# TEHNICI DE IDENTIFICARE A ZONELOR VOCALICE ÎN SECVENȚE ROSTITE ÎN LIMBA ROMÂNĂ

M. ZBANCIOC [1,2], H.N. TEODORESCU [1,2], M. FERARU [1]

[1]*Institutul de Informatică Teoretică, Academia Română – Filiala Iași, România*
[2] *Universitatea Tehnică „Gheorghe Asachi", Facultatea de Electronică, Telecomunicații și Tehnologia Informației, Iași – România*

*{hteodor, zmarius}@etti.tuiasi.ro*

## Rezumat

Se prezintă un set de tehnici de segmentare utilizate pentru identificarea zonelor vocalice. Segmentarea este folosită ulterior de instrumentele de extragere a frecvenței fundamentale $F_0$ și a valorilor formanților $F_1$,...,$F_4$. Comparăm precizia segmentării instrumentului propus cu cea a utilitarului Pratt, folosind fișiere adnotate cu mare precizie. Pentru reducerea timpului de rulare s-a optimizat calcul funcției de autocorelație, prin aplicarea unor algoritmi recurenți.

## *1. Introducere*

Segmentarea automată a semnalelor vocale, recunoașterea automată a vorbirii, a limbii de proveniență, identificarea vorbitorului sunt domenii de cercetare cu vechime de câteva decenii, dar încă de mare actualitate. Faza de segmentare este importantă deoarece erorile acesteia afectează în mod direct performanțele extractorului de informații prozodice. Deși în literatura de specialitate se găsesc numeroase articole ce descriu diverse tehnici de segmentare automată (Rabiner & Schafer, 1978), (Calliope, 1989), (Rowde, 1991), problema segmentării nu este complet rezolvată, datorită cvasi-periodicității semnalului vocalic și a gradului mare de variabilitate a caracteristicilor fonemelor de la o limbă la alta.

În (Vidal & Marzal, 1990) se face o trecere în revistă asupra tehnicilor de segmentare insistând asupra metodelor fără constrângeri în ceea ce privește variația contururilor spectrale (SVF), metode ce folosesc o segmentare multi-nivel și o descompunere temporară pentru găsirea limitelor segmentelor. Tehnicile de segmentare combinate cu metodele de recunoaștere automată folosesc HMMs (Hidden Markov Models), parametri acustici, cum ar fi coeficienții MFCC (Mel Frequency Cepstral Coefficients), LPC (Linear Predictive Coding), tehnici de aliniere dinamică în timp (DTW engl. Dynamic Time Warping), etc. (Rabiner & Juang, 1993), (Esposito & Aversano, 2005).

(Juneja & Espy-Wilson, 2002) folosesc HMMs combinate cu SVMs (Support Vector Machines) pentru a detecta vocalele, consoanele finale, consoanele fricative și sonante, respectiv zonele de pauză, folosind 39 de parametri extrași din cepstru. (Matousek et al., 2003) folosește HMMs cu coeficienți spectrali MFCCs raportând procentaje foarte bune de 96% în acuratețea segmentării pe un corpus de date lingvistice pentru limba cehă. (Salam et al., 2009) propune o fuziune a două metode de segmentare a vorbirii, anume metode statistice bazate pe un algoritm de divergență și metode conecționiste de învățare adaptivă MLP (Multi-Layer Perceptron). (Sarkar & Sreenivas, 2005) utilizează

o metodă bazată pe nivelul ALCR (Average Level Crossing Rate) pentru a detecta schimbările temporare semnificative în semnal. Sunt utilizate valori adaptive în funcţie de SNR (Signal-to-noise ratio) şi se compară performanţa de segmentare automată cu fişiere segmentate fonetic manual. Colectivul nostru de cercetare a făcut comparaţii între fişierele de sunet sintetizate şi fişierele de voce naturală (Teodorescu et al., 2009), folosind intervalele de timp specificate prin fişiere de adnotare, pentru a observa diferenţele care apar între acestea la nivelul parametrilor extraşi (durate, valori formanţi, variaţie formanţi). Corpus-ul „Sunetele Limbii Române SRoL" conţine vocale susţinute, fraze pronunţate cu diverse stări emoţionale, sunete gnatosonice, adnotări, instrumente de analiză a semnalului vocalic, care sunt disponibile on-line (Teodorescu et al., 2005).

Pentru determinarea frecvenţei fundamentale ($F_0$) există două tipuri de metode: metode indirecte şi metode directe. Metodele directe de detecţie a lui $F_0$ sunt: metoda impedanţmetrică (electroglotograma), metoda cinematografică (stroboscopică), metoda extracţiei $F_0$ pe baza formei de undă. Metodele indirecte implementate de către noi sunt: metode de analiză în domeniul timp (autocorelaţia, AMDF – Average Magnitude Difference Function), metode de analiză spectrală (metoda cepstrală, HPS – Harmonic Product Spectrum). Nici metodele directe nu pot fi considerate metode absolute, deoarece elementele elastice precum corzile vocale prezintă o vibraţie amortizată.

Pentru validarea metodelor indirecte de extracţie a frecvenţei fundamentale este necesară o comparaţie între rezultatele obţinute în cazul aplicării acestora şi cele obţinute în cazul utilizării metodelor directe. În acest scop am realizat adnotări de mare precizie folosind metoda bazată pe forma de undă. Lucrarea prezintă perfecţionări ale instrumentului expus sumar în (Teodorescu et al., 2007), (Zbancioc, 2006). Scopul cercetării este implementarea unor metode de detecţie de $F_0$ care să furnizeze rezultate mai bune decât utilitarul Praat™ sau alte utilitare existente.

## *2. Descrierea instrumentului de analiză prozodică*

Instrumentul dezvoltat pentru extragerea informaţiei prozodice conţine mai multe blocuri funcţionale corespunzătoare celor trei etape de procesare a semnalului vocal: preprocesarea, extragerea traseului intonaţional pe baza valorilor frecvenţei fundamentale, extragerea valorilor formanţilor superiori (Fig.1).

