

CONTRIBUȚII LA CREȘTEREA EFICIENȚEI NEURONULUI CU PONDERI NUMERE COMPLEXE ȘI FUNCȚIE PERIODICĂ DE ACTIVARE

Teză destinată obținerii
titlului științific de doctor inginer
la
Universitatea *Politehnica* Timișoara
în domeniul ȘTIINȚA CALCULATOARELOR
de către

Ing. Valentin Mircea Lupea

Conducător științific:
Referenți științifici:

prof.univ.dr.ing. Ștefan Holban
prof.univ.dr.ing. Robert Gyorodi
prof.univ.dr.Alexandru Cicortaș
prof.univ.dr.ing. Ionel Jian

Ziua susținerii tezei: 14 Februarie 2014

Seriile Teze de doctorat ale UPT sunt:

- | | |
|---|--|
| 1. Automatică | 9. Inginerie Mecanică |
| 2. Chimie | 10. Știința Calculatoarelor |
| 3. Energetică | 11. Știința și Ingineria Materialelor |
| 4. Ingineria Chimică | 12. Ingineria sistemelor |
| 5. Inginerie Civilă | 13. Inginerie energetică |
| 6. Inginerie Electrică | 14. Calculatoare și tehnologia informației |
| 7. Inginerie Electronică și Telecomunicații | 15. Ingineria materialelor |
| 8. Inginerie Industrială | 16. Inginerie și Management |

Universitatea Politehnica Timișoara a inițiat seriile de mai sus în scopul diseminării expertizei, cunoștințelor și rezultatelor cercetărilor întreprinse în cadrul Școlii doctorale a universității. Seriile conțin, potrivit H.B.Ex.S Nr. 14 / 14.07.2006, tezele de doctorat susținute în universitate începând cu 1 octombrie 2006.

Copyright © Editura Politehnica – Timișoara, 2014

Această publicație este supusă prevederilor legii dreptului de autor. Multiplicarea acestei publicații, în mod integral sau în parte, traducerea, tipărirea, reutilizarea ilustrațiilor, expunerea, radiodifuzarea, reproducerea pe microfilme sau în orice altă formă este permisă numai cu respectarea prevederilor Legii române a dreptului de autor în vigoare și permisiunea pentru utilizare obținută în scris din partea Universității Politehnica Timișoara. Toate încălcările acestor drepturi vor fi penalizate potrivit Legii române a drepturilor de autor.

România, 300159 Timișoara, Bd. Republicii 9,
Tel./fax 0256 403823
e-mail: editura@edipol.upt.ro

Cuvânt înainte

Teza de doctorat a fost elaborată pe parcursul activității mele în cadrul Departamentului de Automatică și Informatică Aplicată al Universității "Politehnica" Timișoara.

Mulțumiri deosebite se cuvin conducătorului de doctorat Prof.dr.ing. Ștefan Holban, cât și membrilor din comisia de îndrumare: Conf.univ.dr.ing. Doru Todincă, Conf.univ.dr.ing. Dan Pescaru și Ș.L.dr.ing.Cosmin Cernăzanu-Glăvan. Doresc să le mulțumesc pentru timpul și sfaturile acordate de-a lungul ciclului de doctorat.

Vreau, de asemenea să mulțumesc familiei pentru susținerea permanentă pe parcursul acestor ani de studiu.

De asemenea mulțumiri se cuvin și Universității *Politehnica* Timișoara pentru oportunitatea oferită de a urma și finaliza ciclul doctoral, cât și programului POSDRU pentru asigurarea suportului financiar de-a lungul acestei perioade.

Doresc să mulțumesc tuturor celor care în mod direct sau indirect, m-au ajutat la realizarea acestui material.

This thesis was possible with the partial financial support of the strategic grant POSDRU/CPP107/DMI1.5/S/77265 (2010) of the Ministry of Labor, Family and Social Protection, Romania, co-financed by the European Social Fund – Investing in people.

Timișoara, Februarie 2014

Ing. Lupea Valentin Mircea

LUPEA, VALENTIN MIRCEA

Contribuții la creșterea eficienței neuronului cu ponderi numere complexe și funcție periodică de activare

Teze de doctorat ale UPT, Seria 10, Nr. 43, Editura Politehnica, 2014, 97 pagini, 41 figuri, 21 tabele.

ISSN:1842-7707

ISBN:978-606-554-779-7

Cuvinte cheie: rețea neuronală, neuron artificial, neuron artificial cu ponderi complexe, funcție periodică de activare

Rezumat:

Teza de față abordează tematica rețelelor neuronale axându-se în special pe eficiența neuronului cu ponderi numere complexe și funcție periodică de activare (MVN-P). Pe parcursul acestei teze s-a propus o nouă tehnică de căutare și selecție a sectorului (soluției) corect pentru MVN-P, în timpul învățării. Utilizând această tehnică de căutare s-au obținut rezultate îmbunătățite din punct de vedere al timpului de învățare (respectiv numărul de epoci/iterații). Următorul pas al studiului l-a constituit integrarea MVN-P într-o RN multi-strat (MLMVN-P).

În vederea realizării acestui obiectiv s-a construit o RN cu arhitectură reglabilă, care înglobează pe stratul de ieșire un singur MVN-P. Pe stratul de intrare, respectiv pe straturile ascunse s-au utilizat MVN. Rezultatele obținute sunt încurajatoare, observându-se îmbunătățiri comparativ cu situația utilizării unui singur MVN-P.

În paralel cu integrarea MVN-P într-o MLMVN-P, s-a dorit determinarea periodicității optime a posibilelor sectoare/soluții în momentul învățării MVN-P. S-a observat că un număr prea mare sau prea mic de sectoare influențează negativ procesul de învățare. S-a dorit astfel identificarea unei plaje comune de valori pentru parametrul "periodicitate", valori pentru care procesul de învățare converge mai repede.

Cuprins

| | |
|--|----|
| ABREVIERI..... | 7 |
| LISTA DE FIGURI..... | 8 |
| LISTA DE TABELE..... | 10 |
| 1. INTRODUCERE..... | 11 |
| 1.1. Scurt istoric al temei abordate..... | 11 |
| 1.2. Motivația alegerii temei de cercetare..... | 12 |
| 1.3. Prezentarea pe scurt a conținutului lucrării..... | 12 |
| 1.4. Încadrarea temei în preocupările internaționale, naționale, zonale, ale colectivului de cercetare..... | 13 |
| 1.5. Obiectivele științifice pentru rezolvare în cadrul cercetării științifice..... | 13 |
| 1.6. Metoda de cercetare abordată și metodologia cercetării..... | 13 |
| 1.7. Mulțumiri..... | 14 |
| 2. REȚEAUA NEURONALĂ ȘI NEURONUL ARTIFICIAL..... | 15 |
| 2.1. Rețeaua neuronală – aspecte generale..... | 15 |
| 2.2. Tipuri de rețele neuronale..... | 16 |
| 2.3. Algoritmi de învățare..... | 20 |
| 2.3.1. Backpropagation (corecția erorii)..... | 20 |
| 2.3.2. Învățarea hebbiană..... | 20 |
| 2.3.3. Învățarea Boltzmann..... | 21 |
| 2.3.4. Învățarea competitivă..... | 22 |
| 2.3.5. Învățarea pe baza corecției erorii cu exemple negative..... | 22 |
| 2.4. Neuronul artificial clasic..... | 22 |
| 2.5. Neuronul artificial cu ponderi numere complexe..... | 24 |
| 2.5.1. Aspecte generale..... | 24 |
| 2.5.2. Procesul de învățare..... | 25 |
| 2.5.3. Principalele avantaje/dezavantaje ale folosirii MVN..... | 27 |
| 2.6. Neuronul artificial cu ponderi complexe și funcție periodică de activare..... | 27 |
| 2.6.1. Funcția periodică de activare și algoritmi de învățare..... | 28 |
| 2.7. Concluzii..... | 29 |
| 3. CONTRIBUȚII PRIVIND ÎMBUNĂTĂȚIREA PROCESULUI DE ÎNVĂȚARE A MVN-P..... | 31 |
| 3.1. Soluția propusă..... | 31 |
| 3.2. Strategii de implementare..... | 32 |
| 3.3. Concluzii..... | 33 |
| 4. INTEGRAREA MVN-P ÎNTR-O REȚEA NEURONALĂ..... | 35 |
| 4.1. Soluția propusă..... | 35 |
| 4.2. Strategii de implementare..... | 36 |
| 4.3. Concluzii..... | 38 |
| 5. REZULTATE EXPERIMENTALE..... | 39 |
| 5.1. Rezultatele produse de noua strategie de căutare la nivelul MVN-P..... | 39 |
| 5.1.1. Suma modulo k a n valori..... | 39 |
| 5.1.2. Baza de date IRIS..... | 54 |
| 5.1.3. Baza de date „Breast Cancer Wisconsin (Diagnostic)”..... | 59 |
| 5.2. Rezultatele produse în urma integrării MVN-P într-o rețea neuronală..... | 63 |
| 5.2.1. Suma modulo k a n valori..... | 63 |
| 5.2.2. Baza de date IRIS..... | 69 |
| 5.2.3. Baza de date “Breast Cancer Wisconsin (Diagnostic)”..... | 74 |
| 5.3. Concluzii..... | 77 |
| 6. CONCLUZII..... | 78 |
| 6.1. Îndeplinirea obiectivelor propuse..... | 78 |

6 Cuprins

| | |
|---|----|
| 6.2. Contribuțiile proprii..... | 79 |
| 6.3. Direcții viitoare de cercetare..... | 79 |
| ANEXA 1 – SUMA MODULO K A N VALORI | 80 |
| ANEXA 2 – MLMVN-P REZULTATE COMPLETE PENTRU "SUMA MODULO K A N VALORI" | 82 |
| ANEXA 3 – BAZA DE DATE IRIS | 85 |
| ANEXA 4 – BAZA DE DATE „BREAST CANCER WISCONSIN (DIAGNOSTIC)” | 86 |
| ANEXA 5 – MVN-P SIMULATOR | 87 |
| ANEXA 6 – MLMVN-P SIMULATOR | 90 |
| ANEXA 7 – LISTA CU LUCRĂRILE ȘTIINȚIFICE COMUNICATE/PUBLICATE | 91 |

Abrevieri

| | |
|---|----------------|
| Rețea Neuronală Artificială | RNA |
| Rețea Neuronală | RN |
| Inteligență Artificială | IA |
| Neuron cu ponderi numere complexe (M ulti- V alued N euron) | MVN |
| MVN cu funcție P eriodică de activare | MVN-P |
| Rețelele neuronale cu neuroni cu ponderi numere complexe și funcție periodică de activare (M ulti- L ayered MVN with P eriodic activation function) | MLMVN-P |

Lista de figuri

| | |
|--|----|
| Figura 1: Exemplu de rețea neuronală | 16 |
| Figura 2: Perceptron | 17 |
| Figura 3: Perceptron multi-strat | 18 |
| Figura 4: RN de tip BackPropagation | 18 |
| Figura 5: RN de tip Kohonen | 19 |
| Figura 6: RN de tip Hopfield | 19 |
| Figura 7: Structura algoritmului de BackPropagation | 20 |
| Figura 8: Principiul învățării hebbiane | 21 |
| Figura 9: Principiul învățării Boltzmann | 21 |
| Figura 10: Principiul învățării competitive | 22 |
| Figura 11: Structura neuronului artificial clasic | 23 |
| Figura 12: Interpretarea geometrică a funcției de activare a unui MVN | 25 |
| Figura 13: Interpretarea geometrică a regulii de învățare pentru MVN | 26 |
| Figura 14: Interpretarea geometrică a funcției periodice de activare a MVN-P..... | 28 |
| Figura 15: Interpretarea geometrică a strategiei propuse | 31 |
| Figura 16: Arhitectura unei RN de tip MLMVN-P | 35 |
| Figura 17: Un exemplu de propagare al erorii dinspre stratul de ieșire spre stratul de intrare..... | 37 |
| Figura 18: Variația erorii de învățare pentru a) noua strategie de învățare, respectiv b) strategia Aizenberg considerând cazul $n = 3, k = 4, l = 18$ | 43 |
| Figura 19: Variația erorii de învățare pentru a) noua strategie de învățare, respectiv b) strategia Aizenberg considerând cazul $n = 4, k = 3, l = 11$ | 44 |
| Figura 20: Variația erorii de învățare pentru a) noua strategie de învățare, respectiv b) strategia Aizenberg considerând cazul $n = 5, k = 3, l = 20$ | 45 |
| Figura 21: Variația erorii de învățare pentru a) sensul anti-orar, respectiv b) sensul orar considerând cazul $n = 2, k = 5, l = 8$ | 47 |
| Figura 22: Variația erorii de învățare pentru a) sensul anti-orar, respectiv b) sensul orar considerând cazul $n = 3, k = 4, l = 18$ | 48 |
| Figura 23: Variația erorii de învățare pentru a) sensul anti-orar, respectiv b) sensul orar considerând cazul $n = 5, k = 2, l = 19$ | 49 |
| Figura 24: Rezultate pentru $n = 2$ și $k = 3 \div 6$ | 50 |
| Figura 25: Rezultate pentru $n = 3$ și $k = 3 \div 5$ | 51 |
| Figura 26: Rezultate pentru $n = 4$ și $k = 2 \div 4$ | 52 |
| Figura 27: Rezultate pentru $n = 5$ și $k = 2 \div 4$ | 53 |
| Figura 28: Variația erorii medii pătratice pentru noua strategie de învățare în cazul bazei de date IRIS și a) regula de corecție (1.4), respectiv b) regula de corecție (1.6) | 56 |
| Figura 29: Variația erorii de învățare pentru a) regula de corecție (1.4) și a.1) noua strategie de căutare, respectiv a.2) strategiile lui Aizenberg, b) regula de corecție (1.6) și b.1) noua strategie de căutare, respectiv b.2) strategiile lui Aizenberg | 58 |
| Figura 30: Variația erorii de învățare în cazul bazei de date „Breast Cancer Wisconsin (Diagnostic)”, a) pentru $k = 2$, respectiv b) pentru $k = 3$ | 61 |
| Figura 31: Variația erorii medii pătratice pentru noua strategie de învățare în cazul bazei de date „Breast Cancer Wisconsin (Diagnostic)” | 62 |
| Figura 32: Variația erorii de învățare pentru a) un singur MVN-P, respectiv b) MLMVN-P considerând cazul $n = 3, k = 6, l = 60$ | 67 |
| Figura 33: Variația erorii de învățare pentru a) un singur MVN-P, respectiv b) MLMVN-P considerând cazul $n = 4, k = 4, l = 22$ | 68 |

| | |
|--|----|
| Figura 34: Variația erorii de învățare pentru a) un singur MVN-P, respectiv b) MLMVN-P considerând cazul $n = 5, k = 3, l = 57$ | 68 |
| Figura 35: Variația erorii medii pătratice pentru MLMVN-P în cazul bazei de date IRIS și a) regula de corecție (1.4), respectiv b) regula de corecție (1.6) | 71 |
| Figura 36: Variația erorii de învățare pentru a) regula de corecție (1.4) și a.1) un singur MVN-P, respectiv a.2) MLMVN-P, b) regula de corecție (1.6) și b.1) un singur MVN-P, respectiv b.2) MLMVN-P | 73 |
| Figura 37: Variația erorii medii pătratice pentru noua strategie de învățare în cazul bazei de date „Breast Cancer Wisconsin (Diagnostic)” | 75 |
| Figura 38: Variația erorii de învățare pentru a) un singur MVN-P, respectiv b) MLMVN-P | 76 |
| Figura 39: MVN-P simulator | 87 |
| Figura 40: Parametrii folosiți în timpul învățării MVN-P | 88 |
| Figura 41: MLMVN-P simulator | 90 |

Lista de tabele

| | |
|---|----|
| Tabelul 1: Tipuri de RN..... | 17 |
| Tabelul 2: Funcții de activare | 23 |
| Tabelul 3: Rezultate comparative pentru suma modulo k | 40 |
| Tabelul 4: Analiza statistică descriptivă a populațiilor obținute pentru cele două strategii de căutare comparate | 41 |
| Tabelul 5: Rezultate comparative în cazul folosirii a două sensuri diferite de căutare | 45 |
| Tabelul 6: Analiza statistică descriptivă a populațiilor obținute pentru sensul de parcurgere orar, respectiv anti-orar | 46 |
| Tabelul 7: Seturile de date folosite la analiza influenței parametrului l | 49 |
| Tabelul 8: Baza de date IRIS..... | 54 |
| Tabelul 9: Rezultate comparative pentru baza de date IRIS | 55 |
| Tabelul 10: Baza de date „Breast Cancer Wisconsin (Diagnostic)” | 59 |
| Tabelul 11: Rezultate comparative pentru baza de date „Breast Cancer Wisconsin (Diagnostic)”..... | 60 |
| Tabelul 12: Rezultate comparative pentru suma modulo k a n valori | 63 |
| Tabelul 13: Analiza statistică descriptivă a populațiilor obținute pentru un singur MVN-P, respectiv pentru MLMVN-P..... | 64 |
| Tabelul 14: Rezultate comparative pentru baza de date IRIS..... | 69 |
| Tabelul 15: Analiza statistică descriptivă a populațiilor obținute pentru un singur MVN-P, respectiv pentru MLMVN-P..... | 70 |
| Tabelul 16: Rezultate comparative pentru baza de date „Breast Cancer Wisconsin (Diagnostic)”..... | 74 |
| Tabelul 17: Analiza statistică descriptivă a populațiilor obținute pentru un singur MVN-P, respectiv pentru MLMVN-P..... | 74 |
| Tabelul 18: Suma modulo 4 a 3 valori | 80 |
| Tabelul 19: Rezultate comparative complete pentru suma modulo k a n valori..... | 82 |
| Tabelul 20: Datele statistice sumarizate ale bazei de date IRIS | 85 |
| Tabelul 21: Atributele bazei de date „Breast Cancer Wisconsin (Diagnostic)” [ENG] | 86 |

1. INTRODUCERE

Domeniul de cercetare abordat în cadrul lucrării de față îl reprezintă neuronul cu ponderi numere complexe și funcție periodică de activare (MVN-P) cât și rețelele neuronale bazate pe acesta (MLMVN-P). Neuronul cu ponderi numere complexe (MVN) oferă soluția rezolvării acelor probleme neliniare care în mod normal nu pot fi rezolvate folosind un singur neuron artificial clasic (cu ponderi numere reale), fiind necesară folosirea unei rețele neuronale artificiale (RNA) de cele mai multe ori cu o arhitectură destul de complexă. Cu o structură relativ simplă, asemănătoare cu cea a neuronului artificial clasic, ceea ce aduce nou MVN este faptul că ieșirile sale se situează pe cercul trigonometric (împărțit pe sectoare egale ca dimensiune), iar corecția ponderilor se reduce la translatarea ieșirii dintr-un sector greșit într-unul corect (corecție liniară). Încă de la primele experimente acest tip de neuron și-a dovedit eficiența în rezolvarea unor probleme clasice cum ar fi: SAU-exclusiv, Wisconsin Breast Cancer (diagnosticare), funcții neliniare multi-prag, etc. La scurt timp după aceasta s-a dorit îmbunătățirea eficienței MVN prin integrarea acestuia într-o RNA.

