS-urile au o precizie mai bună în stare staționară în comparație cu versiunile non-VR (cel puțin 2-3 dB în termeni de NM pentru ENR=25 dB și mai mult de 7 dB în termeni de NM pentru ENR=10 dB). Diferența de performanță în stare staționară este mai bună în favoarea variantelor VR pe măsură ce ENR scade. Rezultatele simulării (Fig. 6) au arătat că natura *lacomă* a iterațiilor DCD compensează parțial pierderea tracking pentru VR-WL-RLS-DCD și îi permite acestuia să se potrivească cu performanțele de urmărire generate de VR-WL-RLS-CG. Performanțele robuste asociate în condiții ENR scăzut (inclusiv situații DT), combinate cu complexitatea aritmetică generală (care este direct proporțională cu lungimea filtrului adaptiv) fac din VR-WL-RLS-DCD un candidat atractiv pentru implementări hardware pentru SAEC. Cercetările vor analiza implementările hardware eficiente (cu virgulă fixă versus cu virgulă mobilă) ale WL-RLS-DCD care încorporează o abordare VR.

References

- [1] R. A. Otopoleanu, C. L. Stanciu, C. Paleologu, J. Benesty, L. M. Dogariu, and S. Ciochină, “A robust RLS-DCD adaptive algorithm with variable regularization parameter,” in *Proc. IEEE ISSCS*, 2025, 4 pages.
- [2] R. A. Otopoleanu, C. Paleologu, J. Benesty, L. M. Dogariu, C. L. Stanciu, and S. Ciochină, “Robust data-reuse regularized recursive least-squares algorithms,” submitted to *Sensors*, 2025, 20 pages.
- [3] J. Benesty and Y. Huang, Eds., *Adaptive Signal Processing—Applications to Real-World Problems*. Berlin, Germany: Springer-Verlag, 2003.
- [4] S. Haykin, *Adaptive Filter Theory*. Fourth Edition, Upper Saddle River, NJ, USA: Prentice-Hall, 2002.
- [5] Y. V. Zakharov, G. P. White, and J. Liu, “Low-complexity RLS algorithms using dichotomous coordinate descent iterations,” *IEEE Trans. Signal Processing*, vol. 56, pp. 3150–3161, July 2008.
- [6] J. Liu, Y. V. Zakharov and B. Weaver, “Architecture and FPGA design of dichotomous coordinate descent algorithms,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, pp. 2425–2438, Nov. 2009.
- [7] Y. V. Zakharov and V. H. Nascimento, “DCD-RLS adaptive filters with penalties for sparse identification,” *IEEE Trans. Signal Processing*, vol. 61, pp. 3198–3213, June 2013.

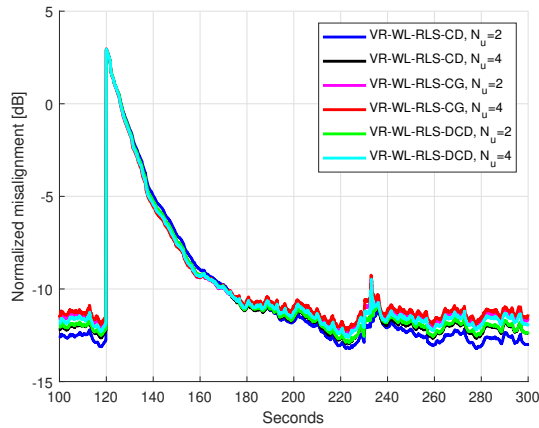


Figure 6: Dezalinierarea pentru VR-WL-RLS-CG, VR-WL-RLS-CD și VR-WL-RLS-DCD, cu valori diferite ale lui N_u . Semnalul de intrare este vorbire. Lungimea celor patru căi de ecou necunoscute este $N = 256$, $\lambda = 1 - 1/(64N)$ și $\gamma = 0.999$. Căile de ecou se modifică la indicele de timp $t_0 = 120$, iar o situație de dublă vorbire apare în intervalul de timp $[230,234]$ secunde. $ENR = 10$ dB.