În etapa de preprocesare se realizează filtrări ale semnalului cu scopul de a elimina zgomotul nedorit şi de limitare a benzii de frecvenţă în care se caută valorile formantice. De asemenea, în această etapă, un algoritm de segmentare permite eliminarea zonelor consonantice şi a celor de pauză între rostiri şi extragerea zonelor vocalice. Doar pentru aceste secvenţe vocalice, care corespund vocalelor limbii române şi consoanelor semivocalice are sens să fie realizată detecţia lui $F_0$ şi a formanţilor.

Acest instrument de analiză este un sistem hibrid neuro fuzzy prin blocul decizional ce determină $F_0$, prin ponderarea rezultatelor furnizate de cele patru metode diferite de extragere a frecvenţei fundamentale (metoda autocorelaţiei 55%, metoda cepstrală 35%, metoda diferenţelor AMDF 5% şi metoda HPS 5%). Ponderile utilizate reflectă performanţele fiecărei metode, estimate ca număr de detecţii eronate. Am asociat ponderi mai mici metodelor cu o probabilitate mai mare de a furniza date eronate. Detecţiile eronate (în principal prima subarmonică, respectiv primele armonici ale lui

$F_0$) sunt găsite prin compararea valorilor de ieşire consecutive furnizate de aceeaşi metodă şi/sau de alte metode, realizându-se şi o corecţie a acestor „false" detecţii de $F_0$.

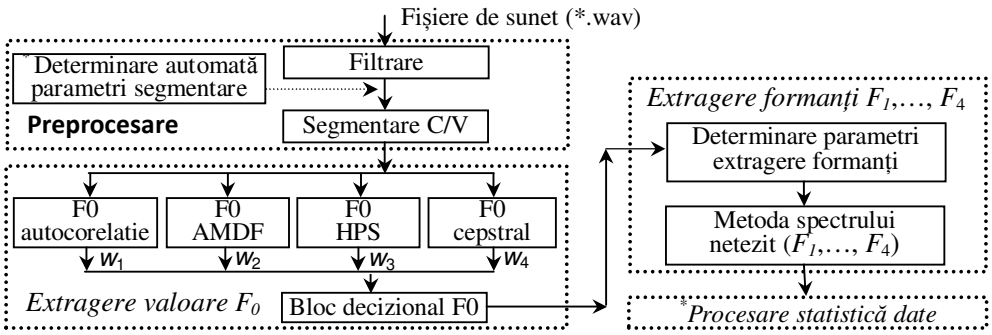



**Figura 1**: Schema bloc sistem hibrid neuro-fuzzy de extragere informaţii prozodice

Etapa de determinare a valorilor formanţilor superiori se bazează pe acurateţea detecţiei frecvenţei fundamentale prin stabilirea unor intervale de căutare a fiecărui formant în funcţie de $F_0$. Sunt folosite tehnici fuzzy de concatenare a „spectrelor netezite" rezultate din cepstrele obţinute cu diferiţi parametri, dar şi asocierea de coeficienţi de apartenenţă pentru fiecare candidat găsit în benzile (intervalele fuzzy) de căutare a fiecărui formant.

Ideea utilizării unui bloc decizional hibrid este, din câte ştim, originală şi a permis obţinerea de rezultate mai bune la o inspecţie vizuală a liniei prozodice, fată de alte instrumente similare de analiză a informaţiei prozodice existente pe web: Klatt, Praat (Boersma & Weenink, 2006), Goldwave (www.goldwave.com), Wasp (www.wasp.dk), Speech Analyzer (www.sil.org/computing/sa), Winpitch (www.winpitch.com ).

## *3. Tehnici de segmentare C/V*

În această lucrare prezentăm doar prima fază de execuţie a instrumentului de analiză prozodică şi anume etapa de preprocesare şi segmentare, care trebuie să se realizeze cu cât mai puţine erori posibile, deoarece toate etapele ulterioare se bazează pe aceasta. Erorile din faza de segmentare vor afecta în mare măsură eroarea finală de detecţie. Determinăm erorile de segmentare comparând segmentele vocalice identificate automat cu cele marcate manual într-un fişier adnotat de mare precizie, folosind utilitarul Praat. Metodologia de notare şi marcare a fonemelor este descrisă în cele ce urmează.

### *3.1 Metodologia de adnotare*

Adnotarea prin ascultare este subiectivă şi greu de realizat cu precizie, fiind necesară în paralel şi inspectarea vizuală a formei de undă şi a spectrului semnalului (spectrogramei). Pe baza informaţiilor legate de periodicitatea semnalului, de vârfurile spectrale şi tranziţiile acestora, de componenţa în frecvenţe înalte se poate realiza delimitarea fiecărui fonem. Chiar şi utilizând toate aceste informaţii, stabilirea cu exactitate a acestor limite este dificilă, mai ales datorită perioadelor de tranziţie dintre foneme, a perioadelor de amortizare, a zgomotelor introduse de aerul aspirat/expirat, de sunetele produse la închiderea buzelor etc.

Deoarece adnotările uzuale nu au o precizie suficientă (de ordinul ms), a fost necesară realizarea manuală a unei adnotări de mare precizie, prin metoda (practicată frecvent de al doilea autor) a comparării formei de undă în domeniul timp şi domeniul frecvenţelor.

