

Optimizarea unui sistem BCI folosind tehnica GA

Dan Marius Dobra, Monica-Claudia Dobra

Abstract – Această lucrare, ce continuă o cercetare anterioară, are ca prim obiectiv îmbunătățirea unui sistem de tip interfață creier-calculator (BCI) ce utilizează o nouă metodă de extragere de trăsături, numită proces adaptiv neliniar în amplitudine și fază (ANAPP). În aplicația de față, metoda ANAPP modelează semnalele EEG ca o combinație de cinci „oscilații corticale spontane” alese *a priori* și ale căror amplitudini și faze sunt determinate cu ajutorul unui algoritm adaptiv. Dacă într-o serie de cercetări anterioare [1], [2] trăsăturile de amplitudine ale modelului au fost în mod extensiv utilizate în aplicații de tip BCI, în prezenta cercetare este investigată oportunitatea folosirii suplimentare și a informației de fază. În plus, un alt obiectiv al prezentului studiu îl constituie și optimizarea numărului și a tipului de trăsături de intrare utilizate în cadrul sistemului de clasificare, optimizare realizată cu ajutorul unui algoritm GA. Toate aceste demersuri au ca scop final obținerea unui sistem BCI mai rapid și, totodată, mai performant din punct de vedere al rezultatelor clasificării.

I. INTRODUCERE

Un sistem de tip interfață om-calculator (BCI), bazat pe semnalul EEG, este un echipament capabil să achiziționeze, proceseze, interpreteze și comande alte echipamente sau sisteme folosind pentru aceasta semnalele electroencefalografice (EEG) achiziționate de la un subiect uman. Adesea, sub-sistemul de clasificare – parte integrantă a oricărei aplicații BCI de acest tip – utilizează una sau mai multe trăsături extrase din semnalul EEG în scopul de a discrimina între un număr de task-uri mentale ce se consideră a fi diferențiabile la nivelul dinamicii semnalelor EEG pe care acestea le generează.

Trăsăturile EEG utilizate în clasificarea task-urilor mentale și raportate în literatură sunt de o diversitate rar întâlnită. Cele mai frecvent folosite trăsături în aplicațiile BCI sunt: coeficienții AR [3], [4], [5], [6], modelele AR cu intrări exogene [6], parametrii spectrali de putere [8], [7], [9], [10], [11], sincronizarea statistică a fazei [8], [9], filtrarea spațială [12], valoarea medie a coerenței în fază [8], frecvența de descărcare a unui grup neuronal [15], unda P300 [12], [13], [14] etc.

Lucrarea de față adresează aceeași problemă, respectiv, aceea de a găsi acele trăsături EEG cele mai adecvate pentru aplicațiile de clasificare de task-uri cognitive. Rezultatele obținute și prezentate în această lucrare fac parte dintr-un proiect mult mai complex; scopul final al acestui proiect este acela de a dezvolta și implementa un sistem robotic autonom, cu auto-organizare, comandat mental de un subiect ce emite una din următoarele comenzi (asociate unor task-uri mentale): înainte, înapoi, stânga, dreapta. Întrucât sistemul se dorește a fi unul on-line, vom urmări în cele ce urmează să îmbunătățim și să optimizăm sistemul BCI dezvoltat și prezentat anterior în alte două lucrări [1], [2]. Acest sistem BCI discutat în [1], [2] a implementat în faza de procesare a semnalului EEG o nouă metodă de extragere a vectorilor de trăsături – metodă denumită proces adaptiv neliniar în amplitudine și fază (ANAPP).

Metoda ANAPP a modelat semnalele EEG ca o combinație de cinci „oscilații corticale spontane” specificate *a priori* iar amplitudinile și fazele acestora au fost stabilite printr-un algoritm adaptiv. Aceste oscilații corticale spontane, necesare a fi specificate *a priori* în modelarea ANAPP, au fost determinate folosind funcția de coerență calculată pentru fiecare task mental în parte. Mai departe, parametrii estimați ai modelului ANAPP au fost utilizați ca date de intrare într-o rețea neuronală artificială (ANN) de tip perceptron multistrat (MLP) care la ieșire a furnizat clasa corectă de apartenență a eșantionului de intrare.

În cele două cercetări anterioare menționate mai sus și vizând îmbunătățirea performanțelor de clasificare în domeniul BCI, doar trăsăturile de amplitudine ale modelului ANAPP au fost exploatate și utilizate ca intrări pentru clasificatorul de tip ANN. Spre deosebire de acestea, studiul de față urmărește două obiective

majore: (i) primul obiectiv este acela de a determina dacă trăsăturile de fază ale modelului ANAPP (singure sau în combinație cu trăsăturile de amplitudine) pot fi utilizate în vederea creșterii performanțelor de clasificare și (ii) al doilea obiectiv constă în optimizarea numărului trăsăturilor de intrare (amplitudine, faza sau o combinație a acestora) cu ajutorul unui algoritm genetic (GA) cu următoarele efecte sperate – performanțe de clasificare superioare, capacitate de generalizare a clasificatorului ANN mai mare și un sistem BCI mai rapid.

