

Noi algoritmi adaptivi cu convergență variabilă

Contract nr. 7 / 05.08.2010, Cod TE_50

Algoritmi adaptivi de tip LMS cu pas de adaptare variabil

- Sinteza fazei unice / 2010 -

Un filtru adaptiv este definit ca un filtru capabil să-și modifice parametrii, în scopul optimizării unor caracteristici ale sale, pe baza unui algoritm recursiv. Scopul acestui sistem este ca ieșirea sa să “modeleze” un așa numit *semnal de referință* (sau semnal dorit). Acest lucru se realizează prin minimizarea *semnalului eroare* (pe baza a diverse modele matematice sau funcții cost), definit ca diferența între semnalul de referință și semnalul de la ieșirea filtrului. Prin urmare, filtrul adaptiv este constituit dintr-un filtru digital, cu parametrii reglabili pe baza unui *algoritm adaptiv*, concretizat printr-un bloc de calcul ce determină valorile coeficienților în raport cu un criteriu dat (minimizarea erorii), conducând la reglarea lor automată.

În cadrul oricărui filtru adaptiv blocul “cheie” este reprezentat de către algoritmul adaptiv. În literatura de specialitate există menționați numeroși algoritmi adaptivi pentru diverse structuri de filtre. Alegerea unei anumite variante poate fi făcută ținând cont de o serie de criterii de performanță, cum sunt: *viteza de convergență* (numărul de iterații necesare pentru a se ajunge la o soluție suficient de apropiată de cea optimă), *dezadaptarea* (măsura în care valoarea finală a erorii medii pătratice diferă față de eroarea medie pătratică dată de filtrul optim), *capacitatea de urmărire* (a variațiilor statistice ale semnalelor sau ale mediului), *complexitatea aritmetică* (numărul de operații aritmetice efectuate în fiecare iterație și capacitatea de memorie necesară) și *efecte ale preciziei finite* (erori de cuantizare, stabilitate numerică).

Algoritmii utilizați în momentul de față pentru filtrarea adaptivă pot fi grupați în două mari categorii [1]: 1) algoritmi bazați pe *minimizarea erorii medii pătratice*, cunoscuți sub denumirea de *algoritmi de tip LMS (least-mean-square)* și 2) algoritmi bazați pe *optimizarea în sensul celor mai mici pătrate*, cunoscuți drept *algoritmi de tip RLS (recursive least-squares)*. Algoritmii de tip LMS au o complexitate aritmetică redusă și o bună robustețe numerică, dar viteza lor de convergență este inferioară algoritmilor de tip RLS. Pe de altă parte, algoritmii de tip RLS au o complexitate aritmetică ridicată și unele probleme de instabilitate numerică.

Algoritmii de tip LMS sunt de obicei utilizați în majoritatea aplicațiilor de filtrare adaptivă, în special datorită simplității lor și a complexității aritmetice reduse. Performanțele acestor algoritmi depind de valoarea *pasului de adaptare (step-size)*, notat de obicei în literatură prin μ , în termenii compromisului viteză de convergență/capacitate de urmărire *versus* dezadaptare. O schemă simplă de implementare a unui pas de adaptare variabil ($VSS =$