Durata minimă a pauzelor intravorbire sau a secvenţelor care pot fi sesizate de urechea umană este de ordin zecimi milisecunde, şi de aceea pentru a se ajunge la precizia de ordin ms este necesară şi analiza vizuală a semnalului. Scopul autorilor este de a determina erorile date de alte instrumente, în cazul de faţă Praat, şi de a le compara cu cele date de extractorii de $F_0$ implementaţi de colectivul SRoL. În acest scop s-au adnotat 10 fişiere din baza de date a sit-ului SRoL (fişierele de sunet provin de la 6 vorbitori, 3 de gen feminin şi 3 de gen masculin). Faţă de metodologia de adnotare aplicată uzual, în plus s-a ţinut cont în procesul de adnotare de următoarele:

- S-au introdus notaţii suplimentare care să permită specificarea sunetelor specifice limbii române (‘â’ = ‘a-‘, ‘ă’ = ‘a+’, ‘ş’ = ‘sh’, ‘ţ’ =’ tz’).
- S-au marcat în mod diferit zonele de pauză astfel: pauzele intervorbire (între rostiri de propoziţii) cu ‘ ‘ (caracterul blanc), pauzele intravorbire (silabe, cuvânt) cu ‘$’, pauzele care nu se aud (prezente în consoanele ‘p’, ‘t’, ‘c’ etc.) cu ‘%’.
- S-au stabilit intervalele de demarcaţie ale fiecărui fonem atât prin inspecţia vizuală a formei de undă (pentru a observa periodicitatea semnalelor în cazul în care acestea sunt vocalice), analiza spectrului şi spectrogramei, cât şi prin inspecţie auditivă.
- S-a validat de mai mulţi evaluatori delimitarea fonemelor şi a pauzelor intravorbire.

### *3.2 Descriere algoritmi de segmentare C/V*

Algoritmii utilizaţi anterior în segmentare estimau pentru fiecare fereastră de analiză (de dimensiune uzuală *N*=512 sau 1024 eşantioane) energia în domeniul timp şi energia spectrală a frecvenţelor joase şi comparându-le cu nişte valori de prag stabileau dacă acele segmente erau vocalice sau consonantice. Etapele algoritmului sunt următoarele:

1) Aplicarea unui filtru Butterworth de ordin 11 în banda [70,6000]Hz (se păstrează informaţia din banda de căutare a formanţilor şi se elimină zgomotul indus de reţea).

2) Parcurgerea întregului semnal şi determinarea energiei maxime $E_W = \sum_{i=1}^{N} |s_i|$ a unei ferestre de analiză *W*, cu dimensiune *N* eşantioane. Folosind această valoare $E_{W\max}$ se parcurge din nou semnalul şi se consideră că în zonele în care energia ferestrei curente este mai mică decât 20% din energia maximă nu avem zonă vocalică.

3) Pentru fiecare fereastră de analiză se calculează energia spectrală totală $E_{FFT}{}^{t} = \sum_{f=0}^{Fs/2} |FFT(s_W)|$ şi energia din banda frecvenţelor joase [70,2500]Hz $E_{FFT}{}^{B} = \sum_{f=70}^{2500} |FFT(s_W)|$ ( $Fs$ este frecvenţa de eşantionare). Dacă $E_{FFT}{}^{B} < 0.5 \cdot E_{FFT}{}^{t}$, energia corespunzătoare frecvenţelor înalte este mai mare decât 50% din energia spectrală totală, atunci se consideră că zona respectivă nu este vocalică.

Primul criteriu de segmentare a zonelor consonatice/vocalice este o metodă globală al cărui scop este acela de a elimina zonele de pauză dintre pronunţii, unde este prezent doar zgomotul ambiental, precum şi o serie de consoane care conţin zone de pauză (de

exemplu plozivele). Al doilea criteriu de segmentare C/V bazat pe o metodă locală de estimare a ponderii componentelor spectrale are rolul de a elimina consoanele care au o energie a frecvențelor înalte mai mare în raport cu frecvențele joase.

Valorile de prag utilizate în segmentare au fost determinate empiric după mai multe simulări. Algoritmul de segmentare nu funcționează bine, atunci când există zone cu energia în domeniul timp mult mai mare decât în restul pronunției. Astfel de zone apar datorită intonației (accentuarea unui cuvânt sau silabe), sau la exprimarea unei emoții (cum ar fi starea de furie, bucurie). Din acest motiv s-a testat și o variantă de segmentare în care valoarea energetică maximă este calculată pe o zonă locală restrânsă la 0.5 secunde în jurul eșantionului curent.

S-a realizat de asemenea și determinarea statistică a valorilor optime de prag folosind un algoritm de antrenare supervizat, care să determine pe baza seturilor de antrenare reguli de identificare a zonelor vocalice de cele consonantice și de zonele de pauză dintre cuvinte sau fraze. Seturile de antrenare conțineau pentru fiecare fonem: *zcr* (rata trecerilor prin zero pe secundă), *avg_e*, *std_e* - energia medie și deviația standard a energiei în domeniul timp a ferestrelor de analiză, energia spectrală în benzile B1 [70, 500]Hz, B2 [500, 1000]Hz, B3 [1000, 2000]Hz, B4 [2000, 5000]Hz. S-a preferat în locul rețelelor neuronale sau algoritmilor genetici, utilizarea arborilor de decizie, deoarece aceștia furnizează la ieșire un set de reguli, care au în premise valorile prag determinate automat pentru o clasificare cu eroare minimă. Exemplu de regulă obținută cu See5 (www.rulequest.com/see5-win.html) pentru identificarea zonelor vocalice:

```
Rule 1: (270/120, lift 1.4) ;aplicabilă pentru 150 pattern-uri din cele 270
     IF   E_MED > 0.000516
          B4 <= 0.068311
     THEN class 1 (vowel)  [0.555]
```

Folosirea de metode de segmentare cu valori prag nu oferă întotdeauna rezultate bune. De exemplu, aceste valori de prag nu mai pot fi folosite pentru a delimita zonele de pauză de cele în care vorbitorul rostește ceva, atunci când se vorbește încet, sau când persoana a fost amplasată prea departe de microfon (deși înregistrările s-au efectuat după un protocol care prevede o distanță optimă de la buze la microfon), sau când nivelul de zgomot ambiental este prea mare sau fonemul (vocala), de la finalul secvenței rostite are o energie scăzută și/sau o durată de atenuare mai mare. O soluție pentru această problemă este ajustarea pragurilor în funcție de SNR.