II. MATERIALE ȘI METODE

A. Achiziția datelor

Datele EEG utilizate în această lucrare au fost achiziționate de la 4 subiecți în timp ce aceștia executau cinci task-uri mentale diferite. Semnalele EEG au fost înregistrate de la nivelul a 3 perechi de electrozi, plasați la nivelul scalpului în următoarele poziții ale sistemului internațional 10-20: central (C3, C4), parietal (P3, P4) și, respectiv, occipital (O1, O2). Canalele EEG au fost referite la mastoidea dreaptă, A2, iar datele achiziționate au fost eșantionate la o frecvență de 250 Hz. Fiecare înregistrare a durat 10 s.

Subiecții au realizat toate task-urile fără vocalizare și cu ochii închiși. Task-urile executate au fost după cum urmează [2]: (1) task de relaxare (subiecții s-au relaxat cât mai mult posibil), (2) scrierea unui scrisori (subiecții au compus mental o scrisoare către un prieten), (3) task de numărare (subiecții au urmărit numere scrise secvențial pe o tablă imaginară), (4) task aritmetic (subiecții au realizat o înmulțire non-trivială) și (5) task de rotire (subiecții au studiat timp de 30 s un obiect 3D după care, obiectul a fost îndepărtat, iar subiecților li s-a cerut să rotească imaginar obiectul în jurul unei axe). În plus, subiecții au realizat câte două înregistrări pentru fiecare task menționat mai sus.

B. Funcția de coerență

Funcția de coerență indică, în domeniul frecvenței, gradul de corelație liniară dintre două semnale diferite. O valoare nulă pentru această funcție are semnificația independenței statistice dintre cele două semnale în timp ce o valoare egală cu 1 pentru aceeași funcție semnifică o dependență liniară completă între semnalele analizate.

O valoare estimată a coerenței a fost calculată pentru fiecare pereche de semnale, $x(t)$ și $y(t)$, folosind formula:

$$\left| \hat{R}_{xy}(\lambda) \right|^2 = \frac{\left| \hat{f}_{xy}(\lambda) \right|^2}{\hat{f}_{xx}(\lambda) \hat{f}_{yy}(\lambda)}. \quad (1)$$

Auto- și cross-spectrele din ecuația (1) au fost, la rândul lor, estimate printr-o mediere făcută peste câteva secvențe de semnal EEG suprapuse:

$$\hat{f}_{xy}(\lambda) = \frac{1}{2\pi LT} \sum_{l=1}^L F_x^T(\lambda, l) \overline{F_y^T(\lambda, l)} \quad (2)$$

În relația (2) bara de deasupra lui $F_x^T(\lambda, l)$, $\overline{\quad}$, indică un complex conjugat iar (x, y) sunt perechile de semnale (s_1, s_2) pentru cross-spectrum, și (s_1, s_1) , respectiv, (s_2, s_2) pentru autospectre; aici, prin s_1 și s_2 am reprezentat două semnale EEG diferite, înregistrate la nivelul unui același electrod activ, însă la momente de timp diferite. Un prim pas în obținerea formulelor pentru funcția auto-spectrului și, respectiv, pentru

funcția cross-spectrului, a presupus împărțirea fiecăreia dintre cele două serii de timp analizate în L ferestre de lungime T , suprapuse. Semnalul $x(t)$ a fost înlocuit în cazul nostru cu $s_1(t)$ – prima înregistrare obținută pentru un anumit subiect, un anumit task și, respectiv, un anumit canal EEG iar semnalul $y(t)$ a fost înlocuit cu $s_2(t)$ – a doua înregistrare obținută pentru același subiect, același task și, respectiv, același canal EEG. Pentru a facilita consistența interpretării, un estimat al coerenței în masă [2] a fost calculat folosindu-se. În acest scop un test statistic. Întreaga metodologie utilizată pentru a obține oscilațiile corticale spontane este prezentată în [2].

C. Procesul adaptiv neliniar în amplitudine și fază

După determinarea frecvențelor EEG spontane cu ajutorul funcției de coerență, seriile de timp EEG originale au fost modelate folosind în acest sens un model adaptiv în amplitudine și fază.