1.1. Scurt istoric al temei abordate

Cu o istorie destul de veche, datând din 1880, [1] și [2], RNA a reprezentat încercarea omului de a imita funcționarea creierului uman. În vederea realizării acestui deziderat s-a plecat de la modelul biologic al neuronului și s-a încercat copierea caracteristicilor acestuia cum ar fi: paralelismul prelucrării informației, toleranța la erori și capacitatea de adaptare la situații noi. În 1949, Donald Hebb a dezvoltat o teorie conform căreia legătura dintre doi neuroni devine mai puternică cu cât este mai des folosită [3], punând astfel bazele învățării hebbiane. În 1959, Bernard Widrow și Marcian Hoff, de la Universitatea Stanford, au dezvoltat sistemele denumite "ADALINE" (capabile să recunoască modele de biți) și "MADALINE" (prima RNA cu aplicabilitate reală) [11]. Cu toate că RNA s-a dezvoltat în continuare prin crearea unui algoritm de învățare bazat pe corecția erorii, propus de către aceiași autori în 1962, în scurt timp s-a ajuns la concluzia că RNA nu va putea simula funcționarea complexă a creierului uman. Acest aspect, corelat cu frica unora legată de posibilitatea ca mașinile să ajungă să ia decizii singure, au condus la trecerea RNA într-un con de umbră. Abia după zece ani, în 1972, Kohonen și Anderson [6], [7] au deschis noi orizonturi pentru RNA punând bazele primei RNA multi-strat, apărută în 1975. Cu toate acestea abia în 1982 interesul pentru RNA a crescut din nou. În această perioadă o RNA hibridă (RNA multi-strat, fiecare strat folosind o strategie diferită de rezolvare a problemei) a fost propusă de către Reilly și Cooper. În 1986, modelul propus de Widrow-Hoff pentru corecția erorii a fost extins la nivelul RNA multi-strat, luând astfel naștere RNA "back-propagation" (propagarea erorii de la stratul de ieșire spre cel de intrare). În 1992, Aizenberg a introdus un nou concept, de data aceasta la nivelul neuronului artificial și nu al RNA, și anume neuronul cu ponderi numere complexe [12]. Ce aducea nou acest tip de neuron era faptul că folosea planul numerelor complexe (sau cercul trigonometric) permițând astfel rezolvarea unor probleme neliniare (ex. SAU-exclusiv, suma modulo k a n

valori) folosind un singur MVN, lucru imposibil de realizat cu un singur neuron artificial clasic. În anii ce au urmat (1992 - prezent) Aizenberg a propus și dezvoltat diverși algoritmi de învățare pentru MVN, îmbunătățind performanțele acestuia. De asemenea, MVN a fost integrat într-o RNA, cu rezultate mai mult decât încurajatoare [23], [34], [36].

1.2. Motivația alegerii temei de cercetare

Eficiența crescută dar și caracterul inovativ al MVN au reprezentat factorii principali în alegerea acestuia ca temă de studiu a lucrării de față. Cu toate că unele concepte teoretice au părut la început mai dificil de înțeles, pe de-o parte datorită diferențelor față de neuronul artificial clasic, iar pe de alta datorită numărului relativ mic de articole publicate pe această temă, treptat avantajele folosirii MVN au făcut ca acest efort al înțelegerii MVN să merite. Pe lângă factorii motivaționali enumerați mai sus ar mai fi și aria largă de aplicabilitate a RNA: în medicină (e.x. diagnosticarea cancerului încă din fazele incipiente), în sistemele de predicție (ex. bursa de valori), în sistemele de clasificare, la preluarea imaginilor (e.x. filtrarea zgomotului, evidențierea muchiilor), etc. În toate aceste domenii, RNA oferă o soluție alternativă, demnă de luat în calcul, la algoritmii tradiționali de calcul (e.x. Backtracking, algoritmi genetici). Un alt aspect important, în orice domeniu de cercetare este continuitatea oferită de subiectul abordat. În cazul de față noutatea adusă de MVN asigură posibilitatea unor dezvoltări ulterioare destul de largi, care să vizeze, în primul rând, aplicarea și adaptarea unor concepte tradiționale la specificitatea MVN; un exemplu în acest caz ar putea fi aplicarea diverselor tipuri de învățare nesupervizate, învățarea cu exemple negative, etc.

1.3. Prezentarea pe scurt a conținutului lucrării

Lucrarea de față este structurată în patru părți principale: partea generală (care prezintă concepte teoretice fundamentale), o parte specifică (contribuțiile autorului), partea experimentală (prezentarea rezultatelor obținute) și o parte finală de concluzii și posibile viitoare direcții de dezvoltare. În continuare, voi prezenta pe scurt conținutul fiecărei părți în parte.

Prima parte a studiului prezintă conceptele teoretice fundamentale din domeniul RNA. Aici vor fi prezentate aspectele teoretice generale, pe scurt ce reprezintă o RNA, caracteristicile neuronului artificial clasic (cu ponderi numere reale), caracteristicile MVN, evoluția acestuia înspre MVN cu funcție periodică de activare (MVN-P), caracteristicile MVN-P.

A doua parte a studiului se va axa pe prezentarea contribuțiilor autorului în domeniul abordat. Se va prezenta o nouă strategie de selecție a sectorului dorit pentru MVN-P cât și o soluție de integrare a MVN-P într-o RNA. RNA propusă se bazează pe o arhitectură multi-strat clasică (1 strat intrare, n straturi ascunse și 1 strat ieșire) având ca și algoritm de învățare algoritmul de "back-propagation" (propagarea erorii dinspre ieșire spre intrare) adaptat la caracteristicile MVN, respectiv MVN-P.

Partea experimentală, a treia parte a studiului de față, va prezenta rezultatele obținute în cursul validării noii strategii de selecție, cât și a RNA cu MVN-P. În vederea validării strategiei de selecție propuse rezultatele obținute au fost comparate cu cele mai bune rezultate indicate în literatură până la momentul de

față, în vreme ce pentru analiza eficienței RNA cu MVN-P, rezultatele obținute pentru un singur MVN-P, respectiv pentru RNA cu MVN-P, vor fi comparate.

Cea de-a patra parte și ultima de altfel, prezintă concluziile finale ale autorului obținute în urma părții experimentale. Tot aici se vor evidenția posibile viitoare direcții de dezvoltare observate cât și limitări sau obstacole întâlnite pe parcursul studiului întreprins.

1.4. Încadrarea temei în preocupările internaționale, naționale, zonale, ale colectivului de cercetare

Programul doctoral în cadrul căruia s-a desfășurat studiul de față face parte din programul POSDRU/CPP107/DMI1.5/S/77265 (din 2010) al Ministerului Muncii, Familiei și Protecției Sociale al Guvernului României, co-finanțat de Fondul Social European – Investind în oameni. Studiul de față s-a desfășurat în cadrul Universității "Politehnica" din Timișoara, Facultatea de "Automatică și Calculatoare", specialitatea Calculatoare, sub directa îndrumare a Prof.Dr.Ing. Ștefan Holban. Activitatea didactică a domnului Prof.Dr.Ing. Ștefan Holban se încadrează în domeniul IA, câteva dintre materiile predate de dumealui fiind enumerate în continuare: Bazele inteligenței artificiale, Neural networks, Pattern recognition. Pe lângă aceste subiecte mai putem enumera și Modelare și simulare, Explorarea datelor sau Evaluarea performanțelor sistemelor de calcul.

1.5. Obiectivele științifice pentru rezolvare în cadrul cercetării științifice

Obiectivul principal al cercetării științifice a urmărit studierea și creșterea eficienței MVN-P. În vederea atingerii acestui obiectiv s-a urmărit dezvoltarea unei strategii proprii de selecție a sectorului corect, pe parcursul învățării, pentru MVN-P cât și dezvoltarea unei RNA ajustabile în vederea integrării MVN-P.

Principalele cerințe care au fost luate în calcul în etapa de dezvoltare a noii strategii de selecție au fost obținerea unui timp mai redus de învățare cu păstrarea unei erori medii acceptabile (în jurul valorii de 5-10%). Aceste două aspecte sunt relevante în momentul integrării MVN-P într-o RNA unde, pe de-o parte, durata de învățare crește odată cu creșterea complexității rețelei, dar pe de altă parte eroarea medie scade.

1.6. Metoda de cercetare abordată și metodologia cercetării

În vederea analizei eficienței MVN-P, respectiv a MLMVN-P s-au folosit atât metode analitice cât și experimentale. S-a plecat de la o analiză amănunțită a algoritmilor prezenți în literatura de specialitate, insistându-se asupra punctelor forte cât și a posibilelor îmbunătățiri ce pot fi aduse. Mai apoi s-au rulat teste experimentale intense în vederea validării noii strategii de învățare. Pentru realizarea acestor teste s-a dezvoltat un simulator în Microsoft Visual C# care imită caracteristicile MVN/MVN-P. Pasul următor l-a reprezentat dezvoltarea unei RNA configurabile, realizată de asemenea în Microsoft Visual C#, care să încorporeze

MVN pe straturile de intrare și cele ascunse, respectiv a unui singur MVN-P pe stratul de ieșire.

1.7. Mulțumiri

În primul rând aș dori să-i mulțumesc domnului Prof.Dr.Ing Ștefan Holban pentru îndrumarea și suportul oferit pe parcursul celor trei ani de Doctorat.

În al doilea rând aș dori să mulțumesc Universității "Politehnica" din Timișoara pentru șansa oferită de a urma o școală doctorală și pentru suportul financiar oferit pe parcursul acesteia.

În al treilea rând, dar nu în cele din urmă, aș dori să mulțumesc programului POSRDU pentru suportul financiar oferit în această perioadă.

2. REȚEAUA NEURONALĂ ȘI NEURONUL ARTIFICIAL

Cu o istorie relativ veche datând din 1880 [1], [2], RN, respectiv neuronul artificial au reprezentat și reprezintă încercarea omului de a simula funcționarea creierului uman. Variînd de la arhitecturi simple (un singur neuron artificial) până la arhitecturi complexe, cu număr crescut de straturi și neuroni, RN și-au dovedit eficiența în rezolvarea unor probleme de clasificare, predicție, diagnosticare sau identificare.

De-a lungul timpului, odată cu dezvoltarea lor, RN au devenit alternative viabile ale algoritmilor de calcul clasici ca Backtraking, algoritmi genetici sau metode euristice.

Datorită puterii mari de calcul, cât și datorită eficienței crescute, RN au înlocuit cu succes vechile structuri de calcul tradiționale, în domenii ca diagnosticare medicală, prelucrare de imagini sau în sistemele de predicție.

În continuare, sunt prezentate, pe scurt, principalele caracteristici ale RN, respectiv ale neuronului artificial.

2.1. Rețeaua neuronală – aspecte generale

Ca și idee generală, o RN reprezintă modelul matematic simplificat al funcționării creierului uman.

O RN încearcă să mimeze procesele complexe care au loc la nivelul creierului uman în momentul învățării.

Ca și construcție, o RN este formată din mai multe straturi, iar pe fiecare strat în parte se găsesc un anumit număr de neuroni artificiali. Legăturile dintre neuroni, stabilite în timpul învățării, sunt cele ce determină eficiența unei RN.

În Figura 1 este prezentată o structură de RN în care se pot identifica principalele componente constructive:

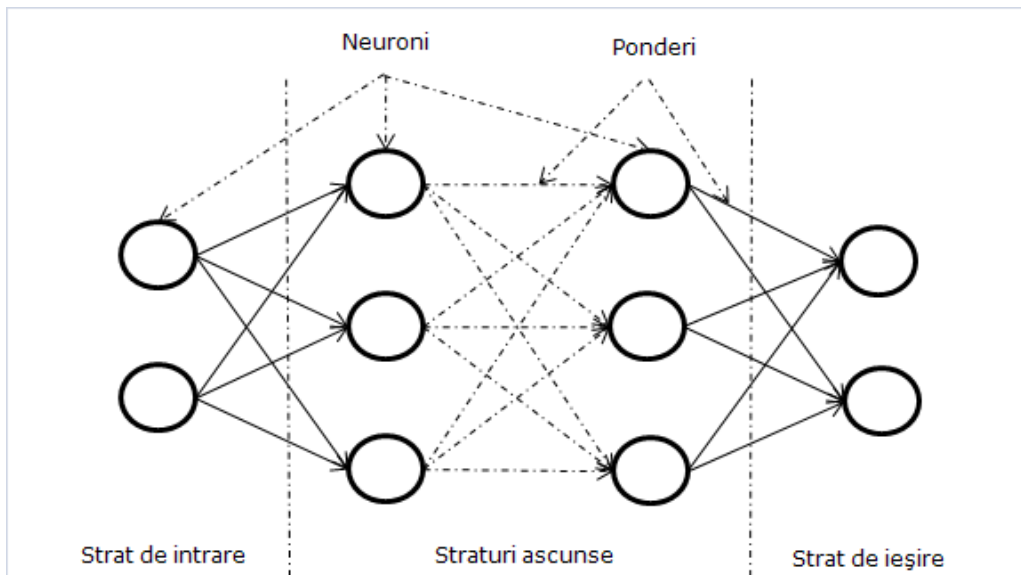


Figura 1: Exemplu de rețea neuronală

Putem identifica un strat de intrare, un strat de ieșire și mai multe straturi intermediare, numite straturi ascunse. Numărul de neuroni de pe stratul de intrare este numărul de parametri de intrare al setului de date, iar numărul de neuroni de pe stratul de ieșire este egal cu numărul de parametri de ieșire. Aceste două valori sunt cunoscute în momentul începerii învățării. Pentru stratul sau straturile intermediare, numărul de neuroni nu este cunoscut la începutul învățării și poate varia în funcție de specificitatea setului de date folosit. În cele mai multe cazuri s-a observat că un număr de neuroni ascunși, egal cu radicalul sumei pătratelor neuronilor de intrare și al neuronilor de ieșire ($\sqrt{\text{intrări}^2 + \text{ieșiri}^2}$), este suficient pentru rezolvarea majorității problemelor.

Există RN de tip feedforward, cât și RN cu feedback. În cazul RN cu feedforward eroarea de pe stratul de ieșire se propagă înapoi spre stratul de intrare (de unde și numele) în vreme ce RN cu feedback sunt caracterizate de existența unui semnal de reacție din partea neuronilor de ordin superior pentru cei de ordin inferior sau chiar pentru propriile lor intrări.

Metoda de învățare poate varia de asemenea. Învățarea supravegheată este acea învățare în care se cunoaște ieșirea, iar pe baza acesteia se corectează treptat eroarea RN, iar învățarea nesupravegheată este aceea în care ieșirea nu se cunoaște, iar corecția se realizează pe baza legăturilor dintre neuroni (e.x. învățarea hebbiană – legea lui Hebb [3]).

2.2. Tipuri de rețele neuronale

În literatură sunt prezentate un număr variat de RN fiecare cu caracteristicile sale specifice. În Tabelul 1 sunt prezentate câteva astfel de RN:

Tabelul 1: Tipuri de RN

| Nume | Arhitectură | Caracteristici |
|---------------------------|---|---|
| Perceptron | 1 singur strat (cel de intrare) | Tip: feedforward Valori de intrare: binare sau reale Metoda de antrenare: supravegheată Utilitate: operații logice simple, clasificări |
| Perceptron Multi-strat | 1 strat de intrare 1 sau mai multe straturi ascunse 1 strat de ieșire | Tip: feedforward Valori de intrare: binare sau reale Metoda de antrenare: supravegheată Algoritm de antrenare: backpropagation Utilitate: operații logice complexe, clasificare |
| RN de tip BackPropagation | 1 strat de intrare 1 sau mai multe straturi ascunse 1 strat de ieșire | Tip: feedforward Valori de intrare: reale Metoda de antrenare: supravegheată Algoritm de antrenare: backpropagation Utilitate: operații logice complexe, clasificare tipare, analiză vocală |
| Hopfield | 1 matrice | Tip: feedback Valori de intrare: binare Metoda de antrenare: nesupravegheată Algoritm de antrenare: regula delta Utilitate: recunoaștere tipare, probleme de optimizare |
| Kohonen | 1 strat de intrare 1 mulțime de neuroni | Tip: feedforward /feedback Valori de intrare: binare sau reale Metoda de antrenare: nesupravegheată Algoritm de antrenare: autoorganizarea Utilitate: clasificarea tiparelor, problemele legate de optimizare, simulări |

În continuare sunt prezentate, pe scurt, câteva dintre cele mai cunoscute arhitecturi de RN.

Introdus prima dată de către F.Rosenblatt în 1958 [4], Perceptronul (Figura 2) este o RN foarte simplă, care are ca intrări numere binare sau reale. Procesul de antrenare este unul supravegheat și bazat pe corecția erorii dintre valoarea obținută și cea dorită. Acest tip de RN este eficient în special în cazul unor probleme logice simple ca SAU logic, ȘI logic, etc dar nu poate fi aplicat în cazul unor probleme mai complexe ca SAU-EXCLUSIV.

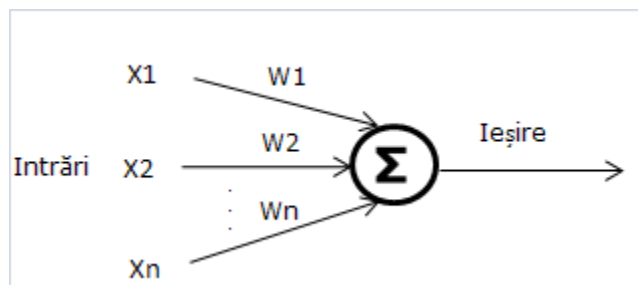


Figura 2: Perceptron

18 Rețeaua neuronală și neuronul artificial - 2

Perceptronul multi-strat (Figura 3) a fost introdus pentru prima dată de către M.Minsky și S.Papert în 1969 [5] și reprezintă o extensie a Perceptronului simplu prin inserarea unui număr de straturi ascunde între stratul de intrare și cel de ieșire. Odată cu creșterea complexității structurale a crescut și eficiența învățării putând fi aplicat și în cazul unor probleme logice mai complexe cum ar fi SAU-EXCLUSIV.

