

Contribuții la teoria și implementarea metodelor de reducere adaptivă a zgomotului de fond în sistemele multi-canal de comunicații de voce

Contract nr. 331/01.10.2007, Cod ID_777

Dezvoltarea unor algoritmi adaptivi pentru sisteme ANC multi-canal și analiza sistemelor de microfoane cu lobi programabili.

- Sinteza fazei unice / 2009 -

În cadrul sistemelor de comunicații cu “mâini-libere” (“*hands-free*”), utilizate în numeroase aplicații precum telefonie mobilă și teleconferințe, sunt necesare reducerea unor perturbații aferente mediului acustic, cum ar fi zgomotul de fond și ecoul acustic. Schemele ce stau la baza acestor aplicații au ca bloc principal filtrul adaptiv.

Recent, o categorie de algoritmi adaptivi rapizi bazați pe proiecții afine (*FAP – fast affine projection*) au fost analizați în contextul acestor sisteme [1], [2]. Acești algoritmi oferă o soluție de compromis avantajoasă între viteza de convergență și complexitatea aritmetică implicată. Chiar dacă din punct de vedere al vitezei de convergență sunt inferiori algoritmilor bazați pe optimizarea în sensul celor mai mici pătrate (*LS – least squares*), algoritmi FAP compensează printr-o complexitate aritmetică mai redusă comparativ cu algoritmi de tip LS. Există însă o serie de probleme de natură numerică ale algoritmilor FAP, ce le limitează într-o anumită măsură utilizarea la scară largă în cadrul aplicațiilor din domeniul acustic. Structura matematică a acestor algoritmi implică rezolvarea unui sistem de ecuații liniare. Numărul ecuațiilor este dat de ordinul de proiecție afină al algoritmului. Alegerea unui ordin mare de proiecție afină conduce la o soluție trivială din punct de vedere al complexității numerice. Rezolvarea acestui set de ecuații liniare din cadrul algoritmului FAP presupune însă utilizarea unui algoritm de tip LS rapid (*fast LS*), ce poate conduce la probleme de instabilitate numerică specifice, în special în cazul multi-canal. În consecință, au fost investigate o serie de soluții alternative. Metoda Gauss-Seidel constituie una din tehnicile utilizate în acest context [3]. Ulterior, în scopul reducerii complexității aritmetice, se utilizează proiecțiile pseudo-afine (*PAP – pseudo-affine projection*), rezultând o nouă familie de algoritmi, denumiți GS-PAP (*Gauss-Seidel pseudo-affine projection*) [4], [5]. De asemenea, pentru o reducere suplimentară a complexității aritmetice au fost utilizate metode bazate pe iterații DCD [6].

În cadrul algoritmilor bazați pe proiecții afine, pasul de adaptare al algoritmului reprezintă un parametru cheie, deoarece stă la baza realizării compromisului între criteriile de performanță (viteză de convergență și capacitate de urmărire *versus* dezadaptare și robustețe). Algoritmi de adaptare cu pas variabil (*VSS – variable step-size*) urmăresc rezolvarea acestei probleme. Soluția propusă în cadrul acestui proiect de cercetare are ca rezultat o formulă *non-parametrică* a pasului de adaptare, ceea ce constituie o facilitate importantă pentru

aplicabilitatea sa practică. Algoritmii propuși, de tip FAP/PAP ce au la bază metoda de reducere a complexității GS și pas de adaptare variabil (VSS) [7], sunt prezentați succint în continuare. Algoritmii similari, bazați de metoda DCD au fost prezentați în [8].

În dezvoltarea algoritmilor bazați pe metoda GS sunt utilizate următoarele notații:

- L este lungimea filtrului adaptive;
- K este ordinal de proiecție al APA;
- δ este factorul de regularizare;
- λ este un factor de ponderare;
- \mathbf{I}_K este matricea identitate de dimensiune $K \times K$ identity matrix;
- $\mu(n)$ este pasul de adaptare variabil al algoritmului;
- $\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-L+1)]^T$ este vectorul semnalului de intrare la momentul de timp n (operatorul T semnifică operația de transpunere);
- $\mathbf{R}(n)$ este matricea de autocorelație a semnalului de intrare (de dimensiune $K \times K$);
- $\xi(n) = [x(n), x(n-1), \dots, x(n-K+1)]^T$ este un vector de dimensiune $K \times 1$;
- $\mathbf{u}(n) = [u(n), \dots, u(n-L+1)]^T$ este vectorul aproximat;
- $\bar{\mathbf{u}}(n)$ conține primele $L-1$ elemente ale vectorului $\mathbf{u}(n)$;
- $\mathbf{b} = [1 \quad \mathbf{0}_{(K-1) \times 1}^T]^T$;
- $\hat{\mathbf{h}}(n) = [\hat{h}_0(n), \dots, \hat{h}_{L-1}(n)]^T$ este vectorul coeficienților filtrului adaptive;
- $\mathbf{w}(n) = [w_0(n), \dots, w_{L-1}(n)]^T$ este vectorul coeficienților filtrului auxiliar [9];
- $\mathbf{r}(n)$ este vectorul de autocorelației (de dimensiune $K \times 1$);
- $\tilde{\mathbf{r}}(n)$ conține ultimele $K-1$ elemente ale vectorului $\mathbf{r}(n)$;
- $\mathbf{p}(n)$ este vectorul soluție al problemei LP (de dimensiune $K \times 1$) [9];
- $p_i(n)$ (cu $i = 1, \dots, K$) reprezintă elementul i din vectorul $\mathbf{p}(n)$;
- $\mathbf{e}(n)$ este un vector de dimensiune $K \times 1$;
- $e_K(n)$ reprezintă ultimul element al vectorului $\mathbf{e}(n)$;
- $\bar{\mathbf{e}}(n)$ conține primele $K-1$ elemente ale vectorului $\mathbf{e}(n)$.

Matricea $\mathbf{R}(n)$ este reactualizată prin înlocuirea primul rand și al primei coloane cu elementele vectorului $\mathbf{r}(n)$, iar cealaltă parte a matricei [de dimensiune $(K-1) \times (K-1)$] este înlocuită cu matricea de dimensiune $(K-1) \times (K-1)$ rezultată din primele $K-1$ linii și coloane ale matricei $\mathbf{R}(n-1)$ [5].

Pasul de adaptare variabil pentru algoritmii VSS-GS-PAP și VSS-GSFAP este calculat pe baza următoarelor relații [10], [11]:

$$\hat{\sigma}_s^2(n) = \lambda \hat{\sigma}_s^2(n-1) + (1-\lambda) [d^2(n) - \hat{y}^2(n)] \quad (1)$$

$$\hat{\sigma}_e^2(n) = \lambda \hat{\sigma}_e^2(n-1) + (1-\lambda)e^2(n) \quad (2)$$

$$\mu(n) = \left| 1 - \frac{\hat{\sigma}_s^2(n)}{\varepsilon + \hat{\sigma}_e^2(n)} \right| \quad (3)$$

unde $\hat{\sigma}_e^2(0) = 0$, $\hat{\sigma}_s^2(0) = 0$, ε este o constantă pozitivă de valoare foarte mică, iar factorul de ponderare λ este calculat ca în [12]. Relațiile care definesc algoritmul VSS-GS-PAP propus sunt următoarele:

Inițializare

$$\begin{aligned} \mathbf{x}(0) &= \mathbf{0}_{L \times 1}, \mathbf{R}(0) = \delta \mathbf{I}_K, \mathbf{u}(0) = \mathbf{0}_{L \times 1}, \hat{\mathbf{h}}(0) = \mathbf{0}_{L \times 1}, \\ \mathbf{p}(0) &= \mathbf{0}_{K \times 1}, \xi(0) = \mathbf{0}_{K \times 1}, \mathbf{r}(0) = \left[\delta, \mathbf{0}_{(K-1) \times 1}^T \right]^T \end{aligned} \quad (4)$$

For $n = 1, 2, \dots$

$$\mathbf{r}(n) = \mathbf{r}(n-1) + x(n)\xi(n) - x(n-L)\xi(n-L) \quad (5)$$

reactualizează $\mathbf{R}(n)$ utilizând $\mathbf{r}(n)$ (6)

rezolvă $\mathbf{R}(n)\mathbf{p}(n) = \mathbf{b}$ (utilizând o iterație Gauss-Seidel) (7)

$$\mathbf{u}(n) = \left[\xi^T(n)\mathbf{p}(n)/p_1(n) \quad \bar{\mathbf{u}}^T(n-1) \right]^T \quad (8)$$

$$\hat{y}(n) = \mathbf{x}^T(n)\hat{\mathbf{h}}(n-1) \quad (9)$$

$$e(n) = d(n) - \hat{y}(n) \quad (10)$$

calculează $\mu(n)$ conform (1)–(3)

$$\bar{e}(n) = \frac{\mu(n)e(n)}{\delta + \mathbf{u}^T(n)\mathbf{x}(n)} \quad (11)$$