Ipoteza fundamentală a modelului ANAPP pornește de la ideea că semnalul EEG poate fi în mod adecvat descompus în câteva componente fundamentale de frecvență (oscilații spontane, specificate *a priori*) precum și în frecvențele cuplate neliniar asociate acestora (oscilații auto-cuplate, respectiv, oscilații de cros-cuplare). Mai exact, două unde oscilatorii (de frecvență f_1 și f_2 – așa cum e cazul semnalelor generate de doi oscilatori corticali), ce trec printr-un sistem neliniar de gradul doi (spre exemplu, diferite căi neuronale) generează două tipuri de frecvențe armonice: armonici de auto-cuplare ($2 \cdot f_1$ and $2 \cdot f_2$) și, respectiv, armonici de cros-cuplare, ($f_1 \pm f_2$).

În mod corespunzător, semnalul EEG modelat, $y[n]$, s-a presupus a fi compus din $K = 5$ oscilații diferite ($x_j, j=1 \div K$). În ecuațiile prezentate în (3), T_s este rata de eșantionare, f_j este a j -a frecvență fundamentală, ϕ_j este faza sa inițială iar a, b, c și d reprezintă parametrii de amplitudine ai modelului. O dată determinați, parametrii de amplitudine și de fază ai modelului EEG au fost apoi utilizați ca date de intrare ale sistemului de clasificare.

$$\left\{ \begin{array}{l} y[n] = \sum_{m=1}^K a_m[n] x_m[n] + \sum_{m=1}^L b_m[n] \alpha_m[n] + \\ + \sum_{\substack{l=1 \\ l \neq m}}^M \sum_{m=1}^N c_{lm}[n] \beta_{lm}[n] + \sum_{\substack{l=1 \\ l \neq m}}^P \sum_{m=1}^Q d_{lm}[n] \theta_{lm}[n] \\ x_j[n] = \sin(n \cdot T_s \cdot 2\pi \cdot f_j + \phi_j) \\ \alpha_i[n] = \sin(n \cdot T_s \cdot 2\pi \cdot \underbrace{2f_j}_{f_i} + \phi_i) \\ \beta_{ij}[n] = \sin[n \cdot T_s \cdot 2\pi \cdot (f_i - f_j) + \phi_{ij}] \\ \theta_{ij}[n] = \sin[n \cdot T_s \cdot 2\pi \cdot (f_i + f_j) + \phi_{ij}] \end{array} \right. \quad (3)$$

Algoritmul LMS a fost utilizat în estimarea adaptivă a parametrilor modelului (mai exact, a amplitudinilor a_j, b_j, c_{ij}, d_{ij} și a fazelor $\phi_i, \phi_{ij}, i, j = 1 \div K, i \neq j$). Pentru aceasta, eroarea pătratică, dată de:

$$e[n]^2 = [s[n] - y[n]]^2 \quad (4)$$

a fost utilizată în definirea funcției de cost J (mai precis, $J = 1/2 \cdot E\{e[n]^2\}$) ce a trebuit minimizată. În ecuația (4), $s[n]$ este semnalul EEG real în timp ce $y[n]$ reprezintă semnalul EEG modelat.

Aplicând relația LMS, în [1], [2] am dedus următoarele formule de ajustare a parametrilor de amplitudine ai modelului ANAPP:

$$a_j[n+1] = a_j[n] + \eta_{a_j} e[n] x_j[n] \quad (5)$$

$$b_j[n+1] = b_j[n] + \eta_{b_j} e[n] \alpha_j[n] \quad (6)$$

$$c_{ij}[n+1] = c_{ij}[n] + \eta_{c_{ij}} e[n] \beta_{ij}[n] \quad (7)$$

$$d_{ij}[n+1] = d_{ij}[n] + \eta_{d_{ij}} e[n] \theta_{ij}[n] \quad (8)$$

În ceea ce privește parametrii de fază asociați, după aplicarea relațiilor LMS ale lui Windrow, am obținut următoarele relații:

$$\phi_j[n+1] = \phi_j[n] + \eta_{\phi_j} \cdot e[n] \cdot a_j[n] \cdot \cos(n \cdot T_s \cdot 2\pi \cdot f_j + \phi_j) \quad (9)$$

$$\varphi_i[n+1] = \varphi_i[n] + \eta_{\varphi_i} \cdot e[n] \cdot b_i[n] \cdot \cos(n \cdot T_s \cdot 2\pi \cdot \underbrace{2f_j}_{f_i} + \varphi_i) \quad (10)$$

$$\phi_{ij}[n+1] = \phi_{ij}[n] + \eta_{\phi_{ij}} \cdot e[n] \cdot c_{ij}[n] \cdot \cos(n \cdot T_s \cdot 2\pi \cdot (f_i - f_j) + \phi_{ij}) \quad (11)$$

$$\varphi_{ij}[n+1] = \varphi_{ij}[n] + \eta_{\varphi_{ij}} \cdot e[n] \cdot d_{ij}[n] \cdot \cos(n \cdot T_s \cdot 2\pi \cdot (f_i - f_j) + \varphi_{ij}) \quad (12)$$

III. REZULTATE ȘI DISCUȚII

În această lucrare o raportare a rezultatelor obținute de noi la alte rezultate similare prezentate în literatură s-a făcut doar pentru un singur subiect, și anume subiectul 2, la fel ca și în [1] și [2]. Această alegere a fost una justificată în principal de faptul că acest subiect este indicat în literatura de specialitate ca înregistrând întotdeauna, indiferent de metodologia de analiză folosită, performanțele cele mai mari de clasificare. Acest fapt ne-a făcut să presupunem că subiectul s-a concentrat într-o măsură mai mare decât ceilalți 3 subiecți asupra task-urilor executate.