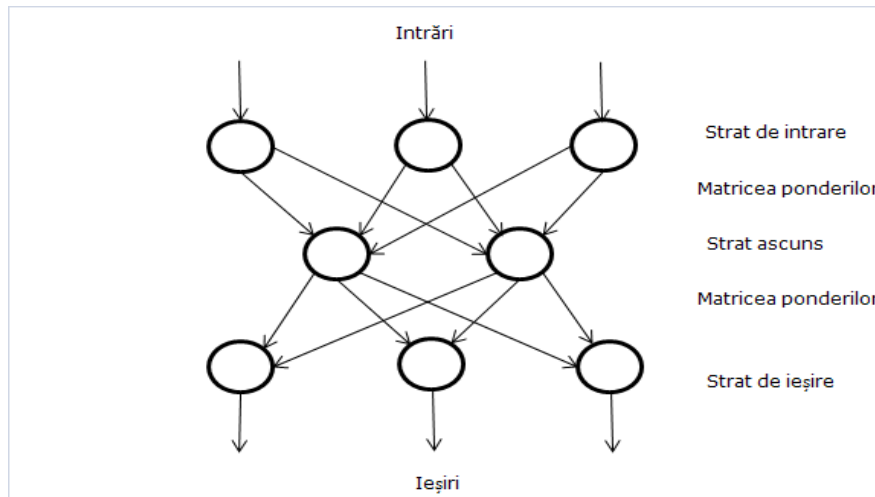


Figura 3: Perceptron multi-strat

Introdusă pentru prima dată de către G.E.Hinton, E.Rumelhart și R.J.Wiliams în 1986 [10], RN de tip BackPropagation (Figura 4) este asemănătoare cu RN de tip Perceptron multi-strat diferind algoritmul de antrenare, care în cazul acesteia este bazat pe propagarea erorii dinspre ieșire spre intrare.

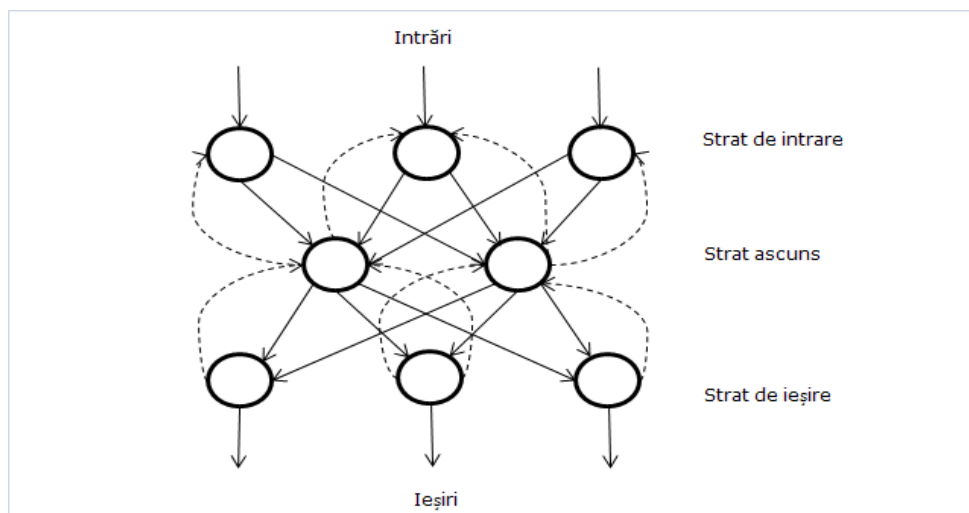


Figura 4: RN de tip BackPropagation

RN de tip Kohonen (Figura 5) au fost prima dată propuse de către Teuvo Kohonen în 1982 [8] și sunt formate dintr-un strat de intrare și un strat de ieșire, fără nici un strat intermediar (ascuns). De asemenea, toți neuronii sunt interconectați între ei, formând o așa numită matrice de ponderi.

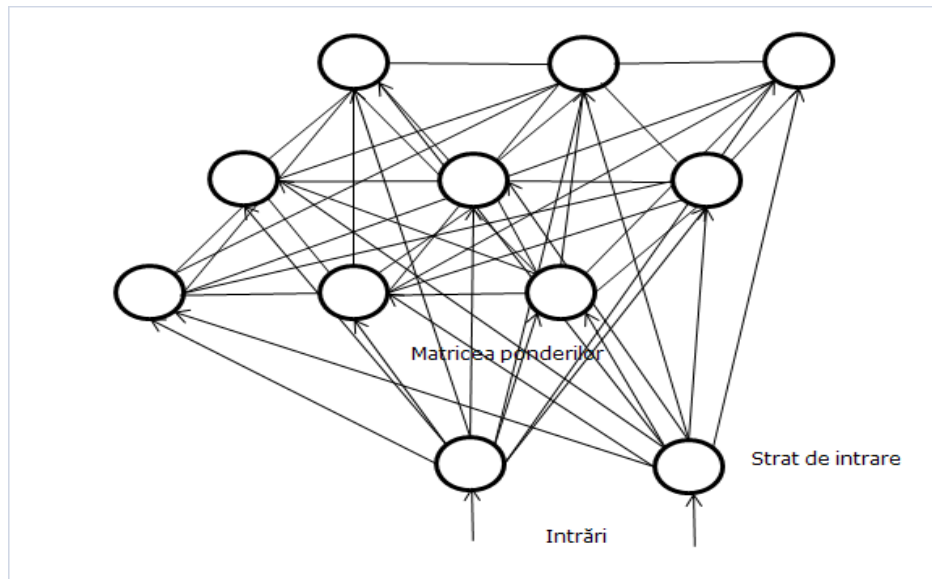


Figura 5: RN de tip Kohonen

RN de tip Hopfield (Figura 6) au fost prima dată introduse de către J.J.Hopfield în 1982 [9] și sunt compuse dintr-un set de neuroni fără diferențierea acestora pe straturi de ieșire, intrare sau ascunde. Fiecare neuron din acest set este conectat cu fiecare alt neuron, iar ponderile sunt simetrice: $w_{ij} = w_{ji}$ unde w_{ij} reprezintă ponderea legăturii neuronului i cu neuronul j .

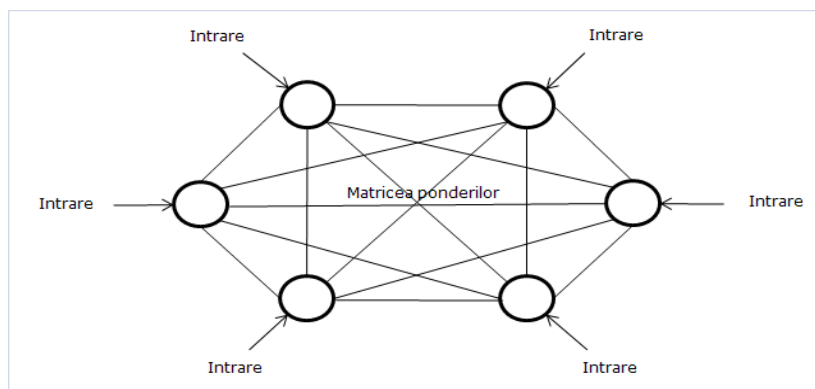


Figura 6: RN de tip Hopfield

2.3. Algoritmi de învățare

În continuare, sunt prezentați, pe scurt, algoritmi de învățare folosiți cu diversele arhitecturi de RN, prezentate în subcapitolul anterior.

2.3.1. Backpropagation (corecția erorii)

Algoritmul de Backpropagation reprezintă o învățare de tipul supervizată (Figura 7), în care se compară valoarea de ieșire obținută cu valoarea dorită; pe baza diferenței dintre cele două valori, ponderile neuronilor sunt modificate în vederea minimizării erorii de învățare.

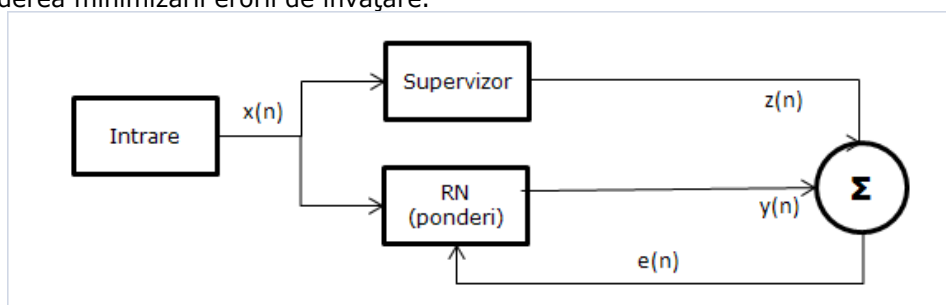


Figura 7: Structura algoritmului de BackPropagation

Notăm cu:

- $x_n = (x_1, x_2, \dots, x_i)$, vectorul intrărilor
- $y_n = (y_1, y_2, \dots, y_i)$, vectorul ieșirilor
- $z_n = (z_1, z_2, \dots, z_i)$, vectorul valorilor dorite

În timpul învățării diferența dintre vectorul $z(n)$ și vectorul $y(n)$ denotă vectorul eroare $e(n)$ obținut pentru vectorul de intrare $x(n)$. Învățarea constă în minimizarea vectorului eroare $e(n)$.

2.3.2. Învățarea hebbiană

Învățarea hebbiană are la bază un principiu enunțat de Hebb [3] și anume: "Când un axon al celulei nervoase A este suficient de aproape de faza de excitare a unei celule nervoase B, și în mod repetat sau persistent ia parte la activarea sa, un anumit proces de creștere sau de modificare metabolică are loc într-una sau în ambele celule nervoase, astfel încât eficiența celulei nervoase A este mărită din punct de vedere al contribuției la activarea celulei B".

Acest principiu se transpune în cazul RN astfel: dacă doi neuroni sunt activi în același timp atunci ponderea legăturii comune crește, altfel ea scade până la 0 dacă situația se păstrează (Figura 8).

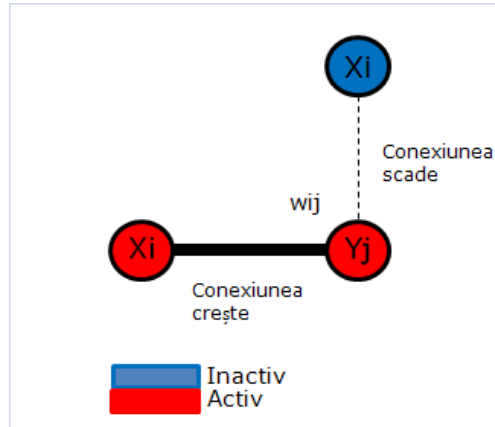


Figura 8: Principiul învățării hebbiane

Notăm cu:

- X_i , neuronul de intrare
- Y_j , neuronul de ieșire
- w_{ij} , ponderea celor doi neuroni

2.3.3. Învățarea Boltzmann

În cadrul acestui tip de învățare, RN este reprezentată printr-o structură de neuroni activi, cu valoarea (+1), respectiv neuroni inactivi, cu valoarea (-1) (Figura 9). Întreaga RN este caracterizată de o energie E :

$$E = -\frac{1}{2} \sum_{i,j \neq i} w_{ij} s_j s_i \quad (1.1)$$

unde s_i reprezintă starea neuronului i , s_j reprezintă starea neuronului j , iar w_{ij} reprezintă ponderea neuronilor i și j .

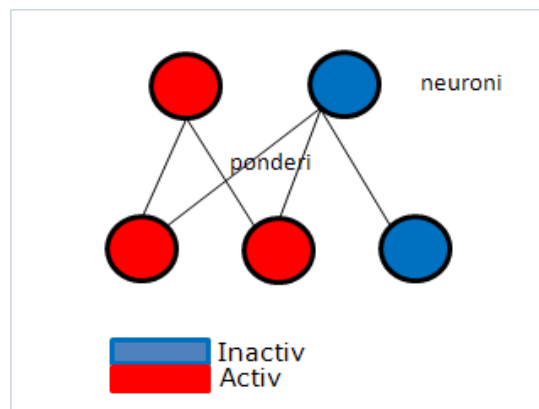


Figura 9: Principiul învățării Boltzmann

În timpul învățării modificarea ponderilor este realizată în direcția atingerii unui punct de echilibru termic.

2.3.4. Învățarea competitivă

Acest tip de învățare se bazează pe competiția neuronilor de pe stratul de ieșire (Figura 10). La un moment dat, un singur neuron de ieșire poate fi activ, încercându-se astfel specializarea fiecărui neuron în recunoașterea anumitor trăsături ale datelor de intrare.

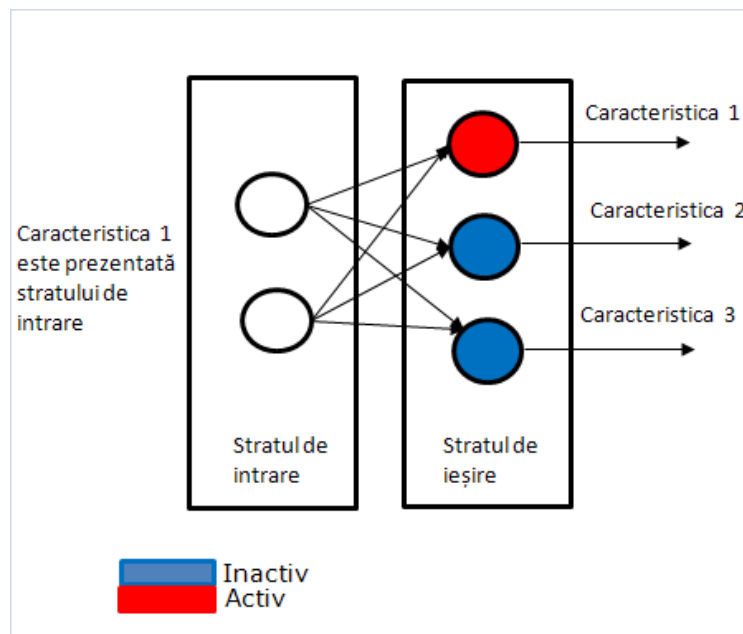


Figura 10: Principiul învățării competitive

2.3.5. Învățarea pe baza corecției erorii cu exemple negative

Această învățare propusă în [45] este o învățare supervizată și are la bază tehnica de învățare pe baza corecției erorii.

Această tehnică de antrenare folosește un procent exact de exemple negative în setul de antrenare [46] și utilizează un neuron suplimentar pe stratul de ieșire pentru a semnaliza tiparele negative de la intrare [47].

2.4. Neuronul artificial clasic

Neuronul artificial reprezintă modelul matematic al neuronului biologic, constituind elementul fundamental al unei RN. În Figura 11 este prezentat modelul matematic simplificat al neuronului artificial.

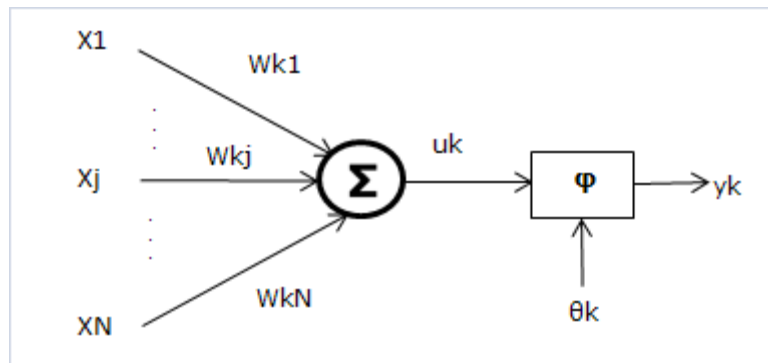


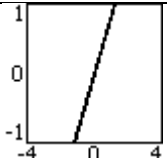
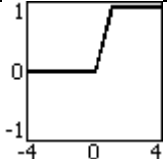

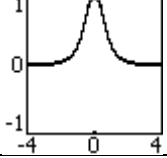
Figura 11: Structura neuronului artificial clasic

Se pot identifica următoarele elemente constructive ale neuronului artificial: un număr N de intrări notate $x_1 \dots x_N$ (valori reale continue) cu ponderile corespunzătoare $w_{k1} \dots w_{kN}$ (valori reale continue), un sumator Σ având valoarea de ieșire $u_k = x_1 \cdot w_{k1} + \dots + x_j \cdot w_{kj} + \dots + x_N \cdot w_{kN}$ și o funcție de activare φ cu factor bias θ_k care respectă relația $f \varphi = y_k$, unde y_k reprezintă ieșirea neuronului artificial.

Există mai multe funcții de activare pentru neuronul artificial, câteva dintre ele fiind prezentate în Tabelul 2. De asemenea, în [24] sau [37], se prezintă comparativ, eficiența diverselor funcții de activare.

Tabelul 2: Funcții de activare

| Nume | Funcție | Reprezentare grafică |
|-----------------------------|--|----------------------|
| Funcția prag | $\varphi v = \begin{cases} 1, v \geq 0 \\ 0, v < 0 \end{cases}$ | |
| Funcția semn | $\varphi v = \begin{cases} 1, v \geq 0 \\ -1, v < 0 \end{cases}$ | |
| Funcția sigmoid | $\varphi v = \frac{1}{1 + e^{-av}}, a = const.$ | |
| Funcția tangent hiperbolică | $\varphi v = \frac{1 - e^{-2v}}{1 + e^{-2v}}$ | |

| | | |
|--|---|--|
| Funcția lineară | $\varphi v = v$ |  |
| Funcția lineară cu saturație | $\varphi v = \begin{cases} 0, v < 0 \\ v, 0 \leq v \leq 1 \\ 1, v > 1 \end{cases}$ |  |
| Funcția lineară cu saturație simetrică | $\varphi v = \begin{cases} -1, v < 0 \\ v, 0 \leq v \leq 1 \\ 1, v > 1 \end{cases}$ |  |
| Funcția gaussiană | $\varphi v = e^{-v^2}$ |  |

În timpul învățării, ponderile corespunzătoare intrărilor unui neuron variază continuu până când acestea ating un punct de stabilitate. În acel moment procesul de învățare poate fi considerat încheiat.

2.5. Neuronul artificial cu ponderi numere complexe

2.5.1. Aspecte generale

Neuronul artificial cu ponderi numere complexe (în Engleză Multi-Valued Neuron - MVN) a fost introdus și detaliat în [12], [13], [14] și [15]. Noutatea introdusă, comparativ cu neuronul artificial clasic, o constituie utilizarea numerelor complexe atât pentru ponderile sale cât și în procesul de învățare (funcții de activare, valori de ieșire, etc.). Datorită utilizării numerelor complexe, ieșirile MVN sunt plasate pe cercul trigonometric. În Figura 12 este ilustrată reprezentarea geometrică a MVN [13].

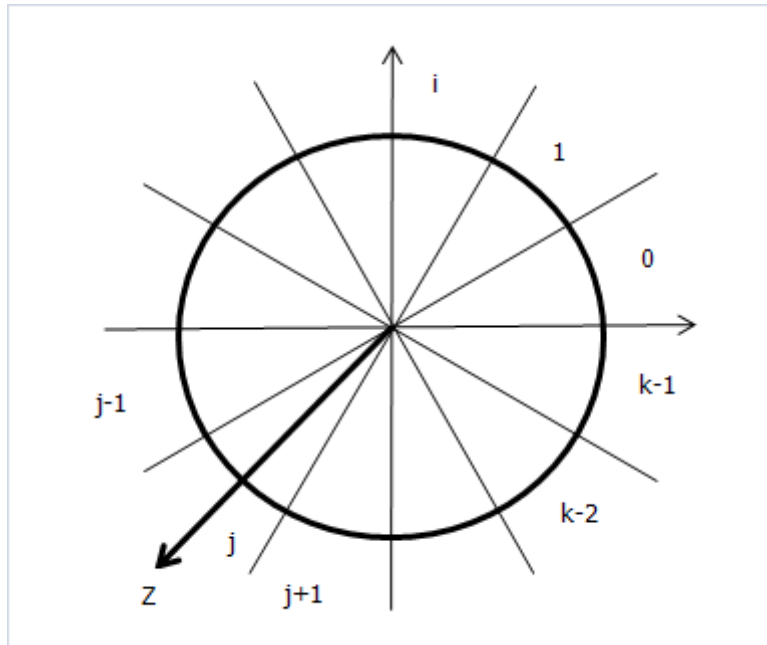


Figura 12: Interpretarea geometrică a funcției de activare a unui MVN

Se poate observa, din figura de mai sus, un număr n de sectoare (egale ca dimensiune), reprezentând numărul total de clase/soluții ale problemei studiate, în care ieșirea Z a MVN se poate localiza.