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mathbf{u}(n)\bar{e}(n) \quad (12)$$

Algoritmul VSS-GS-PAP algorithm necesită $2L + K^2 + 3K + 10$ multiplicări, 4 împărțiri și o operație de extragere a radicalului. Mai multe detalii despre algoritmi de tip GS-PAP pot fi găsite în [13].

Algoritmul VSS-GSFAP propus este definit de următoarele relații:

Inițializare

$$\begin{aligned} \mathbf{x}(0) &= \mathbf{0}_{L \times 1}, \mathbf{R}(0) = \delta \mathbf{I}_K, \mathbf{e}(0) = \mathbf{0}_{K \times 1}, \mathbf{w}(0) = \mathbf{0}_{L \times 1}, \\ \mathbf{p}(0) &= \mathbf{0}_{K \times 1}, \xi(0) = \mathbf{0}_{K \times 1}, \mathbf{r}(0) = \left[\delta, \mathbf{0}_{(K-1) \times 1}^T \right]^T, \mu(0) = 1 \end{aligned} \quad (13)$$

For $n = 1, 2, \dots$

$$\mathbf{r}(n) = \mathbf{r}(n-1) + x(n)\xi(n) - x(n-L)\xi(n-L)$$

reactualizează $\mathbf{R}(n)$ utilizând $\mathbf{r}(n)$

rezolvă $\mathbf{R}(n)\mathbf{p}(n) = \mathbf{b}$ (utilizând o iterație Gauss-Seidel)

$$\hat{y}(n) = \mathbf{x}^T(n)\mathbf{w}(n-1) + \mu(n-1)\tilde{\mathbf{r}}(n)\bar{\mathbf{e}}(n-1) \quad (14)$$

$$e(n) = d(n) - \hat{y}(n)$$

$$\mathbf{e}(n) = \begin{bmatrix} 0 & \bar{\mathbf{e}}^T(n-1) \end{bmatrix}^T + e(n)\mathbf{p}(n) \quad (15)$$

calculează $\mu(n)$ conform (1)–(3)

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu(n)\mathbf{x}(n-K+1)e_K(n) \quad (16)$$

Algoritmul VSS-GSFAP necesită $2L + K^2 + 4K + 10$ multiplicări, 2 împărțiri și o operație de extragere a radicalului. Mai multe detalii despre algoritmi de tip GS-FAP pot fi găsite în [9].

Relațiile necesare pentru calculul pasului de adaptare [(1)–(3)] necesită 9 multiplicări, 6 adunări, o operație de extragere a radicalului și o împărțire. Prin urmare, complexitatea aritmetică a algoritmilor propuși (VSS-GSFAP și VSS-GS-PAP) este comparativă cu cea a algoritmilor cu pas de adaptare fix (GS-FAP/PAP).

Pentru comparație s-a ales algoritmul VSS-PAP propus în [14], care utilizează o strategie diferită de evaluare a pasului, pe baza minimizării MSD (*mean-square deviation*). Complexitatea algoritmului VSS-PAP este de $2L + K^2 + 5K + 12$ multiplicări și împărțiri. În Fig. 1–4 sunt prezentate rezultatele simulărilor efectuate într-un scenariu de compensare a ecoului acustic. Acestea indică superioritatea algoritmilor propuși, VSS-GSFAP și VSS-GS-PAP.

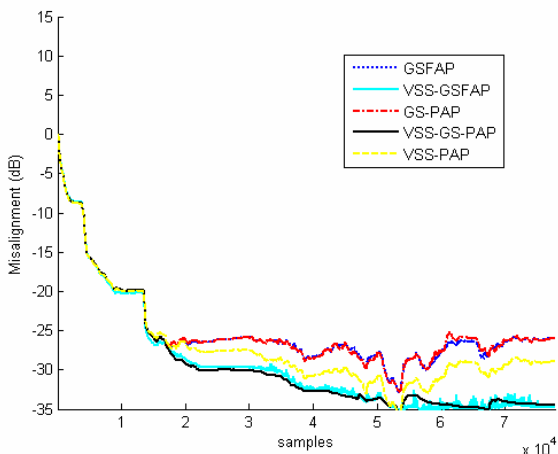


Fig. 1. Dezalinierea (misalignment) pentru algoritmi GSFAP, GS-PAP, VSS-GSFAP, VSS-PAP și VSS-GS-PAP. Cazul single-talk, $L = 512$, $K = 4$, $\text{SNR} = 30\text{dB}$.