Vectorii de trăsături aplicați la intrarea clasificatorului ANN au fost obținuți prin concatenarea parametrilor modelului ANAPP calculați pentru ferestre alunecătoare de semnal EEG de lungime 256 eșantioane, înregistrate simultan de la toate cele șase canale EEG. Dimensiunea vectorilor de trăsături a variat în funcție de analiza făcută, respectiv, dacă s-au folosit doar parametrii de amplitudine, doar parametrii de fază sau dacă s-au folosit ambele tipuri de parametri ai modelului. Toate canalele EEG au fost filtrate digital cu un filtru trece-sus, având o frecvență de tăiere la 20 Hz.

Pentru fereaste alunecătoare de 2250 eșantioane, suprapuse cu 12 eșantioane, am obținut, în final, 1670 vectori de trăsături pentru intrarea clasificatorului ANN (167 de vectori pentru fiecare înregistrare * 2 înregistrări * 5 task-uri mentale). Din acest set de vectori de intrare, 80% au fost selectați aleator pentru a forma setul de antrenare a rețelei (1336 de vectori) și 20% pentru setul de cros-validare (CV), respectiv, 334 vectori.

Necesitatea pre-filtrării digitale a semnalelor EEG a apărut din ideea larg acceptată că cele mai importante vârfuri de frecvență (cu alte cuvinte, cea mai mare parte a puterii semnalului EEG) se regăsește, de obicei, în banda 0 – 20 Hz, lucru care face dificilă obținerea unui model ANAPP de încredere pentru semnalul EEG (aproape toate oscilațiile corticale spontane *a priori* au fost determinate ca fiind situate într-un interval superior frecvenței de 20 Hz). În plus, pentru ca modelul ANAPP propus să nu introducă frecvențe în banda 0 – 20 Hz sau mai mari de jumătate din frecvența de eșantionare (frecvența Shannon în cazul nostru fiind de

125 Hz), am forțat toate ratele de învățare, precum și valorile inițiale ale amplitudinilor să devină zero doar pentru acele frecvențe derivate care se regăseau în intervalele de valori menționate mai sus.

În acest mod, prin eliminarea componentelor de frecvență de valoare zero, vectorii de trăsături formați din parametrii de amplitudine au fost reduși de la 180 de componente la doar 104 componente. Aceeași situație o regăsim și în cazul vectorilor de trăsături formați din parametrii de fază.

TABEL 1 MATRICEA CONFUZIILOR OBTINUTĂ PENTRU CLASIFICAREA CELOR 4 TASK-URI [1]

		Clase atribuite			
		T2	T3	T4	T5
Clase reale	T2	79.6%	3.2%	6.3%	10.9%
	T3	2.9%	84.1%	5.7%	7.3%
	T4	6.1%	13.6%	77.3%	3%
	T5	14.7%	10.3%	2.9%	72.1%

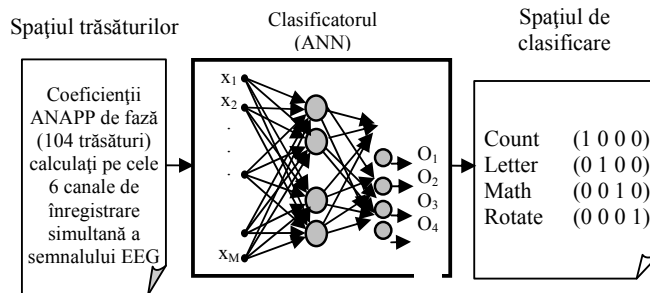


Figura 1. Procesul de clasificare

Utilizând modelul ANAPP al semnalelor EEG, un clasificator de tip MLP și metodologiile prezentate în lucrările [1] și [2], am obținut în final rezultatele prezentate în Tabelul 1. La aceste rezultate s-a ajuns în urma unei analize extensive privind topologia ANN, ratele de învățare, precum și ratele de moment. Munca de căutare a celei mai bune configurații pentru rețeaua ANN a avut la bază pe lângă strategia unui expert uman, și un mare număr de încercări ce au vizat atingerea celei mai bune performanțe de clasificare. Arhitectura cea mai performantă găsită pentru rețeaua MLP a constat într-un strat de intrare cu 104 intrări, un strat ascuns cu 40 de elemente de procesare (EPs) și un strat de ieșire cu 4 neuroni de ieșire. Fiecare dintre acești ultimi neuroni au corespuns la câte un task mental din cele 4 folosite în procesul de clasificare.