Z este de forma:

$$Z = x_1 \cdot w_{k1} + \dots + x_j \cdot w_{kj} + \dots + x_N \cdot w_{kN} \quad (1.2)$$

, cu precizarea că valorile x_i, w_{ki} sunt de forma $a + b \cdot i$. Datorită acestor caracteristici procesul de învățare, respectiv corecția erorilor este simplificată, față de cea a neuronului clasic, și se reduce la translatarea ieșirii MVN dintr-un sector (e.x. J) greșit într-unul corect (e.x. $J-1$); acest proces este prezentat în subcapitolul următor. Ca și funcție de activare se folosește ecuația (1.3) [13]:

$$P(z) = \exp(i2\pi j/k) \quad (1.3)$$

, unde $2\pi j/k \leq \arg(Z) < 2\pi(j+1)/k$.

2.5.2. Procesul de învățare

În Figura 13 este ilustrată reprezentarea geometrică a procesului de învățare a MVN[13].

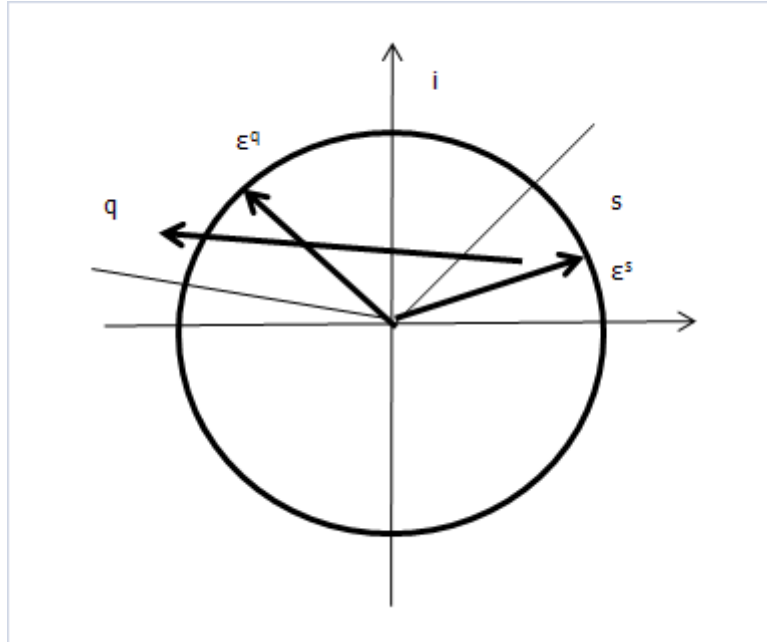


Figura 13: Interpretarea geometrică a regulii de învățare pentru MVN

Presupunând că în timpul învățării ieșirea MVN se situează în sectorul s , se analizează corectitudinea acestuia prin comparație cu sectorul dorit, q . Dacă cele două sectoare sunt identice nu este nevoie de nici o corecție suplimentară și etapa curentă de învățare se încheie. În cazul în care cele două sectoare nu coincid o corecție este necesară. Această corecție constă în aplicarea unui algoritm de căutare și selecție prin care ieșirea MVN este translatată din sectorul s (sectorul curent) în sectorul q (sectorul corect). Pe baza acestei translatare, ponderile MVN sunt modificate și etapa curentă de învățare se încheie. Ecuațiile (1.4) și (1.5), prezintă două reguli posibile de modificare a ponderilor în timpul învățării [19], [29]:

$$W_{m+1} = W_m + \frac{C_m}{N+1} (\varepsilon^q - \varepsilon^s) \bar{X}, \quad (1.4)$$

$$W_{m+1} = W_m + \frac{C_m}{N+1} \frac{Z_r}{Z_r} (\varepsilon^q - \varepsilon^s) \bar{X}, \quad (1.5)$$

unde $W_m = (w_{k,1}, \dots, w_{k,j}, \dots, w_{k,N})$ reprezintă vectorul ponderilor, W_{m+1} este vectorul ponderilor după corecție, C_m o constantă de învățare de obicei egală cu 1, N numărul de intrări ale MVN, Z_r este dat de ecuația (1.2), ε^q și ε^s valoarea dorită a ieșirii MVN, respectiv valoarea obținută a MVN, $\bar{X} = (x_1, \dots, x_j, \dots, x_N)$ vectorul conjugat al intrărilor, iar m este etapa (iterația) curentă de învățare.

Ecuațiile (1.6) și (1.7), pot fi de asemenea folosite pentru probleme cu neliniaritate crescută. Ecuațiile (1.6), (1.7) reprezintă modificări ale ecuațiilor (1.4) respectiv (1.5), pe baza transformării $\varepsilon^s \rightarrow \frac{Z_r}{Z_r}$; raportul $\frac{Z_r}{Z_r}$ reprezintă proiecția Z_r pe cercul unitar [38].

$$W_{m+1} = W_m + \frac{c_m}{N+1} \left(e^q - \frac{z_r}{z_r} \right) \bar{X}, \quad (1.6)$$

$$W_{m+1} = W_m + \frac{c_m}{N+1} \frac{1}{z_r} \left(e^q - \frac{z_r}{z_r} \right) \bar{X}, \quad (1.7)$$

2.5.3. Principalele avantaje/dezavantaje ale folosirii MVN

În cazul folosirii MVN există avantaje și dezavantaje față de folosirea neuronului artificial clasic (cu ponderi numere reale).

Principalele avantaje revin din folosirea numerelor complexe și sunt prezentate în continuare.

Un prim avantaj semnificativ îl constituie regula de corecție a erorilor care în cazul MVN se reduce la o transformare liniară, proporțională cu diferența dintre ieșirea obținută și cea dorită (vezi ecuația (1.4)). Învățarea devine astfel o parcurgere a cercului trigonometric, mai mult orice direcție de parcurgere fiind corectă.

Un alt avantaj îl constituie eficiența crescută comparativ cu cea a neuronului artificial clasic. Astfel probleme neliniare tradiționale ca SAU-EXCLUSIV [35], suma modulo n a k numere [28], [35], etc. pot fi rezolvate utilizând un singur MVN, lucru imposibil de realizat folosind un singur neuron artificial clasic, reducându-se astfel complexitatea RN. De asemenea MVN, respectiv MLMVN și-au demonstrat eficiența în problemele de predicție [43], [44], clasificare [21], [26], [31], [32], [42], cât și în cazul prelucrării de imagini ca recunoașterea de forme [16], [17], [20], reducerea zgomotului din imagini [25], [27], [33], prelucrarea imaginilor alb-negru (cu diverse nuanțe de gri) [18], [22], domenii importante în cadru sistemelor inteligente.

Ca și dezavantaje putem enumera nevoia unor cunoștințe matematice avansate de trigonometrie, numere complexe, respectiv funcții neliniare.

Punând în balanță principalele avantaje și dezavantaje ale folosirii MVN putem concluziona că acesta reprezintă o soluție alternativă atrăgătoare și demnă de luat în calcul la neuronul artificial clasic.

2.6. Neuronul artificial cu ponderi complexe și funcție periodică de activare

Neuronul artificial cu ponderi numere complexe și funcție periodică de activare (MVN-P) a fost introdus pentru prima dată de Aizenberg în [38] și detaliat cu rezultatele în [39], [40] și [41]. Acesta reprezintă o evoluție a MVN prin introducerea ideii de periodicitate a posibilelor soluții. Astfel dacă în cazul MVN cercul trigonometric a fost divizat în k sectoare egale (reprezentând numărul posibilelor ieșiri), în cazul MVN-P numărul sectoarelor a fost extins la $m = k \cdot l$. În figura 14 este prezentată această transformare [38].

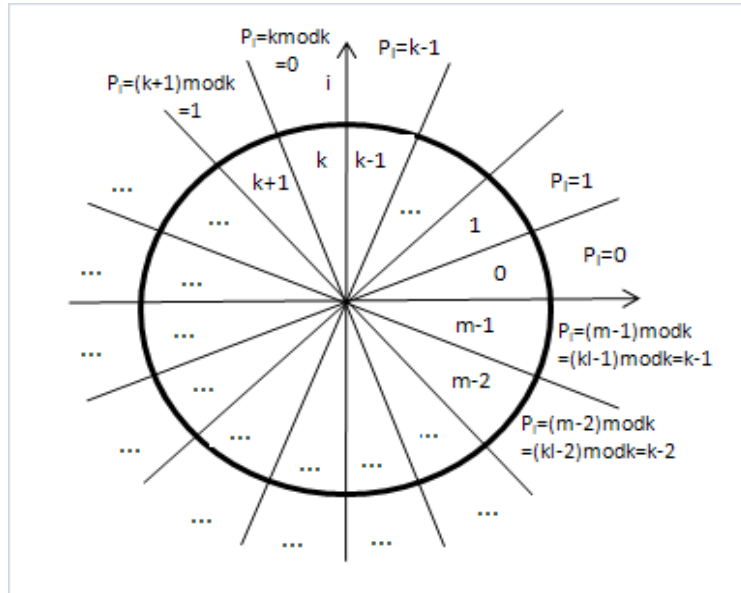


Figura 14: Interpretarea geometrică a funcției periodice de activare a MVN-P

Factorul l indică periodicitatea soluțiilor având astfel o influență importantă în procesul de învățare, după cum vom observa în partea experimentală a acestei lucrări.

În continuare se prezintă în detaliu funcția periodică de activare și conceptele matematice care stau la baza acestuia așa cum sunt prezentate în literatură.

2.6.1. Funcția periodică de activare și algoritmi de învățare

În ecuația (1.8) se prezintă funcția periodică de activare [38]:

$$P_l z = j \text{ mod } k \quad (1.8)$$

satisfăcând condiția:

$$\frac{2\pi j}{m} \leq \arg z < \frac{2\pi(j+1)}{m}, j = 0, 1, \dots, m-1; m = k \cdot l, l \geq 2$$

Valorile obținute în urma aplicării funcției de activare (1.8) reprezintă numere naturale pozitive în intervalul $[0, k)$.

Pornind de la ecuația (1.8), în [38] Aizenberg propune două strategii de învățare. Cele două strategii sunt identice din punctul de vedere al calculului ieșirii (Z), al corecției ponderilor (W), al funcției de activare (P_l) diferența constând în metoda de alegere a sectorului corect. În cazul primului algoritm se pleacă de la sectorul curent (de regulă greșit) și se aleg cele mai apropiate două sectoare corecte de o parte și de alta a acestuia, decizia finală fiind făcută pe baza distanței unghiulare; sectorul corect aflat cel mai aproape este ales, iar corecția se face pe baza acestuia [38]. În cazul celui de-al doilea algoritm de învățare selecția

sectorului corect se face doar din subdomeniul în care se află sectorul curent; cu alte cuvinte căutarea este limitată la intervale de dimensiunea intervalului $[0, k)$ [38].

În cazul primului algoritm de învățare timpul de căutare este mai mare decât în cazul celui de-al doilea algoritm datorită căutării sectorului corect în ambele direcții; acest lucru poate influența negativ procesul de învățare în cazul problemelor cu număr mare de posibile soluții ($k \geq 6$). Totodată eficiența primului algoritm este mai crescută datorită ariei mai extinse de căutare; în cazul celui de-al doilea algoritm căutarea este redusă la un subdomeniu, lucru ce ar putea cauza limitări.

Ca și concluzie preliminară, ambii algoritmi de învățare prezintă avantaje și dezavantaje. Plecând de la această observație, în partea experimentală se va prezenta o nouă strategie de selecție a sectorului corect al cărei scop este obținerea unui timp redus de căutare, păstrându-se în același timp o eficiență crescută a învățării.

2.7. Concluzii

În Capitolul 2, au fost prezentate conceptele teoretice care stau la baza RN și a neuronului artificial clasic, respectiv, a neuronului cu ponderi numere complexe (MVN) și a neuronului cu ponderi numere complexe și funcție periodică de activare (MVN-P).

Pentru RN au fost enumerate principalele tipuri de arhitecturi cum ar fi: perceptronul, perceptronul multi-strat, RN de tip BackPropagation, RN Hopfield și RN de tip Kohonen. Pentru fiecare arhitectură, au fost indicate caracteristicile principale cât și aria de aplicabilitate. De asemenea, au fost enumerate principalele tehnici de învățare folosite în cazul RN: învățarea pe baza propagării erorii dinspre stratul de ieșire spre cel de intrare (Backpropagation), învățarea hebbiană, învățarea Boltzman, învățarea competitivă și învățarea cu exemple negative. Aceste tipuri de învățări pot fi clasificate în două mari clase și anume: învățare supervizată, respectiv nesupervizată. Învățarea supervizată este cea învățare în care corecția erorii se face pe baza diferenței dintre ieșirea dorită și cea obținută, în vreme ce învățarea nesupervizată se bazează pe caracteristica legăturii dintre neuroni. Astfel, ca și învățare supervizată putem enumera: Backpropagation și învățarea cu exemple negative, iar ca și învățare nesupervizată putem enumera: învățarea hebbiană, învățarea Boltzman și învățarea competitivă.

Pentru neuronul artificial, ca și componentă fundamentală a RN, au fost identificate principalele caracteristici constructive cât și o serie de funcții de activare, cum ar fi: funcția prag, funcția semn, funcția sigmoid, funcția tangent hiperbolică, funcția lineară, funcția lineară cu saturație, funcția lineară cu saturație simetrică și funcția gaussiană. Pentru fiecare din funcțiile enumerate mai sus, caracteristica de activare a fost prezentată, atât în formă matematică cât și prin reprezentare grafică.

Neuronul cu ponderi numere complexe (MVN) a fost prezentat în detaliu, cu funcția specifică de activare, caracteristicile procesului de învățare cât și câteva reguli specifice de corecție a erorilor în timpul învățării. Acest tip de neuron reprezintă un concept relativ nou, noutatea constând în folosirea numerelor complexe pentru ponderi, respectiv valori de ieșire ale neuronului. Folosirea numerelor complexe a condus la plasarea claselor de ieșire pe cercul trigonometric. Astfel, cercul trigonometric a fost împărțit în sectoare egale, fiecare reprezentând o clasă de ieșire, procesul de învățare reducându-se la deplasarea dintr-un sector greșit într-unul corect. Această abordare a condus la posibilitatea rezolvării unor

probleme clasice folosind un singur neuron în locul unei RN multi-strat, aceasta reprezentând principalul avantaj al folosirii MVN.

În continuare, a fost prezentat neuronul cu ponderi numere complexe și funcție periodică de activare (MVN-P). MVN-P poate fi considerat ca o evoluție a MVN. Această evoluție constă în introducerea unui factor numit "periodicitate", cu ajutorul căruia se multiplică, cu o valoare dată, numărul de sectoare în care se situează clasele de ieșire. Astfel, dacă pentru MVN avem doar k posibile sectoare (k reprezentând numărul claselor de ieșire) pentru MVN-P avem kl sectoare, l reprezentând factorul de "periodicitate". Pentru acest tip de neuron au fost prezentate funcția specifică (periodică) de activare cât și două strategii de învățare, definite în literatură. Pentru fiecare dintre aceste două strategii au fost identificate avantaje și dezavantaje, referitoare la durata totală de învățare, respectiv eroarea de învățare. Plecând de la aceste avantaje și dezavantaje, în Capitolul 3, se va prezenta o nouă strategie de învățare, care dorește să îmbine avantajele celor două strategii existente.

3. CONTRIBUȚII PRIVIND ÎMBUNĂȚĂȚIREA PROCESULUI DE ÎNVĂȚARE A MVN-P

Contribuția adusă în vederea îmbunătățirii procesului de învățare se referă la propunerea și implementarea unei noi tehnici (sau strategii) de căutare a sectorului (soluției) corect.

Punctul de plecare al dezvoltării noii tehnici de căutare îl constituie cei doi algoritmi de învățare propuși de Aizenberg în [38] prezentați de asemenea, cu avantajele și dezavantajele lor, în Capitolul 2.4.1 al lucrării de față.

Ce s-a dorit a fost realizarea unei strategii care să îmbine avantajele celor doi algoritmi în vederea obținerii unui număr cât mai redus de iterații și păstrarea unei erori într-o limită acceptabilă ($< 10\%$).

3.1. Soluția propusă

Idea care stă la baza noii strategii de căutare propuse este căutarea soluției corecte (respectiv, a sectorului corespunzător de pe cercul trigonometric) urmând un sens unic de căutare [48]. Acest sens poate fi sensul orar sau anti-orar; în partea experimentală a acestei lucrări se va putea remarca influența majoră a alegerii unui sens de parcurgere sau al altuia.

În Figura 15 este prezentată interpretarea geometrică a strategiei propuse pentru sensul anti-orar:

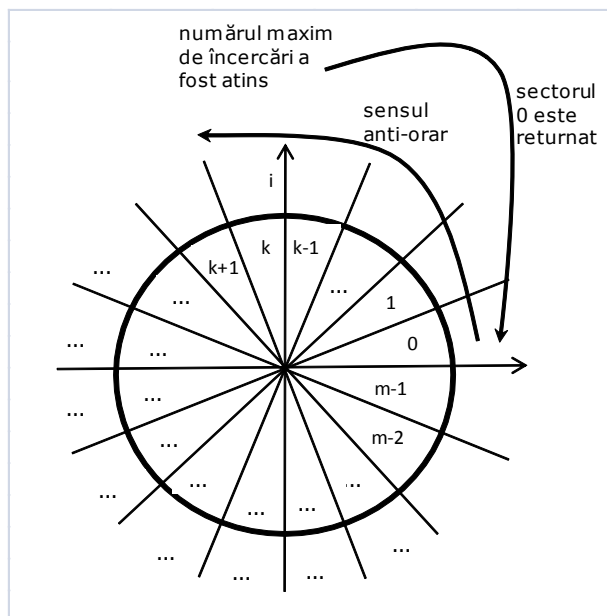


Figura 15: Interpretarea geometrică a strategiei propuse

În cadrul strategiei propuse [48] se pleacă de la sectorul curent (obținut pentru valoarea de ieșire calculată) folosind un pas de incrementare predefinit (în majoritatea cazurilor poate fi ales pasul 1). Numărul de încercări în vederea găsirii sectorului corect este setat dinamic, în timpul rulării algoritmului, în funcție de numărul de iterații parcurse. Scopul nu este căutarea continuă a sectorului corect deoarece acest lucru duce la creșterea semnificativă a timpului de învățare, în special în cazul problemelor cu număr crescut al posibilelor valori de ieșire. Dacă numărul maxim de încercări este atins și sectorul corect nu este găsit, sectorul 0 este returnat ca sector corect pentru iterația curentă. Numărul de încercări a fost setat între valoarea minimă 0 și valoarea maximă $\log_2^k + 1$, unde k reprezintă numărul de posibile ieșiri ale problemei [48].