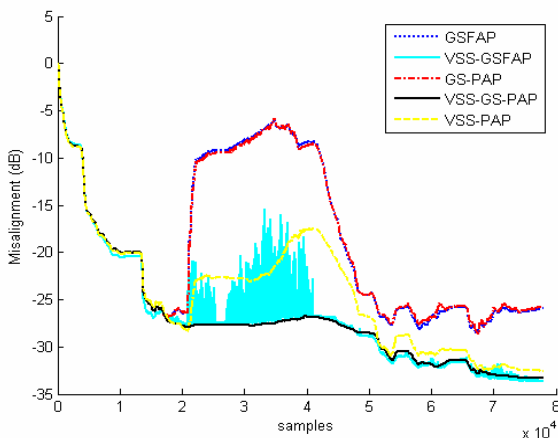


Fig. 2. Dezalinierea (misalignment) pentru algoritmi GSFAP, GS-PAP, VSS-GSFAP, VSS-PAP și VSS-GS-PAP. Cazul single-talk, $L = 512$, $K = 4$, SNR variabil (30dB-10dB).

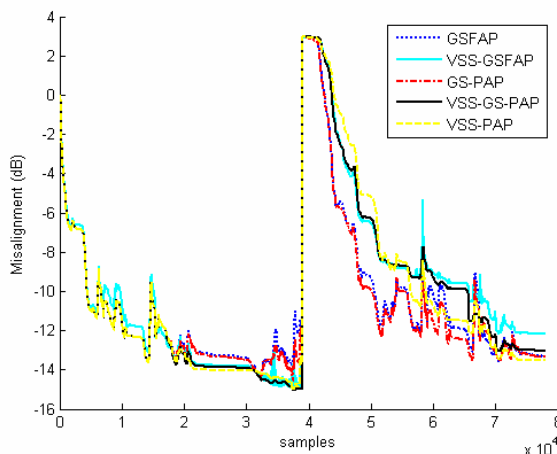


Fig. 3. Dezalinierea (misalignment) pentru algoritmi GSFAP, GS-PAP, VSS-GSFAP, VSS-PAP și VSS-GS-PAP. Cazul single-talk, $L = 512$ (cale de ecou 1024 - submodelare), $K = 4$, $\text{SNR} = 30\text{dB}$, schimbarea căii de ecou.

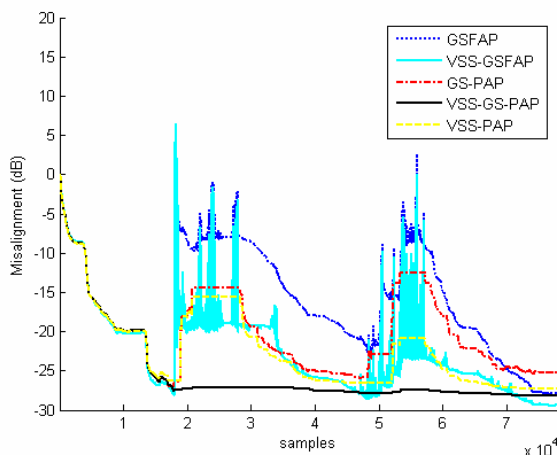


Fig. 4. Dezalinierea (misalignment) pentru algoritmi GSFAP, GS-PAP, VSS-GSFAP, VSS-PAP și VSS-GS-PAP. Cazul double-talk (cu Geigel DTD), $L = 512$, $K = 4$, $\text{SNR} = 30\text{dB}$.

Performanțele globale ale oricărei structuri ANC multi-canal sunt puternic influențate de performanțele sistemului de microfoane utilizate [15]. Acesta constă într-un set de microfoane (senzori) distribuite într-un mod spațial, astfel încât să se asigure caracteristici de directivitate optime. Una din cele mai importante funcții ale sistemului de microfoane este să extragă semnalul de interes dintr-o sumă de alte semnale. Metoda tipică este să se formeze un lob al caracteristicii de directivitate și să se îndrepte pe direcția dorită. Ca rezultat, semnalele de pe direcția favorizată (“privită”) sunt amplificate, iar celelalte atenuate.