Rezultatele afișate în Tabelul 1 vor fi considerate, în cele ce urmează, ca date de referință pentru analiza procesului de optimizare a clasificării pe care îl vom prezenta în cele ce urmează; acest proces urmărește, în esență, obținerea unui sistem BCI mai rapid (operabil în timp real) și cu performanțe de clasificare superioare. În același timp, rețeaua MLP folosită pentru a obține aceste prime rezultate va fi și ea, de asemenea, folosită ca parte nemodificată (dată) a sistemului BCI.

A. Trăsăturile de fază ale semnalului EEG

Într-o primă analiză s-a testat capacitatea trăsăturilor de fază de a conține, ele singure, informații utile capabile să conducă la rate de clasificare superioare. Pentru această analiză seturile de date de intrare și a datelor dorite au fost construite așa după cum se prezintă în Figura 1.

Rezultatele obținute pentru sistemul de clasificare de mai sus sunt cele indicate în Tabelul 2. În acest tabel, ca de altfel și în toate tabelele care îi urmează, task-urile au fost indicate astfel: T2 – count, T3 – letter,

T4 – math, și T5 – rotate.

TABEL 2. MATRICEA CONFUZIILOR OBȚINUTĂ ÎN CAZUL FOLOSIRII DOAR A INFORMAȚIILOR DE FAZĂ

		Clase atribuite			
		T2	T3	T4	T5
Clase reale	T2	35.54%	17.52%	31.35%	15.59%
	T3	20.19%	51.2%	16.27%	12.34%
	T4	18.29%	16.39%	54.79%	10.53%
	T5	12.71%	24.94%	15.75%	46.6%

Se poate ușor remarca faptul că utilizarea doar a informației de fază din modelul ANAPP a condus la obținerea unor performanțe de clasificare net inferioare celor considerate în această lucrare drept rezultate de referință (a se vedea Tabelul 2 versus Tabelul 1). În concluzie, parametrii de fază ai modelului ANAPP sunt capabili să diferențieze cele 4 task-uri, însă performanțele de clasificare sunt mult sub cele obținute folosind doar informația de amplitudine oferită de modelul ANAPP.

Într-o a doua analiză, s-a testat capacitatea ambelor tipuri de informații furnizate de modelul ANAPP (parametrii de fază și de amplitudine la un loc) de a obține un efect sinergic, reflectat în rate de clasificare mai mari decât cele obținute folosind fie doar informația de fază, fie doar pe cea de amplitudine. În ciuda tuturor așteptărilor, rezultatele obținute de această dată s-au dovedit a fi chiar mai mici decât cele obținute în Tabelul 2. o posibilă explicație a unui astfel de comportament ar putea fi, printre multe alte explicații posibile, și un neajuns tehnic, dat de mărimea setului de antrenare care, în cazul nostru, este una foarte mică.

Se știe că numărul vectorilor de trăsături din setul de antrenare, N , necesar pentru a clasifica corect un set de test, cu o eroare ε dată poate fi calculat estimativ cu următoarea relație [16]:

$$N > \frac{W}{\varepsilon}. \quad (13)$$

unde W este numărul de ponderi ale clasificatorului ANN. În cazul nostru particular, considerând o rețea MLP cu 208 intrări (104 parametri de amplitudine și alți 104 parametri de fază) și având un număr de cel puțin 40 de neuroni pe stratul ascuns și alți 4 neuroni pe stratul de ieșire, conform calculului obținem un număr total minim de 8480 de ponderi ale rețelei. În această situație, pentru o eroare de cca 10%, numărul minim de vectori de trăsături ce constituie setul de antrenare ar trebui să fie mai mare de 84800 – lucru nerealizabil ținând cont că dispunem doar de 1336 astfel de vectori. În consecință, mărimea setului de date de antrenare ar putea fi un motiv justificabil pentru performanțele de clasificare mai mici obținute anterior. În aceste condiții, luând în considerare faptele prezentate și constrângerile existente (de ex., mărimea setului de date), în continuare s-a încercat găsirea unei metode de selecție doar a acelor trăsături (de fază și de amplitudine) cu adevărat utile, capabile să îmbunătățească considerabil performanțele clasificării.

A. Optimizarea cu ajutorul tehnicii GA

Pentru a rezolva problema mărimii vectorilor de trăsături s-a folosit, în continuare, metoda algoritmilor genetici (GA). Această metodă selectează dintr-un set dat de trăsături (în cazul nostru acesta poate fi dat fie doar de setul trăsăturilor de fază, fie doar de cele de amplitudine, fie de amândouă tipurile de trăsături) doar acele trăsături care sunt importante pentru procesul de clasificare (de ex., acele intrări care conțin informații ce asigură cea mai mare discriminare între clase).