Avantajul major identificat prin aplicarea acestei strategii de parcurgere a spațiului soluțiilor (respectiv, a cercului trigonometric) într-o singură direcție este reducerea timpului de căutare a sectorului corect și astfel implicit reducerea timpului total de învățare. Ca și minus putem preciza faptul că soluția returnată de către algoritm nu este întotdeauna cea corectă, conducând la o primă concluzie că eficiența acestei strategii este mai scăzută decât a celorlalte două; în partea experimentală se va putea observa că acest lucru nu este întotdeauna valabil, mai mult fiind posibilă eliminarea acestui neajuns odată cu integrarea MVN-P într-o RN multi-strat.

În subcapitolul următor sunt prezentați, în detaliu, pașii de implementare ai strategiei de căutare propuse.

3.2. Strategii de implementare

În continuare este prezentată strategia de căutare [48] detaliată prin pașii individuali de implementare. Ca și precizare, un număr maxim de 150000 de posibile iterații a fost setat; dacă acest număr este atins și problema nu este învățată în proporție de 100%, se consideră că algoritmul a eșuat în rezolvarea problemei și învățarea este oprită.

Implementarea strategiei de căutare propuse [48]:

- **Pasul 1 (START):** crt_set = 0, crt_epoch = 0, increment = 1, retries = 0, learning = false, success = false;
- **Pasul 2:** se calculează ieșirea curentă z, unde $z = w_0 + w_1x_1 + \dots + w_nx_n$, cu $x_1 \dots x_n$ valori de intrare, și $w_0 \dots w_n$ ponderile corespunzătoare (valoarea inițială a ponderilor este 0+0i); se identifică sectorul corespunzător și valoarea de ieșire folosindu-se funcția de activare descrisă de ecuația (1.8); dacă rezultatul este cel dorit se trece la **Pasul 3** altfel se trece la **Pasul 4**;
- **Pasul 3:** crt_set++; dacă crt_set == max_nb_sets și learning == false atunci success = true și se trece la **Pasul 6**, altfel crt_set = 0, crt_epoch++ și learning = false; dacă crt_epoch < max_nb_epochs se actualizează numărul de încercări (retries), funcție de valoarea crt_epoch, și se trece la **Pasul 2**, altfel se trece la **Pasul 6**;
- **Pasul 4:** learning = true și se începe căutarea sectorului corect; se returnează sectorul corect (sau sectorul 0 în cazul în care cel corect nu este găsit și numărul de încercări (retries) a atins valoarea maximă curentă); se trece la **Pasul 5**;

- **Pasul 5:** corecția ponderilor (se folosește regula de corecție descrisă de ecuația (1.4)); se revine la **Pasul 2**;
- **Pasul 6 (STOP):** dacă `success == false` algoritmul nu a reușit să învețe complet (100%) setul de date al problemei analizate.

În continuare sunt explicați pe scurt fiecare din parametrii de mai sus:

- **crt_set:** setul curent de intrare;
- **crt_epoch:** iterația curentă; funcție de valoarea acesteia, numărul de încercări (retries) se modifică astfel:
 - ✓ $0 \leq \text{crt_epoch} < 10001 \Rightarrow \text{retries} = 0$
 - ✓ $10000 \leq \text{crt_epoch} < 20001 \Rightarrow \text{retries} = \log_2^k$
 - ✓ $20001 \leq \text{crt_epoch} < 50001 \Rightarrow \text{retries} = \log_2^k + 1$
 - ✓ $50001 \leq \text{crt_epoch} < 100000 \Rightarrow \text{retries} = \log_2^k$
 - ✓ $100001 \leq \text{crt_epoch} \leq 150000 \Rightarrow \text{retries} = 0$
- **retries:** numărul curent de încercări;
- **learning:** parametru logic care indică dacă o corecție a ponderilor este necesară (`learning == true`) sau nu (`learning == false`);
- **success:** parametru logic care indică faptul că învățarea s-a încheiat cu succes (`success == true`) sau nu (`success == false`);
- **max_nb_sets:** numărul maxim de seturi de intrare (diferă de la o problemă la alta);
- **max_nb_epochs:** numărul maxim de iterații acceptate (setat la valoarea 150000).

În continuare sunt enumerați parametri configurabili ai strategiei de căutare propusă:

- **Constanta de învățare (C_m):** poate fi egală în majoritatea cazurilor cu 1+0i;
- **Periodicitatea posibilelor soluții (l):** are o influență importantă asupra procesului de învățare; poate lua orice valoare întreagă ≥ 2 ;
- **Pasul de căutare (increment):** poate fi egal în majoritatea cazurilor cu 1; în cazul problemelor cu număr semnificativ mare de posibile ieșiri, pasul de căutare poate fi crescut;
- **Direcția de căutare a sectorului corect:** are o influență importantă asupra rezultatului final și poate fi setată după sensul orar sau anti-orar, în momentul de Start al algoritmului;
- **Numărul maxim de iterații:** este setat la o valoare maximă de 150000, dar poate fi modificat în cazul în care problema crește în complexitate. Dacă acest număr de iterații este atins și problema nu a fost învățată 100%, algoritmul este oprit.

3.3. Concluzii

În Capitolul 3 s-a prezentat o nouă strategie de căutare a sectorului corect în vederea îmbunătățirii eficienței de învățare a MVN-P.

Ca și punct de start au fost considerate conceptele teoretice prezentate în Capitolul 2 și anume cele două strategii de învățare propuse de Aizenberg în [38]. O nouă strategie de căutare a sectorului corect a fost astfel propusă, dorindu-se pe de-o parte scăderea duratei totale de învățare, iar pe de altă parte, păstrarea unei erori de învățare scăzute (în jurul valorii de 5-10%). Noua strategie propune folosirea unui singur sens de parcurgere a cercului trigonometric (sensul anti-orar

34 Contribuții privind îmbunătățirea procesului de învățare a MVN-P - 3

sau orar), iar dacă sectorul corect nu este returnat după un număr dinamic de încercări, sectorul 0 este returnat ca sector corect. Pașii compleți de implementare, ai noii strategii, au fost prezentați în detaliu, indicându-se de asemenea parametrii configurabili ai algoritmului.

În Capitolul 4, este prezentată o soluție de integrare a MVN-P, cu noua strategie propusă, într-o RN multi-start.

4. INTEGRAREA MVN-P ÎNTR-O REȚEA NEURONALĂ

În vederea creșterii eficienței MVN-P, cu păstrarea totodată a unui timp redus de învățare, s-a dorit integrarea MVN-P, folosind strategia de căutare propusă [48], într-o RN multi-strat. Arhitectura RN s-a dorit a fi cât mai simplă (număr redus de straturi ascunse, respectiv de neuroni ascunși) pentru a nu crește semnificativ timpul total de învățare (prin introducerea unor calcule adiționale complexe) obținut în cazul unui singur MVN-P.

Ca și precizare pentru cititori, la data realizării acestui studiu nici o altă încercare de acest fel nu a fost identificată în literatura de specialitate, reprezentând o noutate în domeniu.

În subcapitolele următoare se va prezenta soluția adoptată pentru integrarea MVN-P și se va prezenta în detaliu algoritmul de antrenare folosit în timpul învățării.

4.1. Soluția propusă

Soluția propusă [49] pentru integrarea MVN-P într-o RN are în vedere folosirea unui singur MVN-P pe stratul de ieșire; strategia de căutare al sectorului corect pentru MVN-P este cea propusă în Capitolul 3.1, dar pot fi folosite, după preferință, oricare dintre strategiile prezente în literatură [38].

Pe stratul de intrare al RN, respectiv pe straturile ascunse s-au folosit MVN.

În Figura 16 este prezentată o posibilă arhitectură a unei RN de tipul MLMVN-P, în care se pot identifica principalele componente, atât fixe cât și configurabile.

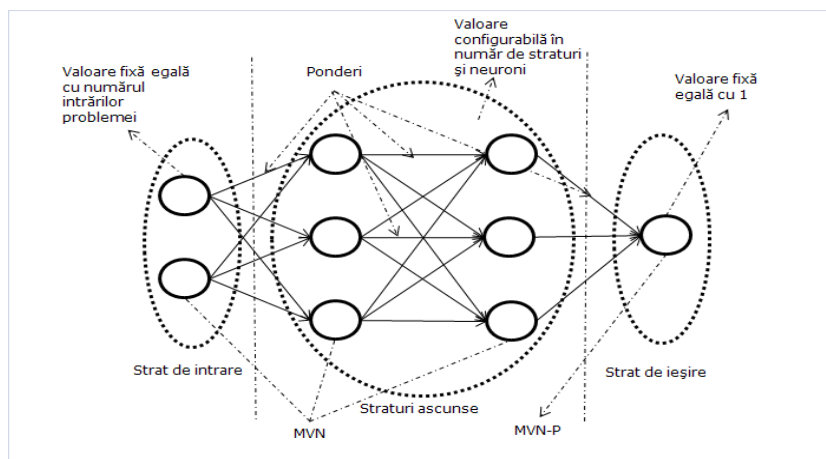


Figura 16: Arhitectura unei RN de tip MLMVN-P

Primul și ultimul strat (stratul de intrare, respectiv cel de ieșire) au structuri fixe [49]. Numărul de neuroni de pe stratul de intrare este dictat de numărul de valori de intrare ale problemei; MVN este folosit pentru acest strat [49]. Referitor la stratul de ieșire, un singur MVN-P este suficient pentru majoritatea problemelor; MVN-P poate lua o arie largă de valori, lucru ilustrat și în Figura 14 [38].

Numărul de straturi ascunse, respectiv numărul de neuroni de pe aceste straturi, este configurabil. În timpul validării MLMVN-P, diverse arhitecturi au fost folosite pornind de la 1 până la 5 straturi ascunse, respectiv 1 până la 10 neuroni pe fiecare strat ascuns; s-a dorit astfel păstrarea unor arhitecturi relativ simple. De asemenea, ca și în cazul stratului de intrare, MVN este folosit și pentru straturile ascunse. Decizia de a folosi MVN atât pentru stratul de intrare cât și pentru straturile ascunse s-a luat în considerentul că decizia de clasificare se realizează la nivelul stratului de ieșire; până în acel moment, valorile de ieșire a neuronilor de intrare și a celor ascunși vor varia într-o plajă mare de valori până vor atinge un punct de stabilitate [49].

În subcapitolul următor este prezentat în detaliu procesul de antrenare al MLMVN-P.

4.2. Strategii de implementare

În timpul procesului de învățare s-a folosit un algoritm clasic de propagare al erorii dinspre stratul de ieșire spre cel de intrare, algoritm adaptat pentru MVN, respectiv MVN-P.

În primul pas se calculează ieșirea fiecărui neuron cu formula propusă în [38]:

$$Z_p^n = \sum_{q=1}^{N_{n-1}} w_q^{p,n} \cdot Y_q^{n-1} + w_0^{p,n} \quad (3.1),$$

unde, Z_p^n reprezintă suma ponderată a neuronului p de pe stratul n (se aplică doar neuronilor ascunși și neuronului de ieșire), $w_q^{p,n}$ reprezintă ponderea intrării q a neuronului p de pe stratul n (ca valoare de inițializare a fiecărui neuron s-a folosit $1+0i$), $w_0^{p,n}$ reprezintă factorul bias al neuronului p de pe stratul n (ca valoare de inițializare a factorului bias s-a folosit $1+0i$), N_{n-1} reprezintă numărul total de neuroni de pe stratul n-1 (stratul precedent) și Y_q^{n-1} reprezintă ieșirea neuronului de intrare q de pe stratul n-1.

Pentru primul strat, Y este egal cu X, vectorul total de intrări ale problemei; pentru toate celelalte straturi următoarele formule se aplică [49]:

$$Y_p^n = Z_p^n \quad (3.2)$$

, pentru straturile ascunse, respectiv [38]:

$$Y_p^n = j \bmod k \quad (3.3),$$

pentru stratul de ieșire, unde j și k sunt cele din ecuația (1.8); j reprezintă sectorul curent, unde este localizată ieșirea și k reprezintă numărul total de valori posibile pe care ieșirea le poate avea.

Pentru neuronii ascunși nu este nevoie de o funcție specifică de activare. Acest lucru se datorează faptului că pentru corecția ponderilor se folosește distanța unghiulară dintre sectoare și nu amplitudinea sumei ponderate, Z_p^n .

Al doilea pas al algoritmului de învățare îl constituie corecția ponderilor. Următoarele formule sunt folosite pentru toți neuronii [23], [29]:

$$w_q^{p,n} = w_q^{p,n} + \frac{C}{N_{n-1}+1} \delta_p^n Y_q^{n-1} \quad (3.4)$$

$$w_0^{p,n} = w_0^{p,n} + \frac{C}{N_{n-1}+1} \delta_p^n$$

, unde C reprezintă constanta de învățare (poate fi egală cu 1+0i în majoritatea cazurilor), δ_p^n reprezintă eroarea neuronului p de pe stratul n, N_{n-1} , reprezintă numărul total de neuroni de pe stratul n-1 și Y_q^{n-1} reprezintă ieșirea neuronului q de pe stratul n-1.

Pentru calculul δ_p^n următoarele formule sunt folosite [23], [29]:

$$\delta_p^n = \frac{1}{N_{n-1}+1} (\varepsilon^q - \varepsilon^s) \quad (3.5)$$

, pentru stratul de ieșire, unde q reprezintă sectorul dorit și s reprezintă sectorul curent, respectiv,

$$\delta_p^n = \frac{1}{N_{n-1}+1} \sum_{q=1}^{N_{n+1}} \delta_q^{n+1} (\delta_p^{q,n+1})^{-1} \quad (3.6)$$

, pentru straturile ascunse. Figura 17 prezintă un exemplu simplificat de propagare al erorii în care autorul poate face corelația între notații și parametrii din (3.4), (3.5) și (3.6) [49]:

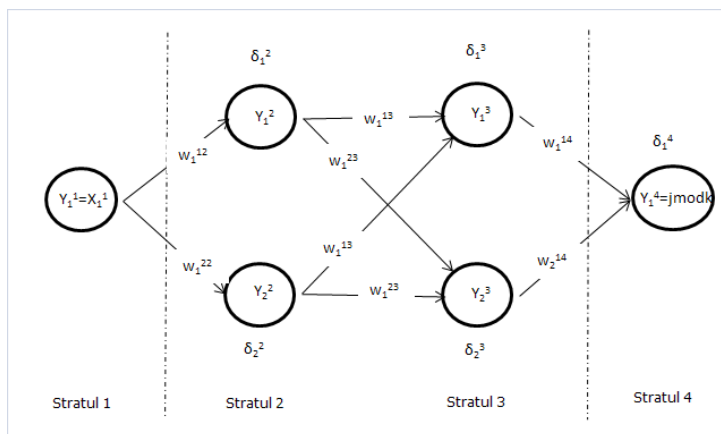


Figura 17: Un exemplu de propagare al erorii dinspre stratul de ieșire spre stratul de intrare

4.3. Concluzii

În Capitolul 4 s-a prezentat o soluție de integrare a MVN-P într-o RN multi-strat în vederea îmbunătățirii eficienței de învățare a MVN-P.

Integrarea MVN-P într-o RN reprezintă o noutate în domeniu, încercându-se astfel creșterea suplimentară a eficienței MVN-P. Arhitectura folosită pentru integrare a fost detaliată și un mecanism de învățare, bazat pe corecția erorii (Backpropagation) a fost prezentat. În cadrul acestui mecanism au fost identificate funcțiile specifice de activare a neuronilor de pe fiecare strat în parte, cât și regulile de corecție a ponderilor acestora.

În Capitolul 5, rezultatele obținute, pe baza conceptelor teoretice prezentate în Capitolul 3 și Capitolul 4, vor fi ilustrate.

5. REZULTATE EXPERIMENTALE

Capitolul de față prezintă rezultatele obținute în timpul validării noii strategii de căutare propuse, cât și rezultatele obținute în urma integrării MVN-P într-o RN. Diverse baze de date des utilizate în literatură au fost folosite în timpul validării, ca: IRIS, adunarea modulo k a n valori, „Breast Cancer Wisconsin (Diagnostic)”.

În cazul fiecărei probleme abordate pentru rezolvare s-au identificat avantajele, respectiv dezavantajele folosirii conceptelor propuse. De asemenea, la sfârșitul fiecărui experiment s-au formulat concluzii pe baza cărora se poate analiza eficiența conceptelor introduse.

Referitor la analiza rezultatelor obținute pentru MLMVN-P, aceasta s-a făcut comparând la fiecare pas rezultatele obținute pentru un singur MVN-P. În acest sens, pentru evaluarea eficienței MLMVN-P comparativ cu cea a unui singur MVN-P s-au luat în calcul parametrii de învățare ca: eficiența învățării (respectiv, rata erorii), timpul total de învățare cât și complexitatea RN.

5.1. Rezultatele produse de noua strategie de căutare la nivelul MVN-P

În continuare sunt prezentate rezultatele obținute în timpul validării noii strategii de căutare a sectorului corect, în timpul învățării pentru MVN-P [48].

Rezultatele obținute au fost comparate cu cele mai bune rezultate raportate în literatura de specialitate până la momentul redactării acestei lucrări, în vederea confirmării sau infirmării supozițiilor inițiale. Astfel s-au putut identifica avantajele, respectiv dezavantajele folosirii noii strategii comparativ cu cele deja existente în literatura de specialitate.

5.1.1. Suma modulo k a n valori

Suma modulo k a n valori reprezintă un set de date cu o caracteristică neliniară, des folosit în literatură pentru a evidenția avantajele folosirii MVN-P. Astfel de seturi de date nu pot fi învățate în întregime folosind un singur MVN, în schimb pot fi învățate 100% folosind un singur MVN-P.

Ecuția (4.1) prezintă funcția matematică a sumei modulo k a n valori:

$$f_{x_1, x_2, \dots, x_n} = x_1 + x_2 + \dots + x_n \text{ mod}(k) \quad (4.1)$$

În Anexa 1 este prezentată o astfel de bază de date pentru $n = 3$ și $k = 4$.

În vederea validării noii strategii de căutare propuse, diverse configurații ale acestui set de date au fost folosite. Astfel, numărul n a variat de la 2 până la 6 valori de intrare, iar k de la 3 până la 6.

În continuare, sunt prezentate rezultatele obținute comparativ cu cele prezentate în literatură [38]. Analiza comparativă a rezultatelor este utilă în identificarea plusurilor și minusurilor unei strategii față de celelalte.