În cazul unui sistem de microfoane, configurarea lobilor se face prin prelucrarea semnalelor de la ieșirile microfoanelor. O serie de algoritmi au fost dezvoltați în acest sens, cel mai simplu fiind configuratorul cu întârziere și adunare, ce a fost dezvoltat inițial pentru aplicații din domeniul subacvatic și pentru sistemele de antene radar. Ideea de bază constă în introducerea unei întârzieri adecvate a semnalelor microfoanelor, astfel încât componentele semnalului dorit să fie sincronizate la nivelul întregului sistem de senzori. Aceste semnale întârziate sunt apoi ponderate și adunate. Deoarece componentele semnalului dorit sunt sincronizate, acest semnal va fi amplificat, în vreme ce celelalte surse de semnal sunt atenuate (sau chiar eliminate). De obicei, coeficienții de ponderare se determină într-un mod adaptiv, astfel încât avem de a face cu un configurator adaptiv [16]. Metoda precedentă a fost dezvoltată pentru semnale de bandă îngustă. Deși ea constituie referința oricărui sistem de acest tip, această tehnică nu este foarte utilă în aplicații din domeniul acustic, deoarece vocea este de obicei un semnal de bandă largă. Prin urmare, caracteristica de directivitate obținută prin metoda cu întârziere și adunare nu va fi aceeași pentru întreaga bandă de frecvențe. Pentru evitarea acestei probleme se utilizează o arie formată din mai multe sub-sisteme de microfoane [17]. Fiecare sub-sistem este proiectat să opereze la o singură frecvență. Cu toate acestea, metoda respectivă nu este foarte fiabilă deoarece presupune utilizarea unui număr mare de microfoane, distribuite într-o arie foarte largă. O altă soluție constă în efectuarea unei descompunerii în benzi și utilizarea unui sistem independent de bandă îngustă pentru fiecare bandă de frecvențe. Prin urmare, sistemul de bandă largă va fi echivalent cu aplicarea unei filtrări de tip FIR (*finite impulse response*) semnalelor provenite de la fiecare microfon și adunarea ulterioară a acestor semnale filtrate, rezultând algoritmul de filtrare și sumare [18]-[20]. Această tehnică oferă o mai mare flexibilitate decât metoda cu întârziere și adunare. De asemenea, coeficienții de ponderare din cadrul algoritmului de filtrare și sumare pot fi determinați într-o manieră adaptivă.

Preocupări de dată recentă [21]-[23] încearcă abordarea problematicii sistemelor de microfoane pe baza similitudinilor cu sistemele de comunicații cu intrări multiple și ieșiri multiple (*MIMO – multiple inputs multiple outputs*). Astfel, algoritmi și metode utilizate în cadrul sistemelor MIMO pot fi extinse și aplicate în sistemele de microfoane cu lobi programabili. Problema majoră rămâne obținerea unor caracteristici optime de directivitate. În acest context, este esențială stabilirea unei legături cu teorema de inversare MINT (*multiple-input/output inverse theorem*), dezvoltată inițial pentru aplicații din domeniul acusticii (obținerea filtrului invers corespunzător acusticii unei incinte)

și filtrarea LCMV (*linearly constrained minimum variance*). Aceste aspecte permit abordarea problemei ținând cont de caracteristicile canalului, fiind posibilă dezvoltarea unor algoritmi specifici de estimare. În cadrul acestor scheme, blocurile de filtrare adaptivă ocupă de asemenea o poziție cheie, astfel încât se pot aplica algoritmi adaptivi dezvoltați pentru aplicațiile ANC multi-canal.