Un cromozom folosit în cadrul metodei GA a fost compus dintr-o serie de valori (în cazul nostru doar două valori au fost permise, respectiv 0 și 1). Numărul de valori (0 și 1) dintr-un cromozom a fost setat ca fiind egal cu numărul componentelor unui vector de trăsături (și egal, de asemenea, cu numărul de intrări ale clasificatorului MLP). Aceste intrări au fost apoi selectate sau deselectate ca rezultat al aplicării algoritmului

genetic; mai precis, o valoare de 0 a deselectat intrarea corespunzătoare în timp ce o valoare de 1 a selectat intrarea corespunzătoare. Costul mediu (costul pătrat mediu al ieșirii rețelei), calculat pe setul de CV, a fost folosit în calitate de criteriu de fitness pentru GA, minimizarea lui fiind un obiectiv al algoritmului GA.



Figure 1. Evoluția algoritmului genetic pentru o ANN având ca intrări trăsăturile de amplitudine și de fază

În mod obișnuit, această tehnică de optimizare necesită ca rețeaua ANN să fie antrenată în repetate rânduri pentru a putea găsi combinația optimă de intrări care să producă cea mai mică eroare (de ex., o populație de 50 de cromozomi ar presupune, în fiecare generație, cca 50 de antrenări ale rețelei ANN). Din acest motiv devine o necesitate găsirea, în prealabil (înaintea începerii optimizării cu GA), a unei rețele MLP optime (topologie, rate de învățare, rate ale momentului, tipuri de neliniarități ale funcțiilor de activare etc.) care să aibă caracteristici de convergență optime; cu alte cuvinte, rețeaua neuronală trebuie să aibă o dinamică stabilă și, mai apoi, un timp de convergență cât mai mic. În acest mod, timpul pierdut de ANN este minim iar, ca un rezultat direct, fiecare generație GA ia un timp mai mic.

Utilizând această abordare (și anume, optimizarea cu GA a mărimii vectorilor de trăsături), problema de clasificare anterioară, în care s-au folosit atât informațiile de fază cât și cele de amplitudine, poate fi depășită.

În Figura 2 este prezentată evoluția algoritmului genetic, cu redarea fitness-ului pentru cel mai bun individ (costul pătratic mediu calculat pe setul de CV, pentru cel mai bun cromozom al fiecărei generații). Din această figură se poate remarca capacitatea algoritmului GA de a îmbunătăți performanțele de clasificare folosindu-se pentru aceasta de un set optim de vectori de trăsături selectați în mod continuu de-a lungul procesului evolutiv. Performanțele de clasificare obținute la sfârșitul evoluțiilor GA sunt cele prezentate în Tabelul 3. Din păcate aceste performanțe sunt în continuare inferioare celor prezentate în Tabelul 1. În acest context, și ținând cont, în plus, și de performanțele cele mai slabe obținute în cazul trăsăturilor de fază, un pas firesc l-a reprezentat în continuare optimizarea doar a setului trăsăturilor de amplitudine.

După o cercetare extensivă, s-a decis că o rețea ANN, având un singur strat ascuns cu 23 Eps, toate ratele momentului egale cu 0.95, cu o rată de învățare de 0.35 pe stratul ascuns și una de 0.03 pe stratul de ieșire, prezintă cele mai bune caracteristici de convergență pentru cazul nostru particular.

TABEL 3. MATRICEA CONFUZIILOR OBȚINUTĂ ÎN CAZUL UTILIZĂRII TRĂSĂTURILOR DE FAZĂ ȘI AMPLITUDINE SELECTATE CU ALGORITMUL GA

		Clase atribuite			
		T2	T3	T4	T5
Clase reale	T2	55.38%	12.31%	21.54%	10.77%
	T3	10%	74.29%	7.14%	8.57%
	T4	17.19%	9.37%	71.88%	1.56%
	T5	10.29%	23.53%	5.88%	60.3%