5.1.1.1. Rezultate comparative

În Tabelul 3 sunt listate rezultatele obținute în timpul validării noi strategii de căutare a sectorului corect în momentul învățării [48]. Rezultatele folosite pentru comparare sunt cele mai bune rezultate raportate în literatură pentru MVN-P [38] până la momentul redactării acestei lucrări. De asemenea, rezultatele prezentate pentru noua strategie sunt cele mai bune obținute în urma rulării a 60 de teste pentru fiecare set de date în parte. Cu fiecare test, valoarea l a fost incrementată cu 1, plecându-se de la valoarea 2.

Regula folosită pentru corecția ponderilor neuronilor, pentru ambele seturi de rezultate, este cea descrisă de ecuația (4.1); pentru această regulă s-au obținut în medie cele mai bune rezultate pentru ambele strategii. De asemenea, pentru setul de rezultate obținute cu noua strategie de căutare, ca direcție de parcurgere a cercului trigonometric a fost ales sensul anti-orar, iar pasul de căutare a fost setat la valoarea 1.

Tabelul 3: Rezultate comparative pentru suma modulo k

| Setul de date | | Rezultate raportate în literatură | | | Rezultate obținute cu noua strategie de căutare | | |
|---------------|-----|-----------------------------------|---------------|---------------|---|---------------|---------------|
| n | k | Iterații | l -periodic | Eficiență (%) | Iterații | l -periodic | Eficiență (%) |
| 2 | 3 | 14 | 2 | 100 | 25 | 49 | 100 |
| 2 | 4 | 2693 | 3 | 100 | 44 | 7 | 100 |
| 2 | 5 | 96 | 4 | 100 | 144 | 8 | 100 |
| 2 | 6 | 272 | 4 | 100 | 363 | 8 | 100 |
| 3 | 3 | 2466 | 10 | 100 | 54 | 11 | 100 |
| 3 | 4 | 2571 | 7 | 100 | 165 | 18 | 100 |
| 3 | 5 | 1202 | 9 | 100 | 3082 | 14 | 100 |
| 3 | 6 | 109733 | 14 | 100 | 10014 | 60 | 100 |
| 4 | 3 | 4296 | 11 | 100 | 83 | 11 | 100 |
| 4 | 4 | 50151 | 10 | 100 | 3317 | 23 | 91.406 |
| 4 | 5 | 4604 | 13 | 100 | 127966 | 24 | 65.440 |
| 4 | 6 | 118128 | 14 | 100 | 19857 | 19 | 50.694 |
| 5 | 3 | 78596 | 18 | 100 | 114392 | 20 | 100 |
| 5 | 4 | 734691 | 13 | 100 | 140021 | 45 | 68.652 |
| 6 | 3 | 237202 | 36 | 100 | 119119 | 20 | 92.181 |

Se poate remarca din Tabelul 3 o creștere semnificativă a eficienței MVN-P pentru o serie de seturi de date. Pentru aceste seturi de date, numărul de iterații a scăzut pentru aceeași eficiență de 100%. Pentru alte seturi de date (e.x. $n = 4$, $k = 5$) noua strategie de învățare s-a dovedit mai puțin eficientă. Pentru aceste seturi de date, eficiență de 100% nu a fost obținută după numărul maxim stabilit de iterații (150000). Pentru astfel de situații, în special, dar nu numai, integrarea MVN-P într-o RN devine o necesitate în vederea creșterii eficienței de învățare. În Capitolul 5.2 se va identifica dacă acest deziderat a fost sau nu atins.

5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 41

Tabelul 4 prezintă analiza statistică (realizată în OriginPro 8 SR0) a celor două populații obținute pentru cele două strategii de căutare comparate, pentru 3 seturi de date alese aleator; aici, prin populație se înțelege un set de date constituit din totalitatea sectoarele/domeniilor returnate ca și sectoare corecte în timpul învățării.

Tabelul 4: Analiza statistică descriptivă a populațiilor obținute pentru cele două strategii de căutare comparate

a) $n = 3, k = 4, l = 18$

a.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Strategia de parcurgere | N | Mean | SD | SEM |
|-------------------------|-------|------------|------------|------------|
| Aizenberg | 10560 | 8.79280303 | 5.00800243 | 0.04873405 |
| Noua strategie | 10560 | 2.00918560 | 5.01642447 | 0.04881601 |
| Diferența | | 6.78361742 | | |

a.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Strategia de parcurgere | N | Mean | SD | SEM |
|-------------------------|-------|------------|------------|------------|
| Aizenberg | 10560 | 1.50151515 | 1.11282188 | 0.01082913 |
| Noua strategie | 10560 | 1.36401515 | 1.10533946 | 0.01075632 |
| Diferența | | 0.1375 | | |

b) $n = 4, k = 3, l = 11$

b.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Strategia de parcurgere | N | Mean | SD | SEM |
|-------------------------|------|------------|------------|------------|
| Aizenberg | 6723 | 5.21448757 | 3.40831170 | 0.04156786 |
| Noua strategie | 6723 | 1.37691506 | 2.56155685 | 0.03124082 |
| Diferența | | 3.83757251 | | |

b.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Strategia de parcurgere | N | Mean | SD | SEM |
|-------------------------|------|------------|------------|------------|
| Aizenberg | 6723 | 0.99910754 | 0.81363662 | 0.00992313 |
| Noua strategie | 6723 | 0.98348951 | 0.80065869 | 0.00976485 |
| Diferența | | 0.01561802 | | |

c) $n = 5, k = 3, l = 20$

c.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Strategia de parcurgere | N | Mean | SD | SEM |
|-------------------------|-------|------------|------------|------------|
| Aizenberg | 29889 | 7.78610860 | 6.47994453 | 0.03748138 |
| Noua strategie | 29889 | 2.12181739 | 4.68839467 | 0.02711867 |
| Diferența | | 5.66429121 | | |

| Strategia de parcurgere | N | Mean | SD | SEM |
|-------------------------|------|-------------|------------|------------|
| Aizenberg | 7290 | 15.46748971 | 1.08043851 | 0.01265424 |
| Noua strategie | 7290 | 1.83415637 | 4.28822492 | 0.05022428 |
| Diferența | | 13.63333333 | | |

c.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Strategia de parcurgere | N | Mean | SD | SEM |
|-------------------------|-------|------------|------------|------------|
| Aizenberg | 29889 | 1.00043494 | 0.81595674 | 0.00471966 |
| Noua strategie | 29889 | 0.99524908 | 0.81218234 | 0.00469783 |
| Diferența | | 0.00518585 | | |

| Strategia de parcurgere | N | Mean | SD | SEM |
|-------------------------|------|------------|------------|------------|
| Aizenberg | 7290 | 1.00919067 | 0.81574439 | 0.00955411 |
| Noua strategie | 7290 | 0.98943758 | 0.81320109 | 0.00952432 |
| Diferența | | 0.01975308 | | |

Parametrii statistici evidențiați mai sus au următoarea semnificație:

- **N:** numărul de valori supuse analizei (dimensiunea populației);
- **Mean:** valoarea medie statistică a populației analizate;
- **SD:** variația populației față de valoarea medie;
- **SEM:** eroarea standard a valorii medii.

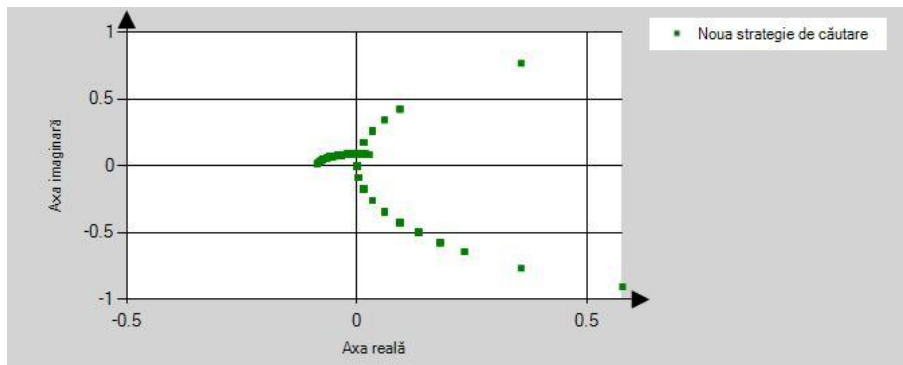
Deoarece numărul de iterații în care problema a fost rezolvată complet de o strategie sau alta, diferă semnificativ, pentru analiza statistică s-au luat în calcul doar valorile returnate după numărul cel mai mic de iterații necesar învățării (e.x. în cazul $n = 3$, $k = 4$, $l = 18$ s-au analizat doar rezultatele returnate de ambele strategii până la iterația 165 inclusiv).

Se poate observa din Tabelul 4 o diferență statistică (identificată prin parametrul "Diferența" din tabel) majoră între valorile medii statistice (identificate prin parametrul "Mean") a celor două populații, de unde și rezultatele semnificativ diferite. Această diferență se poate observa atât în cazul domeniilor returnate, cât și în cazul sectoarelor returnate. Pentru fiecare set de date în parte, numărul total de domenii posibile este egal cu valoarea parametrului l , iar numărul total de sectoare din fiecare domeniu este egal cu valoarea parametrului k . Diferența majoră este cauzată de specificitatea fiecărei strategii de căutare folosite, cu alte cuvinte de felul în care domeniul de posibile soluții este parcurs (complet în cazul strategiei din [38], respectiv parțial în cazul strategiei propuse [48]) și de numărul de încercări de găsire a sectorului corect (în cazul strategiei din [38] sectorul corect este întotdeauna returnat, numărul de încercări putând fi astfel considerat infinit, în vreme ce pentru strategia [48] încercările sunt finite și limitate la valoarea $\log_2^k + 1$).

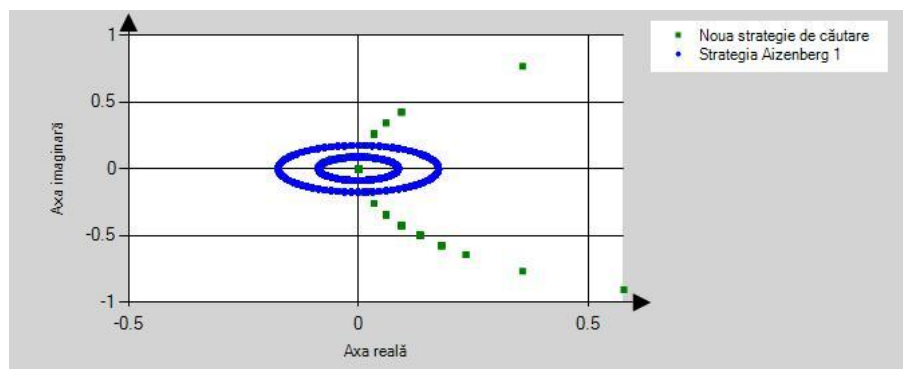
Analize statistice asemănătoare au fost realizate pentru toate seturile de date considerate, cu rezultate similare. În cazul populațiilor de dimensiune mare, ca și în cazul $n = 5$, $k = 3$, $l = 20$, datorită limitărilor programului de analiză, acestea au fost împărțite în populații mai restrânse și analizate separat.

În Figura 18, 19, respectiv 20 sunt ilustrate variațiile erorii de învățare, descrise prin ecuația (3.5), pentru aceleași seturi de date, analizate statistic, de mai sus.

5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 43



a)



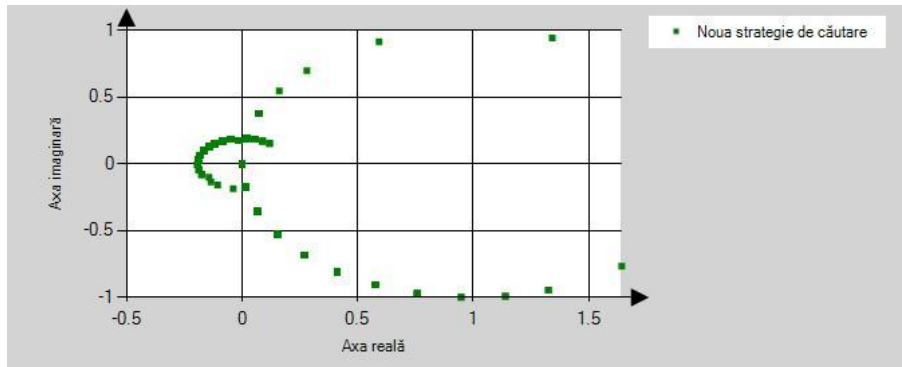
b)

Figura 18: Variația erorii de învățare pentru a) noua strategie de învățare, respectiv b) strategia Aizenberg considerând cazul $n = 3$, $k = 4$, $l = 18$

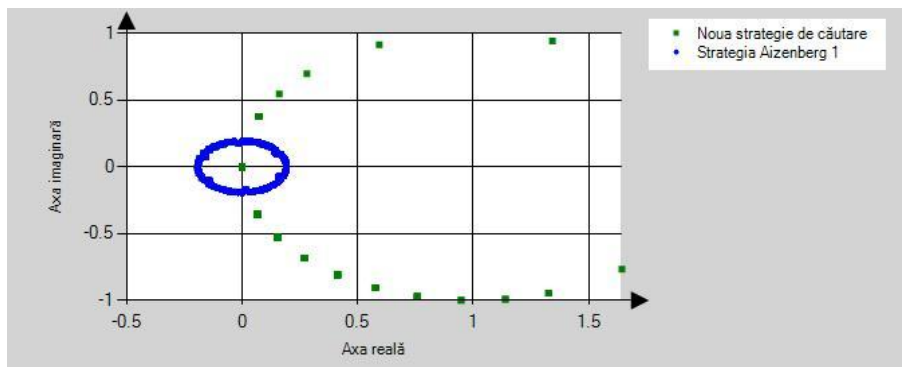
În Figura 18 cu verde este reprezentată variația erorii pentru noua strategie de căutare [48], iar cu albastru variația erorii în cazul strategiei 1 din [38]. Această variație este bazată pe ecuația (3.5) și este proporțională cu diferența dintre sectorul dorit și cel returnat de strategia folosită (vezi Figura 13) pentru fiecare set de valori de intrare, în parte.

În cazul noii strategii propuse, se observă o variație a erorii în afara spațiului comun de valori, dovadă a sectoarelor (soluțiilor) diferite returnate de cele două strategii. Această variație este direct legată de diferența statistică a populațiilor generate de cele două strategii, prezentată în Tabelul 4 de mai sus. Datorită acestui aspect, ponderile neuronilor evoluează distinct, cu o variație dictată de valoarea diferenței ($\varepsilon^q - \varepsilon^s$), valoare total diferită pentru cele două strategii considerate.

Analizând Figura 19 și Figura 20, concluzii similare se pot formula. Din nou, pentru strategia propusă [48] spațiul erorii, descrisă prin ecuația (3.5), variază semnificativ în afara spațiului comun de valori, ca urmare a valorilor distincte returnate de cele două strategii în diferite momente ale învățării. Această variație se poate observa pentru diverse valori ale parametrului periodic l , respectiv ale parametrului k , fiind specifică noii strategii de căutare (lucru evidențiat în Figura 18, Figura 19, respectiv Figura 20).

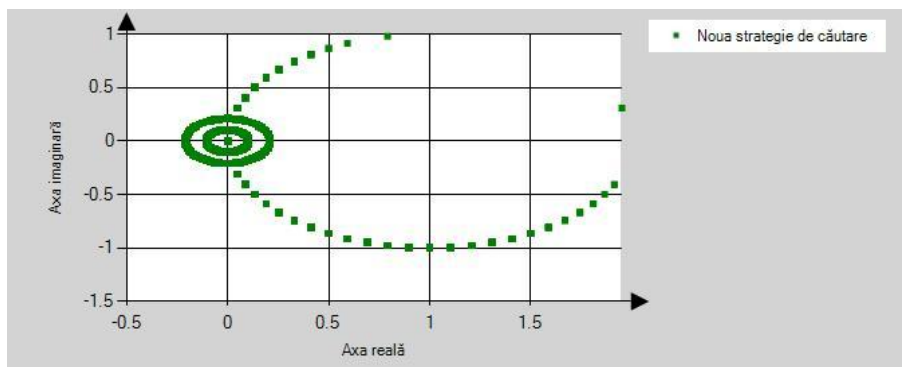


a)



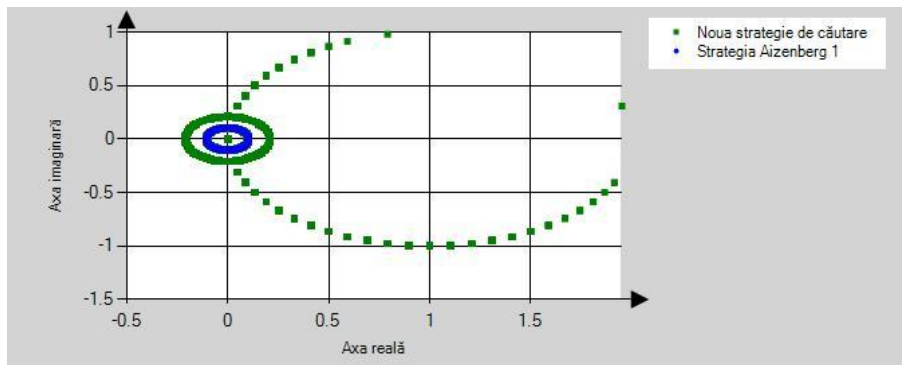
b)

Figura 19: Variația erorii de învățare pentru a) noua strategie de învățare, respectiv b) strategia Aizenberg considerând cazul $n = 4$, $k = 3$, $l = 11$



a)

5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 45



b)

Figura 20: Variația erorii de învățare pentru a) noua strategie de învățare, respectiv b) strategia Aizenberg considerând cazul $n = 5$, $k = 3$, $l = 20$

Până acum, pentru noua strategie de căutare [48] s-a considerat doar sensul anti-orar de parcurgere a cercului trigonometric. În continuare, în Tabelul 5 sunt prezentate câteva rezultate comparative între eficiențele strategiei noi de căutare în cazul folosirii, pe de o parte a sensului anti-orar de căutare, iar pe de altă parte a sensului orar de căutare. După cum s-a enunțat, se poate observa cu ușurință faptul că alegerea unui sens sau a altuia de parcurgere a cercului trigonometric, în momentul căutării sectorului corect, are o influență majoră asupra întregului proces de învățare. Rezultatele prezentate sunt cele mai bune obținute pentru ambele sensuri de căutare, în urma rulării a 60 de teste independente pentru fiecare set de date în parte.