Chiar și criteriul de segmentare bazat pe energia frecvențelor înalte nu reușește să determine unele vocale aflate în hiat/diftong, dificultăți fiind întâlnite în acest caz în zonele de tranziție. Un alt caz este cel al fonemului '*a*' aspirat (pronunțat în timp ce aerul este aspirat pe gură) din propoziția '*A trecut el așa un răstimp*' care este încărcat în frecvențe înalte, dar își păstrează traseele formantice uzuale. O serie de foneme consonantice pot fi găsite în unele pronunții ca zone vocalice (de exemplu 'v', 'z', 'r' etc.), dar există situații în care acestea învecinate fiind cu mai multe consoane nu prezintă acea periodicitate a semnalului și sunt clasificate ca și consoane.

În consecință, este nevoie de un algoritm de segmentare care să verifice dacă semnalul este cvasi-periodic. Criteriile de segmentare utilizate anterior ar putea doar furniza informații suplimentare privind clasa de apartenență (vocală / consoană / pauză rostiri) a

semnalului din fereastra de analiză. În aceste condiții s-a preferat utilizarea funcției de autocorelație, care s-a dovedit a fi și cea mai robustă din punct de vedere a erorilor de extragere a lui $F_0$, într-un algoritm adaptat pentru faza de segmentare:

1) Filtrarea întregului semnal folosind un filtru de mediere de ordin $N$=31;

$$s_{filt}[k] = \sum_{i=1}^{31} s[k+i]/31 \ , \ k = \overline{1, len\_s - 31},$$

unde $len\_s$ reprezintă lungimea semnalului de intrare. Alegerea ordinului filtrului este justificată de rezultatele simulărilor, prezentate în secțiunile următoare.

```
for i = 1 to n_iteratii
```

2) Calcularea vectorului de valori *scorr* prin aplicarea funcției de corelație pentru fereastra curentă de analiză.

3) Căutarea maximului din vector și calcularea valorii $vF0 = Fs / poz_{\max}$. Dacă această valoare este situată între $poz_{start} = Fs/500$ și $poz_{end} = Fs/80$, atunci „*considerăm*" acel segment ca fiind vocalic, altfel $vF0 = 0$.

```
end_for
```

4) Se determină vectorul boolean $\{z(vF0) \mid z : [0,500] \to \{0,1\}\}$ al variabilității semnalului $vF0$, considerând că între două valori consecutive există variabilitate, dacă între acestea avem o tranziție (variație) mai mare $\pm 5\%$.

$$z[k] = \begin{cases} 1 & vF0[k] > 1.05 \cdot vF0[k+1] \ \& \ vF0[k] < 0.95 \cdot vF0[k+1] \\ 0 & altfel \end{cases}, \quad k = \overline{1, len\_s - 1}.$$

5) Dacă în vectorul $z$ avem variabilitate mai mare de $p\%$ atunci considerăm zona respectivă ca nefiind nevocalică. Se obține în final semnalul $\hat{s}_{segm}$

$$\hat{s}_{segm}[k] = \begin{cases} s[k] & , \hat{z}[k] = \sum_{i=-N/2}^{N/2} z[k+i] < N \cdot p\% \\ 0 & altfel \end{cases}.$$

În ultima etapă a algoritmului am considerat pragul de variabilitate de $p$=5%: dacă din $N = 500$ de valori avem mai puțin de 20 care variază cu $\pm 5\%$ față de valorile vecine, considerăm zona respectivă ca fiind vocalică. Pentru vectorul *vF0* se pot stabili și reguli statistice care să impună ca deviația standard pentru un număr consecutiv de valori să nu depășească o valoare prag dată. Variabila *n_iterații* se deduce în funcție de dimensiunea ferestrei de analiză, $W$ și de pasul de deplasare al ferestrei, $n\_iteratii = \lfloor (len\_s - W)/step \rfloor$. Pentru o segmentare cu o rezoluție maximă a semnalului de ieșire, pasul de deplasare se alege $step = 1$.

În etapa a treia se consideră că o valoare de pe poziția $k$ din vectorul *scorr* este maxim local, dacă este mai mare decât valoarea din dreapta *scorr*[$k$+1] și din stânga acesteia *scorr*[$k$-1]. Căutarea maximului corespunzător lui $T_0$ se face doar în lista maximelor locale astfel găsite. Se evită astfel selectarea ca maxim a primei valori din vector $scorr[0] = \sum_{i=1}^{N} s^2[i]$, care corespunde energiei semnalului din fereastra curentă de analiză.

Un dezavantaj al metodei propuse îl constituie numărul mare de operații care trebuiesc efectuate. Sunt necesare cinci parcurgeri ale unor vectori comparabili ca dimensiune cu semnalul de intrare, $s_{filt}$ obținut după filtrare folosind o fereastră de $N$=31 eșantioane,

$s_{corr}$ rezultat după calcul funcției de autocorelație cu *N*=1024 eșantioane, $z$ vector variabilitate pentru fiecare două eșantioane consecutive și $\hat{z}$ vectorul variabilității cumulate pentru *N*=500 (folosit la generarea semnalului de ieșire $\hat{s}_{segm}$).

Pentru a obține timpi de rulare mai mici am optimizat algoritmul de segmentare prin folosirea unor funcții recursive de calcul a vectorilor $s_{filt}$, $s_{corr}$, $\hat{z}$, esențială fiind scăderea timpului de calcul la nivelul funcției de autocorelație.

### *3.3 Optimizarea algoritmului de segmentare prin funcții recursive*

Funcția de autocorelație necesită un număr mare de operații, timpi mari de calcul, fiind un algoritm de complexitate $O(N^2)$, motiv pentru care s-a evitat anterior utilizarea ei în faza de segmentare pentru a studia periodicitatea semnalelor. Din această cauză, în cazul extractorului de $F_0$ s-a preferat folosirea unui pas mai mare de deplasare a ferestrei $step = N/16$, unde *N*=1024 reprezintă dimensiunea implicită a ferestrei de analiză.