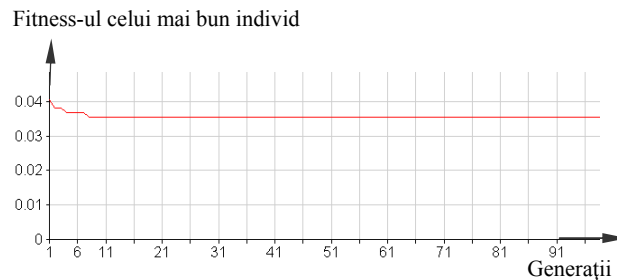


Figure 2. Evoluția algoritmului genetic

Algoritmul GA a fost implementat cu ajutorul unei populații de 15 cromozomi, folosind o metodă de selecție de tip ruletă, un operator de tip crossover uniform și un operator standard ce realizează operația de mutație. Probabilitățile pentru operatorii de tip crossover și mutație au fost setați la valorile de 0.9 și 0.001. De asemenea, au fost testate o serie de metode diferite de crossover (de ex., într-un singur punct și, respectiv, în două puncte), la fel ca și diferite scheme de selecție : de tip rang, de tip turnir, de tip eșantionare stohastică uniformă și de tip eșantionare stohastică cu memorie. Pentru aplicația noastră, cel mai eficient operator de crossover a fost cel uniform, în timp ce cea mai eficientă schemă de selecție s-a dovedit a fi schema de tip ruletă. O evoluție genetică are loc până în momentul în care numărul maxim de generații (100 în cazul nostru) a fost atins.

În Figura 3 este prezentată evoluția algoritmului genetic, cu funcția de fitness pentru cel mai bun individ. Rezultatele clasificării obținute pentru cel mai bun cromozom, pe setul de CV, după convergența algoritmului GA, sunt cele prezentate în Tabelul 4.

TABEL 4. MATRICEA CONFUZIILOR OBTINUTĂ ÎN CAZUL TRĂSĂTURILOR DE AMPLITUDINE OPTIMIZATE CU GA

		Clase atribuite			
		T2	T3	T4	T5
Clase reale	T2	82.81%	6.25%	4.68%	6.26%
	T3	2.9%	79.71%	7.25%	10.14%
	T4	6.06%	7.58%	86.36%	0%
	T5	11.76%	13.24%	8.82%	66.18%

Dacă, comparăm rezultatele prezentate în Tabelul 4 cu ratele de referință din Tabelul 1, observăm o ușoară îmbunătățire a performanțelor de clasificare obținute în cazul vectorilor de amplitudine optimizați cu tehnicile GA. Suma elementelor de pe diagonala principală (a ratelor de clasificare corectă) din Tabelul 1 este de 313.1 și de 315.05 pentru cea din Tabelul 4. Ținând cont de această informație, performanța medie de clasificare în cazul aplicației de referință este de 78.275, în timp ce, cea obținută în cazul optimizării cu GA este de 78.7625.

Îmbunătățirea înregistrată la nivelul ratelor de clasificare pare să fie una ne semnificativă și, în consecință, inconsistentă, însă această ușoară îmbunătățire este susținută și de o altă paradigmă în care o rețea ANN cu 23 EPs pe stratul ascuns (față de cei 40 EPs utilizați în aplicația de referință) și – cel mai important aspect – cu un proces de optimizare de tip GA, implementat, a confirmat rezultatele de mai sus. În urma procesului de optimizare un număr de doar 80 de trăsături au fost selectate din setul complet de 104 trăsături de amplitudine. O consecință directă a acestui rezultat a fost și scăderea complexității sistemului de clasificare de la un număr de 4320 de ponderi (104 intrări * 40 neuroni +40 neuroni * 4 neuroni) pentru rețeaua ANN de referință, la doar 1932 de ponderi (80 intrări * 23 neuroni +23 neuroni * 4neuroni) pentru rețeaua optimizată ANN. În acest mod, complexitatea noului sistem de clasificare a fost redus aproape la jumătate (55.27%) din ANN de referință.

Reducerea complexității noului sistem de clasificare prezintă un număr de avantaje. În primul rând,

sistemul este mai rapid datorită scăderii încărcăturii computaționale asociată cu actualizarea fiecărei ponderi (în timpul algoritmului de backpropagare) și cu asocierea clasei noi pentru vectorul de trăsături de intrare (în pasul forward).

În al doilea rând, utilizând același set de date de antrenare, capacitățile de generalizare a noii rețele ANN pot fi crescut prin folosirea unui număr mai mic de vectori de trăsături, vezi relația (13). Din relația (13) se poate observa faptul că pentru un același set de date și un set mai mic de ponderi (mai mult decât jumătate) eroarea poate fi scăzută corespunzător iar rata de clasificare poate fi crescută.

IV. CONCLUZII

Din rezultatele prezentate anterior se poate remarca capacitatea algoritmului GA de a optimiza setul de trăsături în vederea obținerii unor rate de clasificare mai mari și a unei complexități mai mici a rețelei ANN. Prin utilizarea GA complexitatea rețelei ANN optimizate a fost redusă la mai puțin de o jumătate – ajungând la un număr de 2388 de ponderi. În acest mod, am obținut o rețea neuronală mai rapidă – caracteristică ce o apropie mai mult de obiectivul final, și anume, acela de a implementa un sistem BCI în timp real.