Tabelul 5: Rezultate comparative în cazul folosirii a două sensuri diferite de căutare

| Setul de date | | Rezultate raportate pentru sensul orar | | | Rezultate raportate pentru sensul anti-orar | | |
|---------------|---|--|------------|---------------|---|------------|---------------|
| n | k | Iterații | l-periodic | Eficiență (%) | Iterații | l-periodic | Eficiență (%) |
| 2 | 3 | 10034 | 42 | 100 | 25 | 49 | 100 |
| 2 | 4 | 52923 | 49 | 100 | 44 | 7 | 100 |
| 2 | 5 | 20643 | 7 | 96 | 144 | 8 | 100 |
| 2 | 6 | 31869 | 57 | 77.778 | 363 | 8 | 100 |
| 3 | 3 | 59754 | 16 | 96.296 | 54 | 11 | 100 |
| 3 | 4 | 95065 | 18 | 71.875 | 165 | 18 | 100 |
| 4 | 2 | 12 | 58 | 100 | 60 | 52 | 100 |
| 4 | 3 | 96750 | 12 | 80.247 | 83 | 11 | 100 |
| 5 | 2 | 8973 | 31 | 100 | 279 | 19 | 100 |
| 5 | 3 | 15447 | 2 | 65.432 | 114392 | 20 | 100 |

Pentru toate seturile de date considerate, se poate observa, cu o singură excepție în cazul $n = 4$, $k = 2$, faptul că sensul orar de căutare produce o eficiență

46 Rezultate experimentale - 5

mai scăzută în timpul învățării, comparativ cu sensul anti-orar. Anumite seturi de date care au fost cu ușurință învățate complet într-un număr redus de iterații, nu au putut fi învățate complet în numărul maxim de iterații stabilite (150000).

În Tabelul 6 sunt prezentate distribuțiile statistice ale celor două populații, produse de cele două sensuri de parcurgere, pentru strategia propusă [48]. Seturile de date folosite pentru analiză au fost alese aleator, dar observații similare se pot formula pentru toate seturile de date considerate în Tabelul 5.

Tabelul 6: Analiza statistică descriptivă a populațiilor obținute pentru sensul de parcurgere orar, respectiv anti-orar

a) $n = 2, k = 5, l = 8$

a.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Sensul de parcurgere | N | Mean | SD | SEM |
|----------------------|------|------------|------------|------------|
| Sensul orar | 3600 | 1.53305555 | 2.89436443 | 0.04823940 |
| Sensul anti-orar | 3600 | 0.93 | 2.13274136 | 0.03554568 |
| Diferența | | 1.23152777 | | |

a.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Sensul de parcurgere | N | Mean | SD | SEM |
|----------------------|------|------------|------------|------------|
| Sensul orar | 3600 | 0.88888889 | 1.65145414 | 0.02752423 |
| Sensul anti-orar | 3600 | 1.68888889 | 1.47513950 | 0.02458565 |
| Diferența | | 1.28888889 | | |

b) $n = 3, k = 4, l = 18$

b.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Sensul de parcurgere | N | Mean | SD | SEM |
|----------------------|-------|------------|------------|------------|
| Sensul orar | 10560 | 5.09195075 | 7.78581858 | 0.07576564 |
| Sensul anti-orar | 10560 | 2.00918560 | 5.01642447 | 0.04881601 |
| Diferența | | 3.08276515 | | |

b.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Sensul de parcurgere | N | Mean | SD | SEM |
|----------------------|-------|-------------|------------|------------|
| Sensul orar | 10560 | 0.90321969 | 1.35403871 | 0.01317647 |
| Sensul anti-orar | 10560 | 1.36401515 | 1.10533946 | 0.01075632 |
| Diferența | | -0.46079545 | | |

c) $n = 5, k = 2, l = 19$

c.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Sensul de parcurgere | N | Mean | SD | SEM |
|----------------------|------|-------------|------------|------------|
| Sensul orar | 8928 | 9.05745967 | 5.50838259 | 0.05829710 |
| Sensul anti-orar | 8928 | 9.17898745 | 5.34429044 | 0.05656046 |
| Diferența | | -0.12152777 | | |

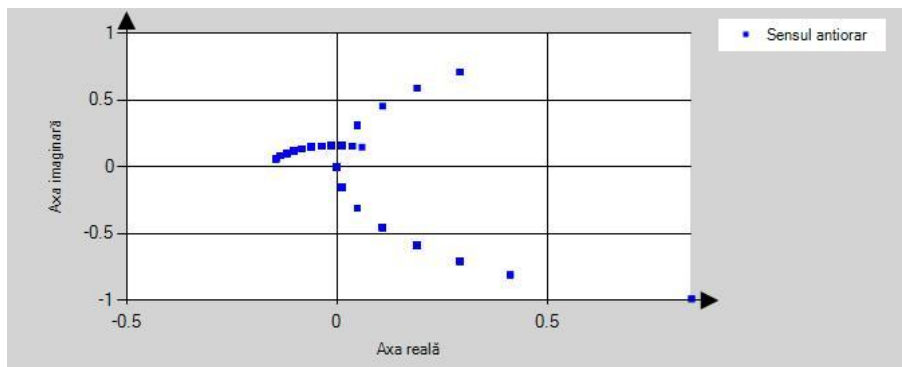
5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 47

c.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

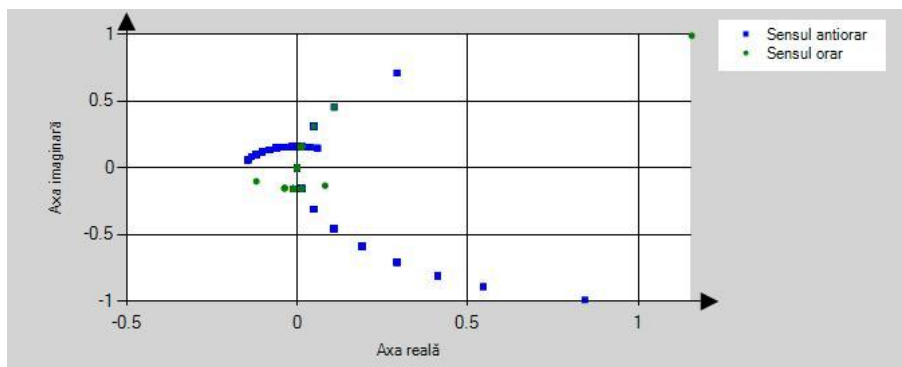
| Sensul de parcurgere | N | Mean | SD | SEM |
|----------------------|------|------------|------------|------------|
| Sensul orar | 8928 | 0.50929659 | 0.49994156 | 0.00529105 |
| Sensul anti-orar | 8928 | 0.50884856 | 0.49994969 | 0.00529113 |
| Diferența | | 0.00044803 | | |

Se poate observa din Tabelul 6 o diferență statistică (identificată prin parametrul "Diferența" din tabel) semnificativă între valorile medii statistice (identificate prin parametrul "Mean") a celor două populații atât la nivelul domeniilor cât și al sectoarelor returnate, excepție fiind cazul $n = 5$, $k = 2$; pentru acest caz diferența statistică a sectoarelor returnate este nesemnificativă datorită numărului redus de posibile clase de ieșire (doar 2 clase de ieșire).

În Figura 21, 22, respectiv 23 sunt ilustrate variațiile erorii de învățare, descrise prin ecuația (3.5), pentru aceleași seturi de date, analizate statistic, de mai sus. Numărul de iterații a fost limitat la numărul minim obținut pentru învățarea completă a setului de date.



a)

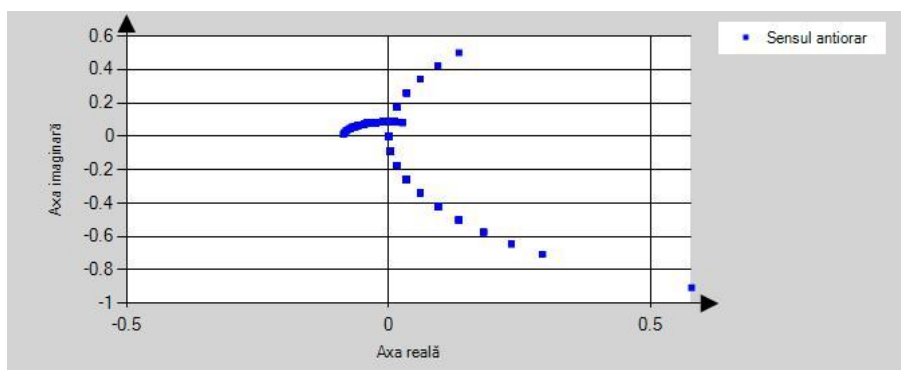


b)

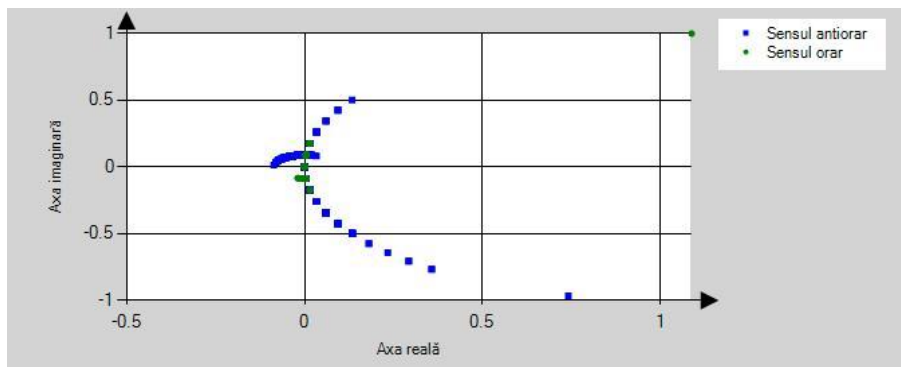
Figura 21: Variația erorii de învățare pentru a) sensul anti-orar, respectiv b) sensul orar considerând cazul $n = 2$, $k = 5$, $l = 8$

În Figura 21 cu verde este reprezentată variația erorii pentru sensul orar, iar cu albastru variația erorii pentru sensul anti-orar. În cazul sensului orar eroarea variază lent lucru indicat prin numărul redus de puncte verzi de pe grafic. Datorită acestui fapt, ponderile evoluează la rândul lor lent (direct proporțional cu diferența $(\varepsilon^q - \varepsilon^s)$), lucru ce conduce la creșterea semnificativă a numărului de iterații în care setul de date este învățat 100%.

Analizând Figura 22, concluzii similare se pot formula. Din nou, pentru sensul orar de parcurgere, spațiul erorii denotă o evoluție lentă a acesteia, și indirect o evoluție lentă a ponderilor.



a)

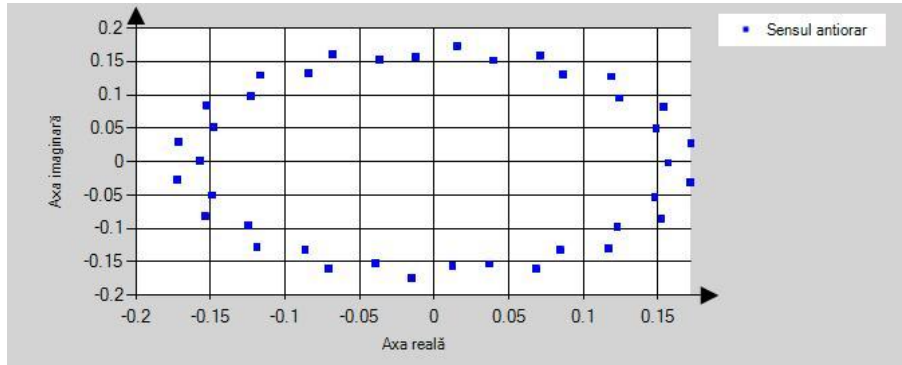


b)

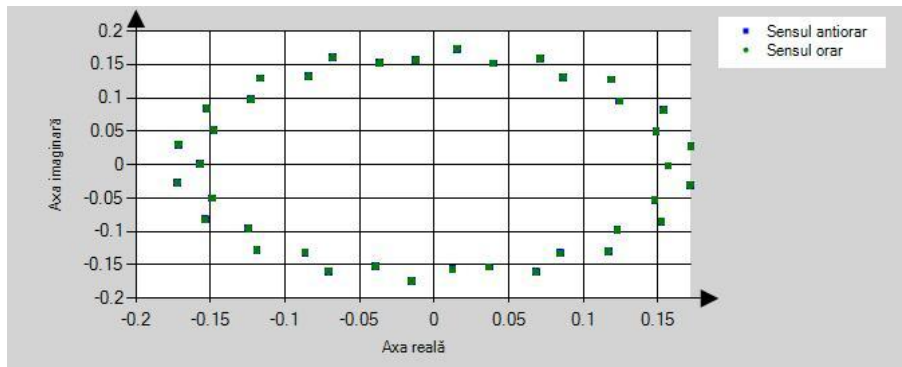
Figura 22: Variația erorii de învățare pentru a) sensul anti-orar, respectiv b) sensul orar considerând cazul $n = 3, k = 4, l = 18$

În Figura 23 evoluția erorii este similară pentru ambele sensuri de parcurgere datorită numărului redus de clase de ieșire. Acest lucru a fost de asemenea ilustrat în Tabelul 6 c.2) prin diferența statistică nesemnificativă a populațiilor sectoarelor returnate de cele două sensuri de parcurgere. În cazul acestui set de date, diferența statistică a domeniilor returnate este cea care conduce la rezultate diferite.

5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 49



a)



b)

Figura 23: Variația erorii de învățare pentru a) sensul anti-orar, respectiv b) sensul orar considerând cazul $n = 5$, $k = 2$, $l = 19$

De-a lungul experimentelor s-a observat o influență majoră a factorului l , care indică periodicitatea soluțiilor, asupra procesului de învățare [50]. Astfel, pentru seturile de date prezentate în acest capitol, s-a realizat un studiu al acestui parametru încercându-se a se identifica o plajă comună de valori ale lui l pentru diverse seturi de date, pentru care eficiența procesului de învățare este cea mai bună. În Tabelul 7 sunt listate seturile de date folosite în cadrul acestui studiu.

Tabelul 7: Seturile de date folosite la analiza influenței parametrului l

| n – numărul de intrări | k – posibile ieșiri | Numărul de seturi de date |
|-------------------------------|----------------------------|----------------------------------|
| 2 | 3 | 9 |
| 2 | 4 | 16 |
| 2 | 5 | 25 |

50 Rezultate experimentale - 5

| | | |
|---|---|------|
| 2 | 6 | 36 |
| 3 | 3 | 27 |
| 3 | 4 | 64 |
| 3 | 5 | 125 |
| 4 | 2 | 16 |
| 4 | 3 | 81 |
| 4 | 4 | 256 |
| 5 | 2 | 32 |
| 5 | 3 | 243 |
| 5 | 4 | 1024 |

În continuare sunt prezentate rezultatele obținute [50].

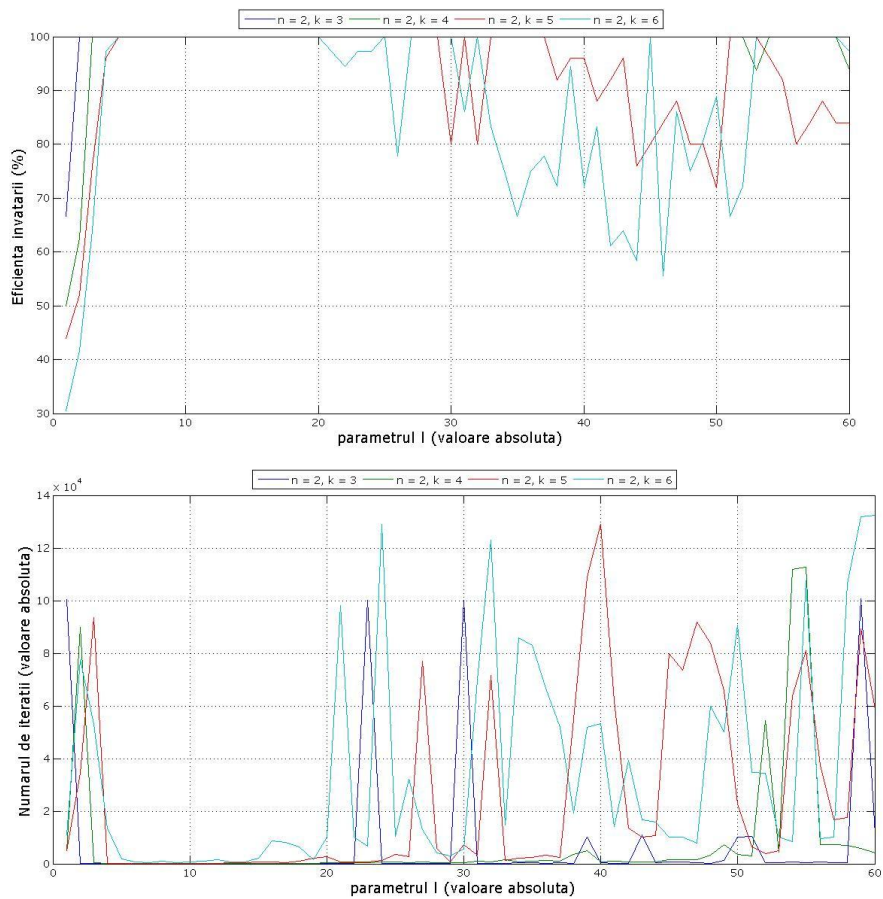


Figura 24: Rezultate pentru $n = 2$ și $k = 3 \div 6$

5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 51

Pentru grupul 1 ($n=2$) cele mai bune rezultate s-au obținut pentru l între 6 și 21 (eficiență de 100% pentru întreg intervalul cu un număr mediu de iterații de 75.3125 ($k=3$), 151.375 ($k=4$), 636.5625 ($k=5$), respectiv 2844.0625 ($k=6$)) [50].

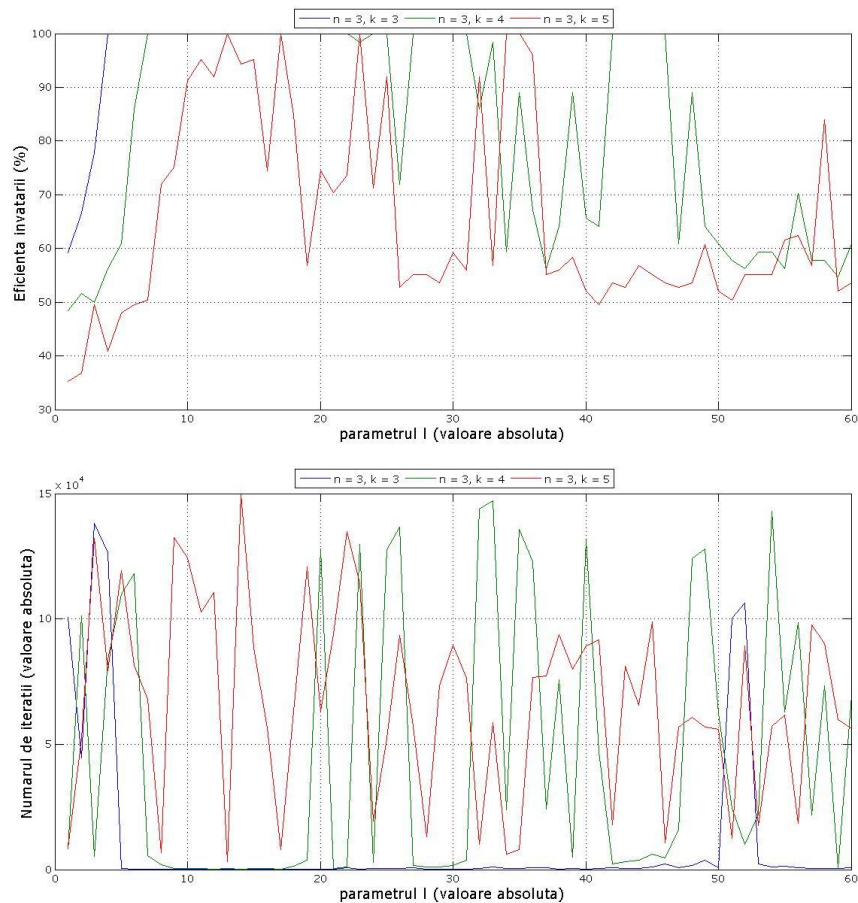


Figura 25: Rezultate pentru $n = 3$ și $k = 3 \div 5$

Pentru grupul 2 ($n=3$) cele mai bune rezultate s-au obținut pentru l între 11 și 19 (eficiență de 100% pentru întreg intervalul, pentru $k=3$ și 4 cu un număr mediu de iterații de 80($k=3$) respectiv 392.167 ($k=4$), și o eficiență medie de 91.91% cu un număr mediu de iterații de 78520.33 pentru $k=5$) [50].