Bibliografie

- [1] M. Bouchard, “Multichannel affine and fast affine projection algorithms for active noise control and acoustic equalization systems”, *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 11, no. 1, pp. 54-60, Jan. 2003.
- [2] M. Bouchard, F. Albu, “The Gauss-Seidel fast affine projection algorithm for multichannel active noise control and sound reproduction systems”, *Int. Journal Adapt. Control Signal Process., Special Issue Adaptive Control Sound and Vibration*, vol. 19, no. 2-3, pp. 107-123, Mar.-Apr. 2005.
- [3] R. Barret et al., *Templates for the Solutions of Linear Systems: Building Blocks for Iterative Models*, 2nd ed., SIAM, Philadelphia, 1994.
- [4] F. Albu, M. Bouchard, “The Gauss-Seidel pseudo-affine projection algorithm and its application for echo cancellation”, *Proc. Asilomar*, Pacific Grove, CA, vol. 2, pp. 1303-1306, Nov. 2003.
- [5] F. Albu, M. Bouchard, Y.V. Zakharov, “Pseudo-affine projection algorithms for multichannel active noise control”, *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1044-1052, Mar. 2007.
- [6] Y. V. Zakharov, T. C. Tozer, “Multiplication-free iterative algorithm for LS problem”, *Electronics Letters*, vol. 40, no. 9, pp. 567-569, April 2004.
- [7] F. Albu, C. Paleologu, J. Benesty, “Gauss-Seidel based Variable Step-Size Affine Projection Algorithms for Acoustic Echo Cancellation”, *Proc. European Signal Processing Conference (EUSIPCO)*, pp. 2007-2011, Glasgow, Scoția, Aug. 2009.
- [8] F. Albu, C. Paleologu, J. Benesty, Y. V. Zakharov, “Variable Step Size Dichotomous Coordinate Descent Affine Projection Algorithm”, *Proc. IEEE EUROCON*, pp. 1366-1371, St. Petersburg, Rusia, 2009.
- [9] F. Albu, J. Kadlec, N. Coleman, and A. Fagan, “The Gauss-Seidel fast affine projection algorithm,” in *Proc. IEEE SIPS*, 2002, pp. 109–114.
- [10] C. Paleologu, J. Benesty, S. Ciochină, “Robust Variable Step-Size Affine Projection Algorithm Suitable for Acoustic Echo Cancellation”, *Proc. European Signal Processing Conference (EUSIPCO)*, pp. 173-177, Lausanne, Elveția, Aug. 2008.
- [11] C. Paleologu, J. Benesty, 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.
- [12] J. Benesty, H. Rey, L. Rey Vega, and S. Tressens, “A nonparametric VSS NLMS algorithm,” *IEEE Signal Processing Letters*, vol. 13, pp. 581–584, Oct. 2006.
- [13] F. Albu and H.K. Kwan, “Combined echo and noise cancellation based on Gauss-Seidel pseudo affine projection algorithm,” in *Proc. IEEE ISCAS*, 2004, pp. 505–508.

- [14] L. Lee, Y.-C. Park, D.-H. Youn, "Robust pseudo affine projection algorithm with variable step-size," *Electronics Letters*, vol. 44, pp. 250–252, Jan. 2008.
- [15] M. Branstein, D.B. Wards (ed.), *Microphone Arrays: Signal Processing Techniques and Applications*, Springer, Germany, 2001.
- [16] H. Cox, R.M. Zeskind, M.M. Owen, "Robust adaptive beamforming," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 10, pp. 1365-1376, Oct. 1987.
- [17] W. Kellerman, "A self-steering digital microphone array," *Proc. IEEE ICASSP*, vol. 5, pp. 3581-3584, 1991.
- [18] C. Marro, Y. Mahieux, K.U. Simmer, "Analysis of noise reduction and dereverberation techniques based on microphone arrays with postfiltering," *IEEE Trans. Speech Audio Process*, vol. 6, no. 5, pp. 240-259, May 1998.
- [19] I. Cohen, "Analysis of two-channel generalized sidelobe canceller (GSC) with post-filtering," *IEEE Trans. Speech Audio Process.*, vol. 11, pp. 684-699, Nov. 2003.
- [20] J. Benesty, S. Makino, J.E. Chen, *Speech Enhancement*, Springer, Germany, 2005.
- [21] Y. Huang, J. Benesty, J. Chen, *Acoustic MIMO Signal Processing*, Springer, Boston, 2006.
- [22] J. Benesty, J. Chen, Y.A. Huang, J. Dmochowski, "On microphone-array beamforming from a MIMO acoustic signal processing perspective," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1053-1065, Mar. 2007.
- [23] J. Chen, J. Benesty, Y. Huang, "An acoustic MIMO framework for analysing microphone-array beamforming," *Proc. IEEE ICASSP*, pp. I-25–I-28, 2007.

**DIRECTOR PROIECT,
Conf. dr. ing. Constantin PALEOLOGU**