variable step-size) are la bază utilizarea a două valori, după următoarea regulă: $\mu(n) = \mu_1$ în faza inițială a procesului adaptiv și $\mu(n) = \mu_2$ în faza de convergență, cu $\mu_1 \gg \mu_2$ [2] (prin n s-a notat indicele temporal). Principalul avantaj al acestui algoritm este simplitatea. Cu toate acestea, schema nu poate fi considerată o soluție practică din două motive: 1) trebuie identificat momentul de comutare între cele două valori ale pasului de adaptare (momentul de intrare în faza de convergență a algoritmului), lucru dificil de evaluat în special în aplicațiile de timp-real și 2) nu este adecvată în situația schimbărilor de mediu produse după intrarea în convergență, când algoritmul va reacționa lent datorită utilizării pasului de adaptare redus μ_2 . O altă categorie de algoritmi de tip VSS-LMS au la bază evaluarea energiei erorii algoritmului și compararea valorii acesteia cu un anumit prag [3]-[5] (“monitorizarea” stării de convergență), decizându-se apoi varierea pasului de adaptare pe baza unor relații recursive. Rezultatele obținute sunt mai bune comparativ cu metoda precedentă, dar sunt influențate de alegerea valorii pragului respectiv; acest lucru limitează aplicarea acestor metode în practică. O abordare mai recentă propune monitorizarea stării de convergență a algoritmului pe baza evaluării “deviației” coeficienților (*MSD = mean-square deviation*) și minimizării acesteia într-o manieră recursivă [6]. Soluția este interesantă, iar rezultatele obținute sunt mult mai bune decât în cazul metodelor precedente. Dezavantajul acestui algoritm este însă tot de natura “practică”. În formula pasului de adaptare apare o constantă (notată cu C în [6]), a cărei valoare depinde atât de puterea semnalului de intrare, cât și de puterea zgomotului de sistem (zgomotul care afectează semnalul de referință al filtrului adaptiv). Acești doi parametri pot fi dificil de estimat cu acuratețe, mai ales în situația în care avem de a face cu medii nestaționare. Algoritmul propus în [6] este foarte sensibil la valoarea acestei constante C , ceea ce constituie o limitare importantă în aplicațiile practice. O abordare oarecum similară, însă dezvoltată într-o manieră mai elegantă și mai practică, a fost propusă în [7]. Condiția care stă la baza acestui algoritm este de a “recupera” zgomotul de sistem în eroarea filtrului adaptiv; principial, acest lucru este echivalent cu minimizarea MSD. Algoritmul propus în [7] este însă mult mai robust decât cel propus în [6]; formula pasului de adaptare depinde doar de puterea zgomotului de sistem. Cu toate acestea, există o serie întregă de aplicații în care acest parametru este dificil de estimat, în special datorită nestaționarității zgomotului de sistem (de exemplu: compensarea ecoului, reducerea zgomotului de fond). Recent, o altă soluție interesantă bazată pe alegerea pasului de adaptare dintr-o tabelă cu valori predeterminate (pe baza unor analize de convergență) a fost propusă în [8]. Limitarea acestei metode este oarecum similară cu a celor propuse în [6] și [7], în sensul că în algoritmul ce stă la baza completării tabelii de căutare intervine ca parametru puterea zgomotului de sistem, cu limitările menționate anterior.

Algoritmul propus în [9] este o variantă a algoritmului din [7], dar nu mai necesită estimarea puterii zgomotului de sistem. Acest parametru este evaluat într-o formă “mascată”, pe baza puterii a două secvențe disponibile în cadrul oricărei aplicații de filtrare adaptivă: semnalul de referință și semnalul de la ieșirea filtrului adaptiv. În consecință, nu mai sunt necesare informații *a priori* legate de mediul în care operează filtrul adaptiv, ceea ce sporește gradul de aplicabilitate practică a metodei. În plus, s-a generalizat soluția în cazul algoritmilor bazați pe proiecții afine (*APA = affine projection algorithm*) [10], o categorie de

algoritmi derivați din familia LMS, dar care beneficiază de o viteză de convergență superioară. Cei doi algoritmi propuși au fost validați pe baza implementărilor pe FPGA [11], [12], certificând astfel comportarea lor în precizie finită și aplicabilitatea practică.

Pentru exemplificare, să presupunem că ne aflăm în contextul unei aplicații de identificare de sistem. Semnalul de la ieșirea sistemului necunoscut, $y(n)$, este obținut prin filtrarea semnalului de intrare $x(n)$ printr-un filtru cu răspuns finit la impuls, cu L coeficienți reali, ale căror valori sunt conținute în vectorul coloană \mathbf{h} . De asemenea, să presupunem că filtrul adaptiv va avea coeficienți reali, ale căror valori sunt conținute în vectorul $\hat{\mathbf{h}}(n)$. Schema de analiză este prezentată în Fig. 1. Una din aplicațiile importante ce au la bază această schema o reprezintă compensarea adaptivă ecoului.