Pentru rezoluția maximă (*step*=1) s-au obținut pentru un fișier de 10-20 secunde timpi de rulare de ordin 5-10 minute. Prezentăm pseudo-codul algoritmului de calcul a vectorului de autocorelație, pentru fiecare fereastră de analiză, *W,* selectată la parcurgerea semnalul de analizat, *s* cu pas de deplasare al ferestrei $step$.

```
for j=1: step : niter*step
//Calculează C_XX^Wj funcția de autocorelație pentru fereastra curentă W_j
   for k=0:N
      for i=1:P
         C_XX[k] = C_XX[k] + x[i]·x[i+k]
```

În limbajul MATLAB de exemplu, funcția de autocorelație folosește doar valori din fereastra curentă (de dimensiune *N*), motiv pentru care variabila *i* variază între 1: *N-k*. În acest caz formula de calcul pentru funcția de autocorelație devine:

$$C_{XX}[k] = \sum_{i=1}^{N-k} x[i] \cdot x[i+k], \quad k = \overline{0, N}$$

Pentru optimizare am folosit o relație de recurență în calculul funcției de autocorelație a unei ferestre $W_2$ pornind de la șirul de valori al ferestrei anterioare $C_{XX}^{W1}$.

$$C_{XX}^{W1}[0] = x_1^2 + x_2^2 + ... + x_N^2 \qquad C_{XX}^{W1}[k] = x_1 \cdot x_{k+1} + x_2 \cdot x_{k+2} + \ldots + x_{N-k} \cdot x_N$$

$$C_{XX}^{W2}[0] = x_2^2 + \ldots + x_N^2 + x_{N+1}^2 \qquad C_{XX}^{W2}[k] = x_2 \cdot x_{k+2} + \ldots + x_{N-k} \cdot x_N + x_{N-k+1} \cdot x_{N+1}$$

$$C_{XX}^{W2}[0] = C_{XX}^{W1}[0] - x_1^2 + x_{N+1}^2 \qquad C_{XX}^{W2}[k] = C_{XX}^{W1}[k] - x_1 \cdot x_{k+1} + x_{N-k+1} \cdot x_{N+1}$$

Relația de recurență de mai sus este calculată pentru cazul în care între cele două ferestre avem un pas de deplasare minimal (*step*=1), fapt ce conduce la o rezoluție maximă a semnalului de prelucrat. Procesul computațional ar necesita în loc de $N \cdot (N+1)/2$ operații de adunare și înmulțire un număr de doar $2 \cdot N$ operații și, teoretic, timpul de calcul ar trebui să scadă de $N/4$ ori. Astfel pentru parcurgerea întregului semnal cu o fereastră de analiză de 1024 eșantioane și calculul funcției de autocorelație am avea teoretic un timp de calcul de aproximativ 250 ori mai mic.

Crescând pasul de deplasare, creşte şi numărul de operaţii de efectuat, astfel încât timpii finali de calcul nu vor fi mai mici decât în cazul în care *step*=1. Aceasta se observă din relaţia de recurenţă rezultată:

$$C_{XX}^{W2}[k] = C_{XX}^{W1}[k] - \sum_{j=1}^{Step} x_j \cdot x_{k+j} + \ldots + \sum_{j=1}^{Step} x_{N-k+2-j} \cdot x_{N+2-j}$$

Pentru filtrarea semnalului s-a preferat folosirea filtrului de mediere neponderat în locul unui filtru digital FIR Butterworth, Bessel, Chebyshev pentru a putea aplica o relaţie de recurenţă. Pentru filtru de mediere ponderat:

$$s_{filt}[k] = \left( \sum_{i=1}^{N} s[k+i] \cdot a[i] \right) \Big/ \sum_{i=1}^{N} a[i],$$

o relaţie de recurenţă devine posibilă doar dacă coeficienţii $a[i]$ sunt egali între ei ($a[0] = \ldots = a[i] = \ldots = a[N]$ obţinându-se un filtru de mediere neponderat):

$$s_{filt}[k+1] = s_{filt}[k] - s[k]/N + s[k+N]/N, \qquad N=31.$$

În aceeaşi manieră s-au folosit funcţii recursive la calculul vectorului de variaţie $\hat{z}$.

$$\hat{z}[k+1] = \hat{z}[k] - z[k] + z[k+N], \quad N=500$$

Folosirea funcţiilor recurente de calcul a condus la obţinerea unor timpi de rulare mai mici de câteva zeci de ori, făcând posibilă aplicarea algoritmului de segmentare propus.

## *4. Simulări şi rezultate*

În urma implementării practice a metodei de segmentare propuse, pentru obţinerea unor erori minime în segmentare s-au formulat următoarele întrebări:
- unde este util să se facă filtrarea: înainte sau după calculul vectorului de autocorelaţie?
- care este ordinul optim pentru filtrul de mediere?
- care este valoarea de prag p% care trebuie folosită pentru obţinerea lui $\hat{s}_{segm}$?

Pentru a răspunde la primele două întrebări s-au folosit semnale armonice de test

$$s = A_1 \cdot \sin(2\pi \cdot f_0 \cdot t) + A_1 \cdot \sin(2\pi \cdot f_1 \cdot t) + A_3 \cdot \sin(2\pi \cdot f_2 \cdot t)$$

în care valorile frecvenţelor $f_0, f_1, f_2$ se aleg apropiate de valorile frecvenţei fundamentale $F_0$ şi respectiv ale frecvenţelor primilor doi formanţi, $F_1$ şi $F_2$. În aceste condiţii, valoarea de maxim care trebuie detectată şi salvată în vectorul *vF0* ar trebui să fie cât mai apropiată de $f_0$.