- [8] C. Paleologu, J. Benesty, and S. Ciochină, "Linear system identification based on a Kronecker product decomposition," *IEEE/ACM Trans. Audio, Speech, Language Processing*, vol. 26, pp. 1793–1808, Oct. 2018.
- [9] 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 Trans. Audio, Speech, Language Processing*, vol. 27, pp. 903–918, May 2019.
- [10] H. He, J. Chen, J. Benesty, and Y. Yu, "Robust recursive least M-estimate adaptive filter for the identification of low-rank acoustic systems," in *Proc. IEEE ICASSP*, 2021, pp. 940–944.
- [11] H. He, J. Chen, J. Benesty, and Y. Yu, "A frequency-domain recursive least-squares adaptive filtering algorithm based on a Kronecker product decomposition," in *Proc. IEEE ICASSP*, 2023, 5 pages.
- [12] J. Benesty, I. Cohen, and J. Chen, *Array Processing–Kronecker Product Beamforming*. Cham, Switzerland: Springer-Verlag, 2019.
- [13] S. S. Bhattacharjee and N. V. George, "Fast and efficient acoustic feedback cancellation based on low rank approximation," *Signal Processing*, vol. 182, id. 107984, May 2021.
- [14] S. S. Bhattacharjee and N. V. George, "Nearest Kronecker product decomposition based linear-in-the-parameters nonlinear filters," *IEEE/ACM Trans. Audio, Speech, Language Processing*, vol. 29, pp. 2111–2122, May 2021.
- [15] X. Wang, J. Benesty, J. Chen, G. Huang, and I. Cohen, "Beamforming with cube microphone arrays via Kronecker product decompositions," *IEEE/ACM Trans. Audio, Speech, Language Processing*, vol. 29, pp. 1774–1784, May 2021.
- [16] G. Huang, J. Benesty, I. Cohen, and J. Chen, "Kronecker product multichannel linear filtering for adaptive weighted prediction error-based speech dereverberation," *IEEE/ACM Trans. Audio, Speech, Language Processing*, vol. 30, pp. 1277–1289, Mar. 2022.
- [17] S. Vadhana, S. K. Yadav, S. S. Bhattacharjee, and N. V. George, "An improved constrained LMS algorithm for fast adaptive beamforming based on a low rank approximation," *IEEE Trans. Circuits Systems II: Express Briefs*, vol. 69, pp. 3605–3609, Aug. 2022.
- [18] S. S. Bhattacharjee, V. Patel, and N. V. George, "Nonlinear spline adaptive filters based on a low rank approximation," *Signal Processing*, vol. 201, id. 108726, Dec. 2022.
- [19] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electron. Commun. Jpn.*, vol. 67-A, pp. 19–27, May 1984.
- [20] Y. V. Zakharov and F. Albu, "Coordinate descent iterations in fast affine projection algorithm," *IEEE Signal Processing Lett.*, vol. 12, pp. 353–356, May 2005.
- [21] C. Paleologu, J. Benesty, and S. Ciochină, "A variable step-size affine projection algorithm designed for acoustic echo cancellation," *IEEE Trans. Audio, Speech, Language Processing*, vol. 16, pp. 1466–1478, Nov. 2008.
- [22] J. Benesty, C. Paleologu, T. Gänslar, and S. Ciochină, *A Perspective on Stereophonic Acoustic Echo Cancellation*. Springer-Verlag, Berlin, Germany, 2011.
- [23] C. Stanciu, J. Benesty, C. Paleologu, T. Gänslar, and S. Ciochină, "A novel perspective on stereophonic acoustic echo cancellation," in *Proc. IEEE ICASSP*, 2012, pp. 25–28.
- [24] C. Stanciu, J. Benesty, C. Paleologu, T. Gänslar, and S. Ciochină, "A widely linear model for stereophonic acoustic echo cancellation," *Signal Processing*, vol. 93, pp. 511–516, Feb. 2013.
- [25] J. Benesty, C. Paleologu, and S. Ciochină, "Regularization of the RLS algorithm," *IEICE Trans. Fundamentals*, vol. E94-A, pp. 1628–1629, Aug. 2011.
- [26] C. Stanciu, C. Anghel and L. Stanciu, "Efficient FPGA implementation of the DCD-RLS algorithm for stereo acoustic echo cancellation," in *Proc. IEEE ISSCS*, 2015, pp. 1–4.

Membrii echipei proiectului,

Cristian-Lucian Stanciu

Radu-Andrei Otopoleanu

Nicolae-Cătălin Ristea