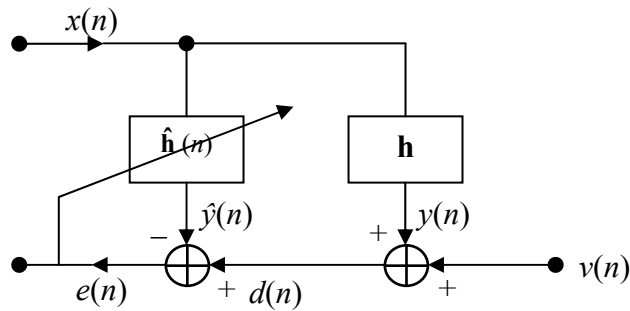


Fig. 1. Configurația de suprimare a interferențelor ce stă la baza aplicației de compensare a ecoului

Schema din Fig. 1 poate fi privită ca o combinație foarte interesantă între două configurații “clasice” de sisteme adaptive. În primul rând, poate fi considerată o schemă de *identificare de sistem*, în care scopul principal îl reprezintă identificarea sistemului necunoscut \mathbf{h} cu ajutorul filtrului adaptiv $\hat{\mathbf{h}}(n)$. În al doilea rând, schema din Fig. 1 reprezintă o configurație de *suprimare a interferențelor*, cu rolul reducerii semnalului perturbator $y(n)$ ce afectează semnalul util $v(n)$. Cele două cazuri sunt practic echivalente.

În scopul unei abordări unitare, vom defini semnalele de eroare *a-priori*, respectiv *a-posteriori* astfel:

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

$$\varepsilon(n) = d(n) - \mathbf{x}^T(n)\hat{\mathbf{h}}(n) \quad (2)$$

unde $\mathbf{x}(n) = [x(n) \ x(n-1) \ \dots \ x(n-L+1)]^T$, iar T reprezintă operatorul de transpunere.

Relația de reactualizare a coeficienților filtrului adaptiv în cazul algoritmilor de tip LMS este:

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mu(n)\mathbf{x}(n)e(n) \quad (3)$$

în care $\mu(n)$ reprezintă pasul algoritmului. Introducând expresia lui $\hat{\mathbf{h}}(n)$ din relația (3) în relația (2) și ținând cont de formula (1), rezultă o legătură între cele două tipuri de erori de forma

$$\varepsilon(n) = e(n)[1 - \mu(n)\mathbf{x}^T(n)\mathbf{x}(n)] \quad (4)$$

O modalitate de determinare a pasului de adaptare o constituie impunerea condiției de anulare a erorii a-posteriori, $\varepsilon(n) = 0$ (presupunând că $e(n) \neq 0$) [13]. În aceste condiții, din relația (4) rezultă că $\mu(n) = 1/[\mathbf{x}^T(n)\mathbf{x}(n)]$. În cazul algoritmului NLMS (*Normalized LMS*), pasul de adaptare este definit ca $\mu(n) = \mu/[\mathbf{x}^T(n)\mathbf{x}(n)]$, cu $0 < \mu < 2$. Prin urmare, valoarea $\mu =$

1 conduce aparent la performanțe optime. Acest lucru este valabil însă doar în situația în care semnalul $v(n)$ este absent. Din Fig. 1 se observă că

$$\varepsilon(n) = \mathbf{x}^T(n)[\mathbf{h} - \hat{\mathbf{h}}(n)] + v(n) \quad (5)$$

Atunci când $v(n) \neq 0$, condiția de anulare a erorii a-posteriori conduce la

$$\mathbf{x}^T(n)[\mathbf{h} - \hat{\mathbf{h}}(n)] = -v(n) \neq 0 \quad (6)$$

ceea ce va afecta estimatul filtrului adaptiv. Prin urmare, forțând anularea erorii a-posteriori $\varepsilon(n)$, se va produce o “scurgere” a semnalului $v(n)$ în coeficienții $\hat{\mathbf{h}}(n)$.