S-au testat două variante de filtrări, una în care se realizează filtrarea înainte (independent) de vectorul funcţiei de autocorelaţie (notată în figurile 2 şi 3 cu (*v2*)) şi una în care se realizează filtrarea în final după calcul vectorului *scorr* (notată cu (*v1*)). Ne interesează acest studiu, deoarece dacă în urma simulărilor s-ar obţine erori de detecţie a frecvenţei fundamentale semnificativ mai mici pentru (*v1*), decât pentru (*v2*) atunci nu s-ar mai putea optimiza algoritmii de segmentare folosind relaţii recurente. În figura 3 este reprezentată eroarea medie de detecţie a lui $F_0$, pentru mai multe studii de caz $\{F_0,F_1,F_2\}$ = {100,350,900}, {200,550,1100}, {70,350,900}, {100,400,800}, etc., variind ordinul filtrului în mulţimea {1,5,11,15,21,25,31,35,41,45,51}.

Din figura 2 se observă că eroarea de detecție a frecvenței $f_0$ este minimă pentru un filtru de mediere de ordin 31 și că algoritmul de segmentare funcționează mai bine atunci când funcția de autocorelație este calculată pentru o fereastră de 1024 eșantioane, în varianta *v*2, după ce în prealabil s-a realizat și filtrarea semnalului.

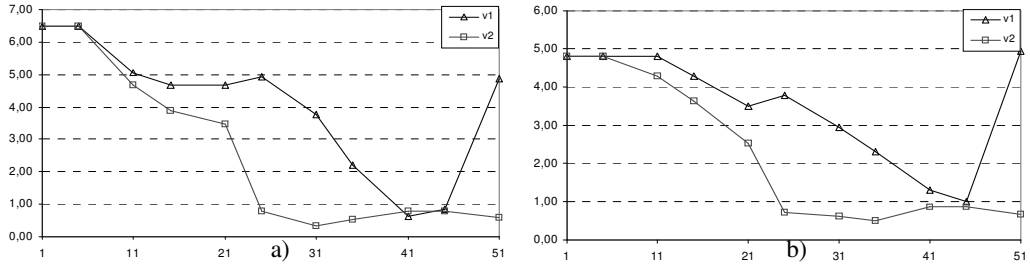



**Figura 2**: Eroare de detecție a lui $F_0$ în condițiile în care (v1) filtrarea se face după calcul funcției de autocorelație, sau (v2) filtrarea se face înainte de calculul vectorului *scorr*
a) fereastră de analiză de 512 eșantioane, respectiv b) 1024 eșantioane

Funcționarea optimă a metodei de segmentare cu un filtru de mediere de ordin *N*=31, poate fi explicată prin comportamentul de filtru trece jos FTJ, cu valoarea riplului de aproximativ 520Hz, în condițiile în care banda de lucru este [70-500]Hz.

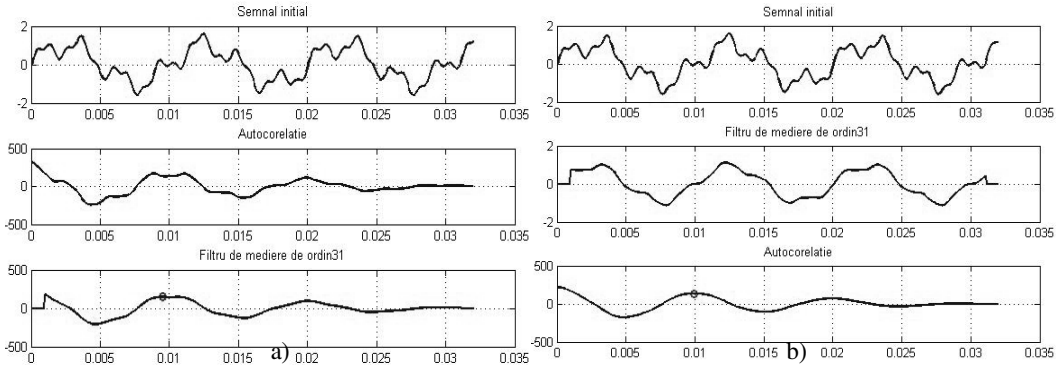


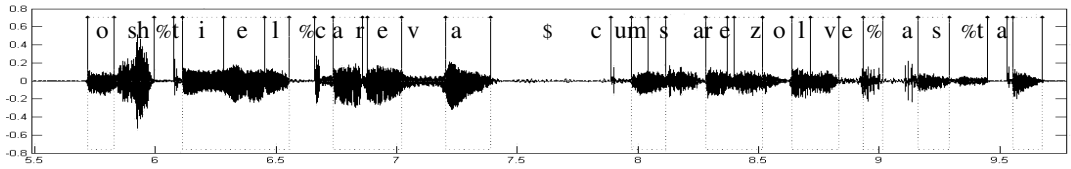

**Figura 3**: Semnal de test (s=sin(2*pi*t*100)+0.5*sin(2*pi*t*350)+0.2*sin(2*pi*t*900))
a) varianta v1 filtrarea se face după calculul funcției de autocorelație, f0=105.3
b) varianta v2 filtrarea se face înainte, f0=100.6

Un extractor de frecvență fundamentală furnizează următoarele erori: (i) E1: determină $F_0$ acolo unde nu este sunet vocalic (fals pozitiv); (ii) E2: nu determină $F_0$ acolo unde este sunet vocalic (fals negativ); (iii) E3: determină eronat $F_0$ ca valoare. (iv) E4: erori de decizie asupra valorii finale a lui $F_0$, atunci când avem mai multe metode de detecție.