În ceea ce privește parametrii de fază ai modelului ANAPP, se poate concluziona faptul că acest tip de trăsături EEG nu aduc informații noi și nici nu cresc puterea discriminatorie a sistemului BCI. Două explicații posibile pentru acest comportament ar putea fi luate în considerare, și anume: în primul rând, în analiza noastră am utilizat valoarea instantanee a parametrului de fază, în loc să folosim forma sa derivată (despre care se crede că poartă mai multă informație) și, în al doilea rând, constrângerile impuse către modelul ANAPP însuși, și anume, selecția doar a 5 componente spectrale fundamentale fapt care a eliminat probabil o parte semnificativă din informațiile utile.

Această cercetare pune accentul încă o dată, dacă mai era cazul, pe importanța mărimii setului de date de antrenare și confirmă performanțele modelului ANAPP.

ACKNOWLEDGMENT

This work was entirely supported by the Romanian National University Research Council under Grant ID 1552.

REFERENCES

- [1] D. M. Dobrea, and M. C. Dobrea, "EEG Classification System – From an Universal System Implementation to a Particular Signal Modeling," Proceedings of the Romanian Academy - Series A: Mathematics, Physics, Technical Sciences, Information Science, Vol. 10, Nr. 2, May–August 2009, pp. 197-204, ISSN 1454-8267
- [2] D. M. Dobrea, and M. C. Dobrea, "An EEG (Bio)Technological System For Assisting the Disabled People," Proceeding of the 5th International Conference on Computational Cybernetics, ICCC 2007, October 19-21 2007, Gammarth, Tunisia, pp. 191-196
- [3] C. Guger, A. Schlögl, C. Neuper, D. Walterspacher, T. Strein, and G. Pfurtscheller, "Rapid prototyping of an EEG-based brain-computer interface (BCI)," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 9, nr. 1, 2001, pp. 49-58
- [4] C. W. Anderson, E. Stolz, S. Shamsunder, "Multivariate Autoregressive Models for Classification of Spontaneous Electroencephalogram During Mental Tasks," IEEE Transactions on Biomedical Engineering, vol. 45, nr. 3, pp. 277-286, 1998
- [5] V. A. Maiorescu, M. C. Dobrea (Șerban), and A.M. Lazar, "Classification of EEG Signals Represented by AR Models for Cognitive Tasks – a Neural Network Based Method," International Symposium on Signal Circuits and System, vol. 2, 10-11 iulie 2003, Iasi, Romania, pp. 441-444

- [6] D. P. Burke, S. P. Kelly, de P. Chazal, R. B. Reilly, and C. Finucane, "A parametric feature extraction and classification strategy for brain-computer interfacing," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 13, nr. 1, 2005, pp. 12-17
 - [7] B. J. Culpepper, and R. M. Keller, "Enabling Computer Decisions Based on EEG Input," *IEEE Trans. on Neural Syst. and Rehab. Eng.*, vol. 11, no. 4, 2003, pp. 354-360
 - [8] E. Gysels, and P. Celka, "Phase synchronization for the recognition of mental tasks in a brain-computer interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 12, nr. 4, 2004, pp. 406 – 415
 - [9] M. C. Dobrea (Șerban), and D.M. Dobrea, "Discrimination between cognitive tasks - a comparative study," "Proceedings of the International Symposium on Signals Circuits and Systems – ISSCS 2005, Iași, România, 14 – 15 iulie 2005, Vol. 2, pp. 805 - 808, ISBN 0-7803-9029-6
 - [10] S. P. Kelly, E. C. Lalor, C. Finucane, G. McDarby, and R. B. Reilly, "Visual Spatial Attention Control in an Independent Brain-Computer Interface," *IEEE Transactions on Biomedical Engineering*, lucrare acceptată spre publicare, 2005, pp. 1 – 9
 - [11] T. Müller, T. Ball, R. Kristeva-Feige, T. Mergner, and J. Timmer, "Selecting Relevant Electrode Positions For Classification Tasks Based On The Electro-Encephalogram," *Medical & Biological Engineering & Computing*, 2000, vol. 38, pp. 62-67
 - [12] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, "BCI2000: a general-purpose brain-computer interface (BCI) system," *IEEE Transactions on Biomedical Engineering*, vol. 51, nr. 6, 2004, pp. 1034-1043
 - [13] J. R. Wolpawa, N. Birbaumer, D. J. McFarlanda, G. Pfurtschellere, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clinical Neurophysiology*, vol. 113, 2002, pp. 767–791
 - [14] H. Serby, E. Yom-Tov, and G. F. Inbar, "An improved P300-based brain-computer interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 13, nr. 1, 2005, pp. 89 – 98
 - [15] S. G. Mason, and G. E. Birch, "A general framework for brain-computer interface design," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 11, nr. 1, 2003, pp. 70-85
- S. Haykin, "Artificial Neuronal Networks: A Comprehensive Foundation," IEEE Press, New York, 1995