Ținând cont de aceste aspecte, o soluție mai viabilă de determinare a pasului de adaptare a algoritmului o reprezintă impunerea condiției $\varepsilon(n) = v(n)$. Astfel, se urmărește ca

$$E\{\varepsilon^2(n)\} = E\{v^2(n)\} \quad (7)$$

unde $E\{\cdot\}$ reprezintă operatorul de mediere statistică. Prelucrând relația (4) în acest sens (ridicare la pătrat și mediere statistică) se obține

$$E\{\varepsilon^2(n)\}[1 - \mu(n)LE\{x^2(n)\}]^2 = E\{v^2(n)\} \quad (8)$$

Rezolvând ecuația de gradul 2 precedentă se obține expresia pasului de adaptare

$$\mu(n) = \frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} \left[1 - \sqrt{\frac{E\{v^2(n)\}}{E\{\varepsilon^2(n)\}}} \right] \quad (9)$$

Estimarea mărimilor de forma $E\{\alpha^2(n)\}$ se poate realiza într-un mod recursiv astfel:

$$\hat{\sigma}_\alpha^2(n) = \lambda \hat{\sigma}_\alpha^2(n-1) + (1-\lambda)\alpha^2(n) \quad (10)$$

unde $\lambda = 1 - 1/(KL)$, cu $K > 1$. Prin urmare, expresia (9) devine

$$\mu(n) = \frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} \left[1 - \frac{\hat{\sigma}_v(n)}{\hat{\sigma}_e(n)} \right] \quad (11)$$

Acest algoritm cu pas variabil, denumit NPVSS-NLMS (*Nonparametric Variable Step Size NLMS*) a fost propus în [7]. Principala sa limitare constă în faptul că semnalul $v(n)$ este în general indisponibil în practică. În plus, în anumite aplicații (ex: compensarea ecoului) acest semnal poate fi nestaționar, conținând atât zgomotul de fond (ce poate varia în timp) cât și vocea vorbitorului de la capătul apropiat (*near-end*), ceea ce complică estimarea puterii sale. Pentru a depăși aceste limitări este de dorit găsirea unei soluții alternative care să evite estimarea directă a semnalului $v(n)$. Se știe că semnalul dorit $d(n)$ al filtrului adaptiv conține atât semnalul $v(n)$, cât și ieșirea sistemului necunoscut, $y(n)$. Cele două secvențe pot fi presupuse necorelate, astfel încât se poate afirma că

$$E\{d^2(n)\} = E\{y^2(n)\} + E\{v^2(n)\} \quad (12)$$

În plus, presupunând că filtrul adaptiv a converș într-o anumită măsură vom avea

$$E\{y^2(n)\} \approx E\{\hat{y}^2(n)\} \quad (13)$$

Prin urmare, din relațiile precedente rezultă că

$$E\{v^2(n)\} = E\{d^2(n)\} - E\{\hat{y}^2(n)\} \quad (14)$$

iar relația (11) devine

$$\mu(n) = \frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} \left[1 - \frac{\sqrt{\hat{\sigma}_d^2(n) - \hat{\sigma}_y^2(n)}}{\hat{\sigma}_e(n)} \right] \quad (15)$$

Expresia (15) este mult mai adecvată în practică deoarece depinde numai de secvențe ce sunt disponibile în mod direct în cadrul aplicației de filtrare adaptivă. Algoritmul VSS-NLMS rezultat a fost propus și analizat în [9].

O altă modalitate practică de a estima puterea semnalului $v(n)$ constă în aplicarea relației (10), pentru semnalul eroare $e(n)$, utilizând însă un parametru $\gamma > \lambda$ [14]:

$$\hat{\sigma}_v^2(n) = \gamma \hat{\sigma}_v^2(n-1) + (1-\gamma)e^2(n) \quad (16)$$

În continuare sunt prezentate rezultatele unor simulări efectuate în cadrul unei configurații de compensare a ecoului acustic [15]. Au fost comparate performanțele algoritmilor NLMS clasic, NPVSS-NLMS [7], VSS-NLMS-1 [bazat pe relația (15)], VSS-NLMS-2 [bazat pe relația (16)] și VSS-NLMS-id [“ideal” - în care se presupune că semnalul $v(n)$ este disponibil]. Sistemul “necunoscut” (calea de ecou acustic) are lungimea $L = 512$, egală cu cea a filtrului adaptiv. Semnalul de intrare $x(n)$ este o secvență vocală, frecvența de eșantionare fiind $F_s = 8$ kHz.