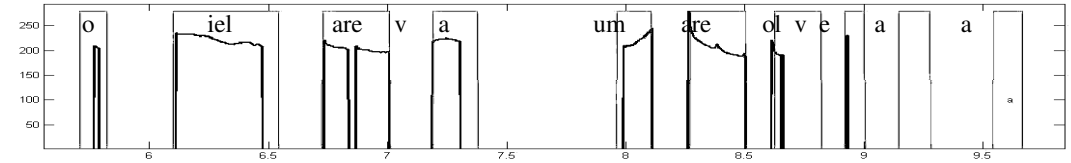
Erorile de segmentare automată sunt specificate de E1 și de E2. La acestea pot contribui și erorile de adnotare (segmentare manuală) în cazul în care aceasta nu este realizată cu precizie. Erorile extractorilor de F0 sunt date de valoarea lui E3, iar la eroarea finală se poate adăuga și eroarea blocului decizional a sistemului hibrid neuro-fuzzy E4.
Semnificative ca valori sunt primele două surse de erori, de care este responsabil blocul de segmentare, E3 fiind eliminate aproape în întregime de algoritmii de corecție, iar valorile lui E4 sunt aproape neglijabile și cuantizabile doar prin inspecție vizuală.
În figurile 3 și 4 sunt prezentate rezultatele segmentării/detecției de F0 cu instrumentele proprii și cu Praat™ pentru aceeași pronunție „*O ști el careva cum să rezolve asta*”) din fișierul de sunet *accent_cuv_urm_v2.wav*. Dacă algoritmii anteriori de segmentare nu

reuşeau să separe toate zonele vocalice, având dificultăţi cu acele foneme care aveau valoarea amplitudinii/energiei mai mică, cu algoritmul nou propus se segmentează foarte bine, chiar şi zone pe care instrumentul Praat nu le detectează. Astfel fonemele '*l*' şi '*e*' din cuvântul '*rezolve'*, vocala '*a*' finală au traseul intonaţional detectat fără discontinuităţi. Nu avem nici false detecţii în cazul fonemelor '*ş*' şi '*c*'. Algoritmul nou propus are câteva zone înguste de ordin ms, situate în zonele de pauză, în care găseşte izolat valori periodice. Pentru a le elimina se pot adăuga restricţii, ca zonele considerate ca fiind foneme vocalice să aibă o durată de minim 5ms, sau restricţii privitoare la energia semnalului care în zonele de pauză sunt semnificativ mai mici decât în segmentele rostite, dacă înregistrarea nu a fost realizată într-un mediu zgomotos.

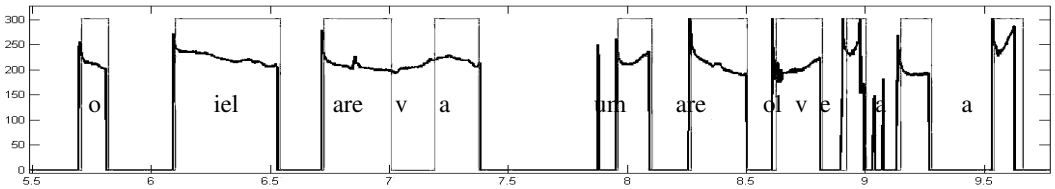
Erorile de segmentare prezentate în tabelul 1 au fost realizate considerând ca zone vocalice segmentele extrase din fişierele de adnotare corespunzătoare vocalelor, diftongilor, consoanelor sonante '*l*','*m*','*n*','*r*'. Nu sunt luate în calcul unele foneme care uneori se comportă atât vocalic (forma lor de undă este periodică), cât şi consonantic. Este cazul fonemului '*v*' care, în '*careva'* este vocalic, dar în '*rezolve'* este consonantic. Zonele de tranziţie dintre foneme vocalice, uneori sunt şi ele nevocalice (de exemplu pentru pronumele '*el*' pronunţat ca regionalism '*iel*', s-au găsit astfel de tranziţii între cele două vocale). Din zonele detectate de program ca fiind vocalice s-au eliminat şi la stanga şi la dreapta o jumătate din fereastră de analiză (N/2 eşantioane), corespunzătoare în general zonelor de tranziţie dintre consoane şi vocale.




a) semnalul iniţial de intrare (pronunţia „O şti el careva cum să rezolve asta")




b) semnalul segmentat cu algoritmi bazaţi pe valori de prag fixe (p1=20% din energia maximă în domeniul timp, p2=50% din energia spectrală)




c) semnal segmentat cu noul algoritm de segmentare (bazat pe studiul periodicităţii semnalului)

**Figura 4.** Rezultate segmentare şi detecţie F0 (instrumente proprii)

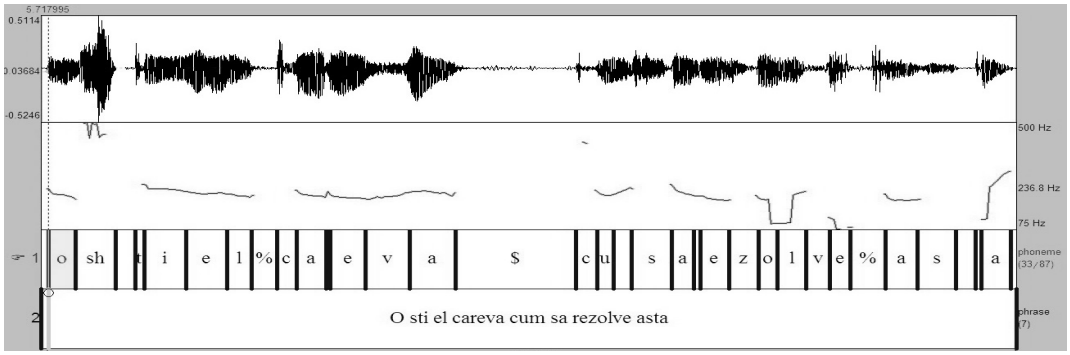



**Figura 5.** Rezultate segmentare și detecție F0 (soft Praat™)

Tabel 1: Erori extractor de frecvență fundamentală

| **Fișier de intrare** | **F0 - Metoda autocorelației** | | | **F0 – Praat™** | | |
|---|---|---|---|---|---|---|
| | **E1** | **E2** | **E3** | **E1** | **E2** | **E3** |
| accent_cuv_urm_v1.wav | 0.133 | 0.064 | 0.032 | 0.192 | 0.011 | 0.057 |
| accent_cuv_urm_v2.wav | 0.084 | 0.096 | 0.058 | 0.153 | 0.044 | 0.049 |
| accent_cuv_urm_v12.wav | 0.128 | 0.100 | 0.065 | 0.184 | 0.018 | 0.763 |

## *5. Concluzii. Direcții viitoare*

Algoritmul de segmentare propus este mai puțin influențat de amplitudinea și fluctuațiile în amplitudine ale semnalului analizat și prin urmare poate fi adaptat și pentru înregistrări cu un nivel de zgomot mai mare. Este necesară introducerea unor criterii noi de segmentare pentru detecțiile izolate de $F_0$ din zonele de pauză, respectiv definirea unor valori de prag flexibile, ajustate automat în funcție de SNR (va trebui estimată amplitudinea zgomotului și cea a semnalului util). Se va încerca utilizarea altor parametri care să fie mai puțin influențați de nivelul energetic, cum ar fi rata trecerilor prin zero, pentru semnalul din care s-au eliminat întâi componentele de joasă frecvență.