Într-un prim experiment, semnalul $v(n)$ constă doar în zgomotul de fond (cazul *single-talk*), cu $\text{SNR} = 20\text{dB}$. Rezultatele sunt prezentate în Fig. 2. Se observă că algoritmul NPVSS-NLMS (în care se presupune cunoscută puterea zgomotului de fond) este practic similar cu cazul “ideal”, iar algoritmul VSS-NLMS-1 obține performanțe apropiate. Viteza de convergență se reduce însă în cazul algoritmului VSS-NLMS-2.

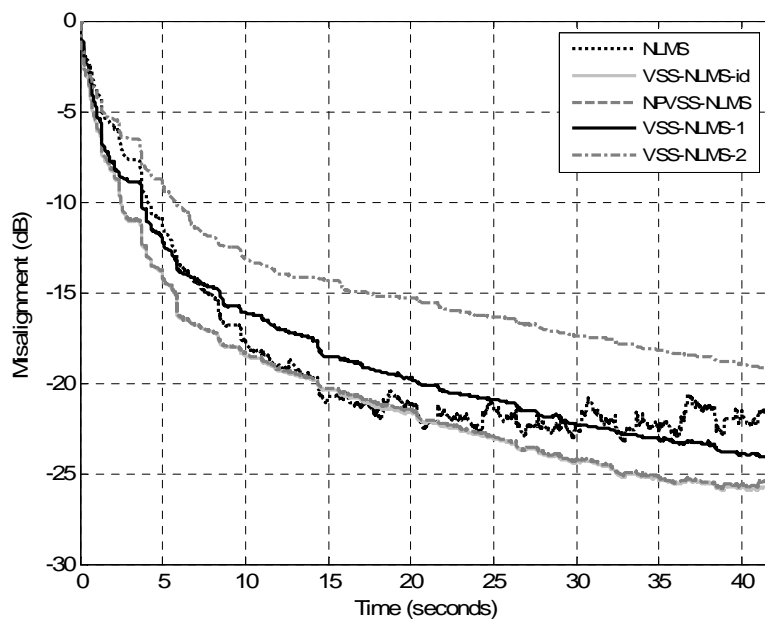


Fig. 2. Performanțele algoritmilor de tip VSS-NLMS în cazul *single-talk*.

În cel de al doilea experiment, pe lângă zgomotul de fond este prezentă și vocea vorbitorului de la capătul apropiat (cazul *double-talk*). Rezultatele sunt prezentate în Fig. 3, fiind evidentă superioritatea algoritmului VSS-NLMS-1.

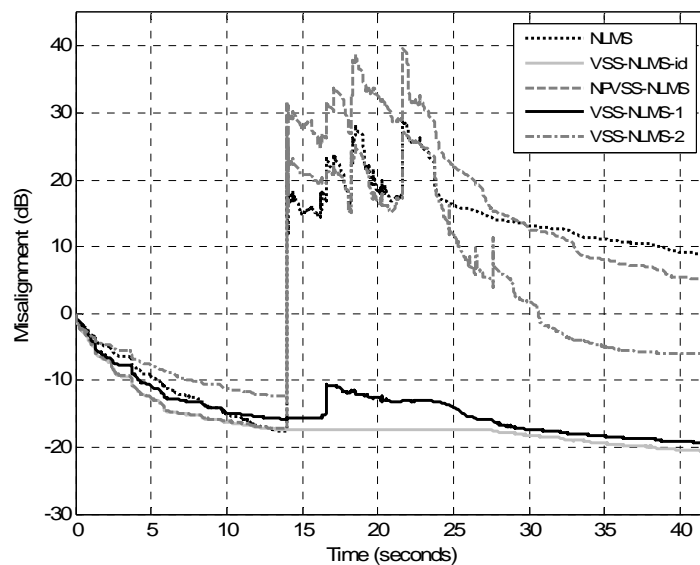


Fig. 3. Performanțele algoritmilor de tip VSS-NLMS în cazul *double-talk*.

Ideea de bază a acestei categorii de algoritmi VSS-NLMS poate fi extinsă și asupra algoritmilor de tip “proportionate” [16], [17], care sunt mult mai adecvați situației în care sistemul ce urmează a fi identificat este de tip *sparse*, așa cum este cazul căilor de ecou.

Bibliografie

- [1] S. Haykin, *Adaptive Filter Theory*, Fourth Edition, Prentice Hall International, Inc. Upper Saddle River, N.J., 2002.
- [2] V. Weerackody, A. Kassam, and K. Laker, “A simple hard-limited adaptive algorithm for blind equalization”, *IEEE Trans. Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 39, no. 7, pp. 482-487, July 1992.
- [3] V. J. Mathews and Z. Xie, “A stochastic gradient adaptive filter with gradient adaptive step-size”, *IEEE Trans. Signal Processing*, vol. 41, no. 6, pp. 2075-2087, June 1993.
- [4] T. Aboulnasr and K. Mayyas, “A robust variable step-size LMS-type algorithm: analysis and simulations”, *IEEE Trans. Signal Processing*, vol. 45, no. 3, pp. 631-639, Mar. 1997.
- [5] A. I. Sulyman and A. Zerguine, “Convergence and steady-state analysis of a variable step-size NLMS algorithm”, *Signal Processing, EURASIP*, vol. 83, issue 6, pp. 1255-1273, June 2003.
- [6] H.-C. Shin, A. H. Sayed, and W.-J. Song, “Variable step-size NLMS and affine projection algorithms”, *IEEE Signal Processing Letters*, vol. 11, no. 2, pp. 132-135, Feb. 2004.
- [7] J. Benesty, H. Rey, L. Rey Vega, and S. Tressens, “A nonparametric VSS NLMS algorithm”, *IEEE Signal Processing Letters*, vol. 13, no. 10, pp. 581-584, Oct. 2006.
- [8] P. Park, M. Chang, and N. Kong, “Scheduled-stepsize NLMS algorithm”, *IEEE Signal Processing Letters*, vol. 16, no. 12, pp. 1055-1058, Dec. 2009.

- [9] C. Paleologu, S. Ciochină, and J. Benesty, “Variable step-size NLMS algorithm for under-modeling acoustic echo cancellation”, *IEEE Signal Processing Letters*, vol. 15, pp. 5-8, 2008.
- [10] C. Paleologu, J. Benesty, and S. Ciochină, “A variable step-size affine projection algorithm designed for acoustic echo cancellation”, *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1466-1478, Nov. 2008.
- [11] C. Anghel, C. Paleologu, J. Benesty, and S. Ciochină, “FPGA implementation of an acoustic echo canceller using a VSS-NLMS algorithm”, in *Proc. IEEE ISSCS*, Iași, Romania, 2009, pp. 369-372.
- [12] C. Anghel, C. Paleologu, J. Benesty, and S. Ciochină, “FPGA implementation of a variable step-size affine projection algorithm for acoustic echo cancellation”, in *Proc. EUSIPCO*, Aalborg, Denmark, 2010, pp. 532-536.
- [13] D. R. Morgan, S. G. Kratzer, “On a class of computationally efficient, rapidly converging, generalized NLMS algorithms”, *IEEE Signal Processing Letters*, vol. 3, no. 8, pp. 245–247, Aug. 1996.
- [14] C. Paleologu, S. Ciochină, “A class of variable step-size NLMS and affine projection algorithms suitable for echo cancellation”, in *Proc. ISETC 2008*, Timișoara, Sept. 2008, pp. 118-123.
- [15] J. Benesty, T. Gaensler, D. R. Morgan, M. M. Sondhi, S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*, Springer-Verlag, Berlin, Germany, 2001.
- [16] C. Paleologu, J. Benesty, S. Ciochină, “*Sparse Adaptive Filters for Echo Cancellation*”, Morgan & Claypool Publishers, 2010.
- [17] J. Benesty, C. Paleologu, S. Ciochină, “Proportionate Adaptive Filters from a Basis Pursuit Perspective”, *IEEE Signal Processing Letters*, vol. 17, no. 12, pp. 985-988, Dec. 2010.

DIRECTOR PROIECT,
Conf. dr. ing. Constantin PALEOLOGU