Conform simulărilor, algoritmii proprii oferă erori mai puține de tipul E1 ($F_0$ în zone nemarcate ca sunet vocalic) și de tipul E3 (se determină eronat $F_0$ ca valoare – cu fluctuații). În lipsa unor metode de corecție, soft-ul Praat™ produce „falsele” detecții, ceea ce reprezintă un inconvenient în eroarea globală a sistemului, în comparație cu instrumentele proprii care elimină în proporție foarte mare aceste erori.

**Mulțumiri**. Cercetarea a fost realizată cu sprijinul Academiei Române, în cadrul temei interne a Institutului de Informatică Teoretică din Iași. Autorii mulțumesc celorlalți co-autori ai sitului Sunetele Limbii Române, precum și referenților anonimi pentru sprijinul și observațiile pertinente.

**Contribuția autorilor**: Primul autor a implementat instrumentelor de analiză în mediile de programare MATLAB și C++; a elaborat metoda de extragere formanți și optimizarea metodelor de segmentare și extragere a $F_0$ și a formanților. Al doilea autor a inițiat tema, a coordonat activitatea de cercetare, a elaborat conceptul general de sistem de decizie și metode de extragere $F_0$ conform cu fig.1, a precizat metoda de adnotare și segmentare manuală cu elemente de noutate privind metoda proprie de combinare a informației temporare cu cea spectrală. Al treilea autor a realizat înregistrările, a efectuat manual adnotările de mare precizie și a identificat problemele de segmentare. Toți autorii au contribuit la analiza rezultatelor și identificarea soluțiilor de îmbunătățire și validarea rezultatelor acestora.

## Referințe bibliografice

Boersma, P., Weenink, D., Institute of Phonetic Science, University of Amsterdam, Praat: doing phonetics by computer, *www.praat.org*.

Calliope (1989). *La parole et son traitement automatique*, ISBN 2-225-81516-X, Masson, France.

Esposito, A. & Aversano, G. (2005). Text independent Methods for speech Segmentation, *Lecture Notes in Computer Science*, ISBN 978-3-540-27441-4, 3445, 261-290 (http://www.springerlink.com/content/81fpb3brpq367j7g/).

Juneja, A. & Espy-Wilson, C. (2002). Segmentation of Continuous speech using acoustic-phonetic parameters and statistical learning, *Proceedings International Conference on Neural Information Processing*, (http://www.ece.umd.edu/~juneja/paper1910.PDF), Universitatea din Maryland, Singapore, SUA.

Matousek, J., Tihelka, D., Psutka, J. (2003). Automatic Segmentation for Czech concatenative speech synthesis using statistical approach with boundary-specific correction, *Processing of Eurospeech* 2003, Geneva, 301-304.

Rabiner, L.R., Juang B.H. (1993). *Fundamentals of Speech Recognition* Englewood Cliffs, N.J.

Rabiner, L.R. Schafer R. W. (1978). *Digital Processing of Speech Signal*, Prentice-Hall, Inc. Englewood Clifford, 11-65.

Rowden, C. (1991). *Speech Processing*, McGraw - Hill Book Company, Chapter 2, 35-74.

Salam, M.S., Mohamad, D., Salleh, S.H. (2009). Improved Statistical Speech Segmentation Using Connectionist Appoach, *J. of Computer Science*, ISSN 1549-3636, 5 (4): 275-282.

Sarkar, A. & Sreenivas, T.V. (2005). Automatic speech segmentation using average level crossing rate information, *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing: ICASSP*, Philadelphia, SUA, 1, 397-400.

Teodorescu, H.N., Feraru, M., Pistol L., Zbancioc, M. şi alții. (2005). SRoL – Proiectul Sunetele Limbii Române (Voiced Sounds of Romanian Language Project), 2005. [http://iit.iit.tuiasi.ro/romanain_spoken_language/ index.htm]

Teodorescu, H.N., Feraru M., Zbancioc M.D. (2009) Assesing the Quality of Voice Synthesizers, In Burileanu C., Teodorescu H.N. (Eds.), *Advances in Spoken Language Technology*, The Publishing House of the Romanian Academy, Bucharest, România, ISBN 978-973-27-1808-7, 53-66.

Teodorescu, H.N., Trandabat, D., Feraru, M., Zbancioc, M., Luca, R. (2007). A Corpus of the Sounds in the Romanian Spoken Language for Language-Related Education, In Carlos Perinan Pasqual (Eds.), *Revisiting Language Learning Resources*, Cambridge Scholars Publishing (CSP), UK, ISBN 1-84718-156-2, 6, 73-89.

Vidal, E. & Marzal, A. (1990). A review and new approaches for automatic segmentation of speech signals, Signal Processing V: *Theories and Applications*, Torres, L., Masgrau, E., Lagunas, M.A. (eds.), Elsevier Science Publishers B.V. – Universitatea Politehnica din Valencia, Spania.

Zbancioc, M. (2006). Tools for the Archive of the Romanian Language Sounds Project, 4th *European Conf. on Intelligent Systems and Technologies*, ECIT'2006, Iași, Romania.