52 Rezultate experimentale - 5

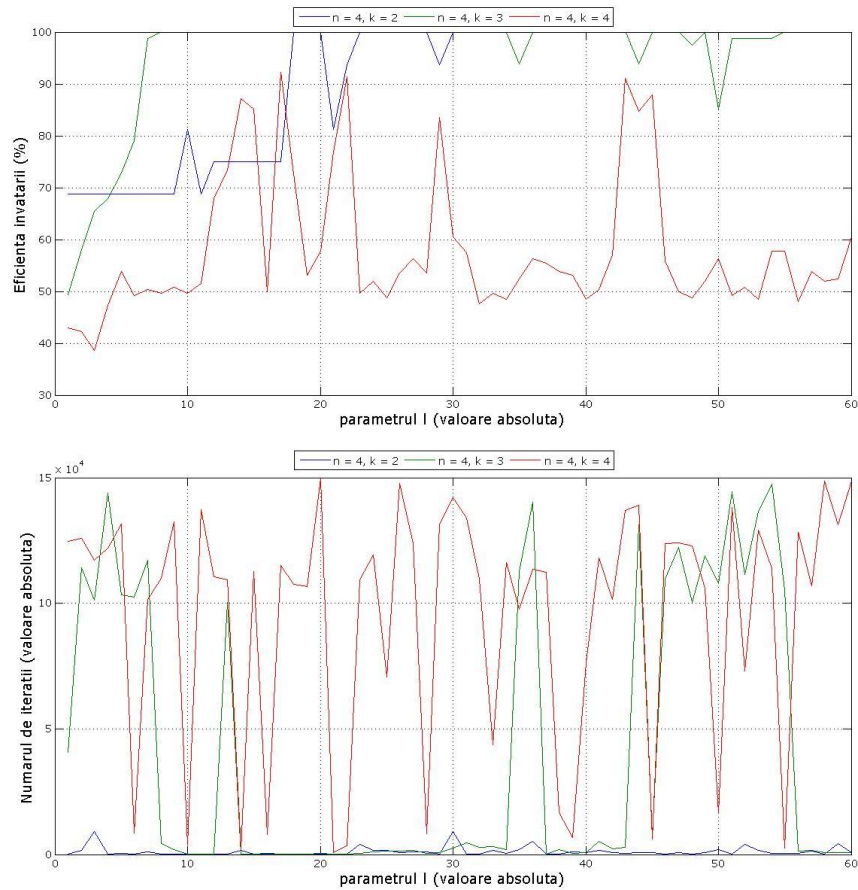


Figura 26: Rezultate pentru $n = 4$ și $k = 2 \div 4$

Pentru grupul 3 ($n=4$) cele mai bune rezultate s-au obținut pentru l între 18 și 23 (eficiență medie de 91.67% cu un număr mediu de iterații de 78.67 pentru $k=2$, eficiență medie de 100% cu un număr mediu de iterații de 224.83 pentru $k=3$, respectiv, o eficiență medie de 74.0235% cu un număr mediu de iterații de 80464 pentru $k=4$) [50].

5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 53

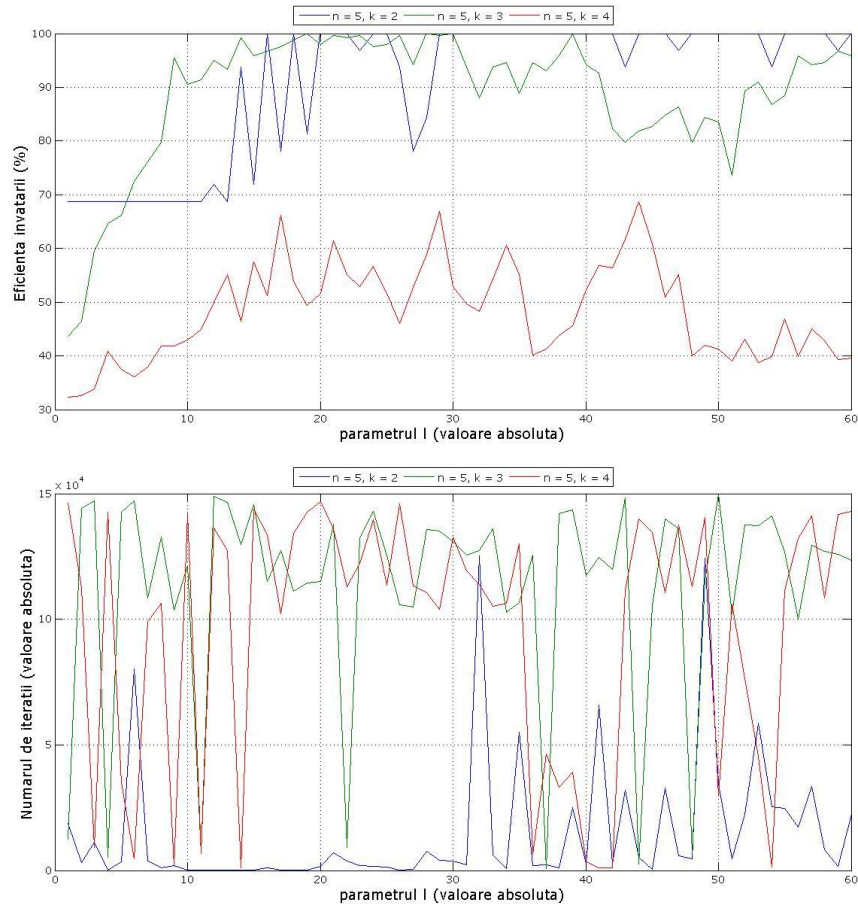


Figura 27: Rezultate pentru $n = 5$ și $k = 2 \div 4$

Pentru grupul 4 ($n=5$) cele mai bune rezultate s-au obținut pentru l între 17 și 26 (eficiență medie de 95.625% cu un număr mediu de iterații de 1872.3 pentru $k=2$, eficiență medie de 98.4772% cu un număr mediu de iterații de 113050.6, respectiv, o eficiență medie de 55.0196% cu un număr mediu de iterații de 128426.4 pentru $k=4$) [50].

Pentru a sumariza rezultatele obținute două concluzii majore au fost formulate: parametrul l are o influență majoră asupra procesului de învățare (pentru fiecare set de date în parte) și o plajă comună de valori pentru l (pentru toate seturile de date analizate) poate fi identificată, valori pentru care se obțin cele mai bune rezultate [50].

5.1.1.2. Concluzii preliminare

Ca și concluzii preliminare pentru rezultatele prezentate în acest subcapitol se pot formula următoarele:

1. Noua strategie de căutare propusă a condus la îmbunătățirea, în anumite cazuri substanțială, a eficienței procesului de învățare.

2. Pentru seturile de date cu număr crescut de intrări, respectiv ieșiri ($k, n \geq 5$) noua strategie s-a dovedit mai puțin eficientă.
3. Diferențele statistice majore dintre populațiile obținute în timpul învățării pentru noua strategie, respectiv pentru strategia lui Aizenberg, sunt cauzele deosebirii majore între rezultatele celor două strategii.
4. Sensul de parcurgere al cercului trigonometric (orar sau anti-orar) în timpul căutării, pentru noua strategie are un impact major asupra populațiilor obținute, respectiv asupra întregului proces de învățare.

5.1.2. Baza de date IRIS

Baza de date IRIS este o altă bază de date folosită în vederea validării noii strategii de căutare.

Această bază de date este compusă din 150 de instanțe, cu un număr de 4 intrări (n) notate cu X_1, X_2, X_3 , respectiv X_4 și 3 (k) posibile valori ale ieșirii (Y).

În Tabelul 8 este prezentată o parte restrânsă a bazei de date IRIS; această bază de date poate fi descărcată integral de pe site-ul UCI Irvine Machine Learning Repository [30], unde se pot găsi de asemenea informații complete despre fiecare parametru în parte.

Tabelul 8: Baza de date IRIS

| X_1 | X_2 | X_3 | X_4 | Y |
|-------|-------|-------|-------|-----|
| 5.1 | 3.5 | 1.4 | 0.2 | 0 |
| 4.9 | 3 | 1.4 | 0.2 | 0 |
| 4.7 | 3.2 | 1.3 | 0.2 | 0 |
| ... | ... | ... | ... | ... |
| 6.6 | 3 | 4.4 | 1.4 | 1 |
| 6.8 | 2.8 | 4.8 | 1.4 | 1 |
| 6.7 | 3 | 5 | 1.7 | 1 |
| ... | ... | ... | ... | ... |
| 6.3 | 3.3 | 6 | 2.5 | 2 |
| 5.8 | 2.7 | 5.1 | 1.9 | 2 |
| 7.1 | 3 | 5.9 | 2.1 | 2 |
| ... | ... | ... | ... | ... |

În continuare, sunt prezentate rezultatele obținute [48] comparativ cu cele prezentate în literatură [38]. Analiza comparativă a rezultatelor este utilă în identificarea plusurilor și minusurilor unei strategii față de cealaltă.

5.1.2.1. Rezultate comparative

În Tabelul 9 sunt listate rezultatele obținute în timpul validării noi strategii de căutare a sectorului corect în momentul învățării [48]. Rezultatele folosite pentru comparare sunt cele mai bune rezultate raportate în literatură pentru MVN-P [38] până la momentul redactării acestei lucrări. De asemenea, rezultatele prezentate

5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 55

pentru noua strategie sunt cele mai bune obținute în urma rulării a 60 de teste pentru setul de date IRIS. Cu fiecare test, valoarea I a fost incrementată cu 1, plecându-se de la valoarea 2.

Regula folosită pentru corecția ponderilor neuronilor, pentru cele două seturi de rezultate, este diferită; pentru noua strategie [48] au fost folosite reguliile descrise de ecuațiile (1.4) și (1.6) (reguli pentru care s-au obținut cele mai bune rezultate), în vreme ce în literatură [38] regula care a condus la cele mai bune rezultate este descrisă de ecuația (1.7) și strategia 2 de căutare (prezentată în Capitolul 2.4.1); în cazul bazei de date IRIS strategia 1 de căutare propusă de Aizenberg în [38] nu a putut învăța complet problema. De asemenea, pentru setul de rezultate obținute cu noua strategie de căutare, ca direcție de parcurgere a cercului trigonometric a fost ales sensul anti-orar (sensul de parcurgere care produce cele mai bune rezultate), iar pasul de căutare a fost setat la valoarea 1. Nici o altă prelucrare nu a fost aplicată setului de date.

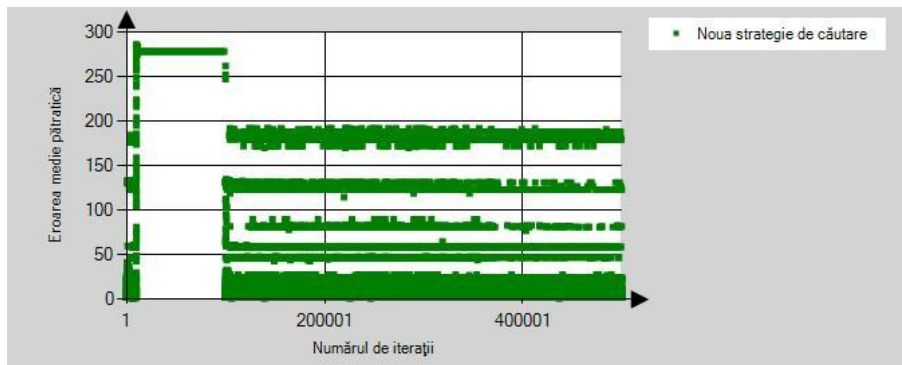
Tabelul 9: Rezultate comparative pentru baza de date IRIS

| Rezultate raportate în literatură | | | | Rezultate obținute cu noua strategie de căutare | | | |
|-----------------------------------|--|------------|---------------|---|----------|------------|---------------|
| Regula de corecție | Iterații | I-periodic | Eficiență (%) | Regula de corecție | Iterații | I-periodic | Eficiență (%) |
| (1.7) | 9 379 027-43 878 728 cu scăderea rapidă a erorii după 50 - 100 iterații | 3 | 100 | (1.4) | 1211 | 5 | 99.333 |
| | | | | (1.4) | 1407 | 8 | 99.333 |
| | | | | (1.4) | 1474 | 6 | 99.333 |
| | | | | (1.4) | 1508 | 7 | 99.333 |
| | | | | (1.4) | 2105 | 10 | 99.333 |
| | | | | (1.6) | 12 | 27 | 99.333 |
| | | | | (1.6) | 12 | 28 | 99.333 |
| | | | | (1.6) | 12 | 29 | 99.333 |
| | | | | (1.6) | 12 | 30 | 99.333 |
| (1.6) | 13 | 17 | 99.333 | | | | |

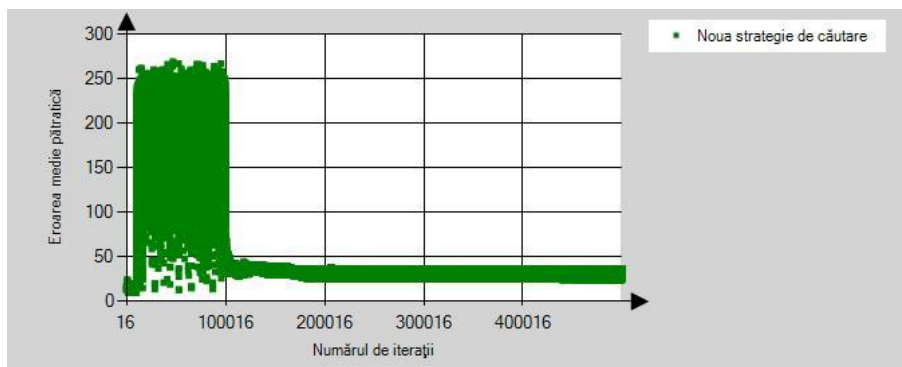
Pentru noua strategie doar cele mai bune 5 rezultate obținute pentru fiecare regulă de corecție sunt prezentate.

În cazul bazei de date IRIS, noua strategie nu a reușit să învețe 100% tot setul de date (spre deosebire de strategia 1 propusă de Aizenberg [38]), chiar dacă numărul maxim de iterații acceptate a fost crescut de la 150000 (valoarea inițială) la 500000 sau chiar 1000000. Cu toate acestea, rezultate foarte bune au fost obținute atât pentru regula de corecție (1.4), dar mai ales pentru regula de corecție (1.6); astfel, o eficiență de 99.333% a fost obținută într-un număr de doar 12 iterații pentru regula de corecție (1.6), respectiv 1211 iterații pentru regula de corecție (1.4).

În Figura 28 este prezentată evoluția de-a lungul învățării a erorii medii pătratice; numărul maxim de iterații a fost setat la valoarea 500000.



a)



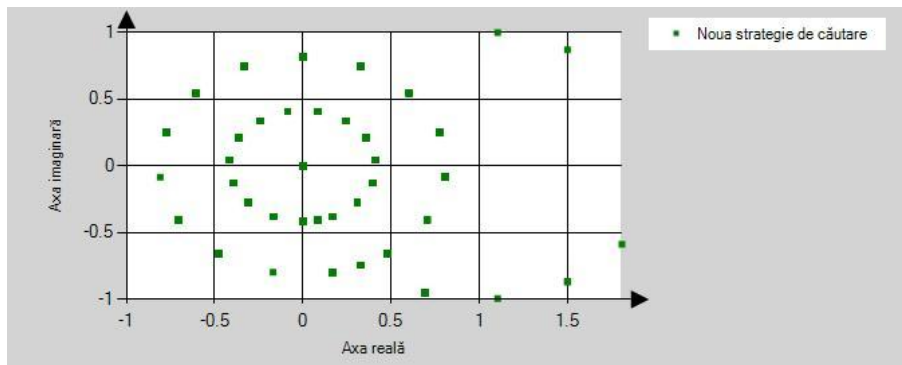
b)

Figura 28: Variația erorii medii pătratice pentru noua strategie de învățare în cazul bazei de date IRIS și a) regula de corecție (1.4), respectiv b) regula de corecție (1.6)

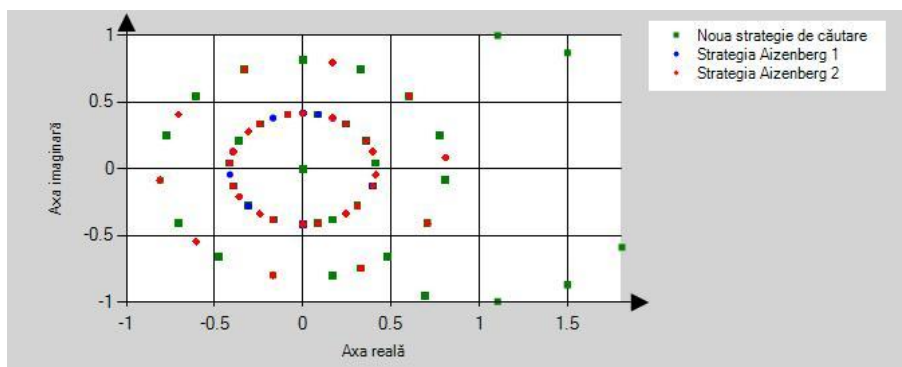
Din Figura 28 a) și b) se poate astfel observa scăderea rapidă a erorii încă din primele iterații [48] pentru ambele reguli de corecție. Mai apoi, eroarea crește semnificativ pentru ambele cazuri, lucru ce denotă o instabilitate a ponderilor MVN-P. Spre sfârșitul învățării, în jurul valorii de 500000 de iterații, 200000 în cazul cel mai fericit al regulii de corecție (1.6), ponderile tind spre stabilizare, consecință fiind scăderea și păstrarea erorii medii pătratice la o valoare medie scăzută.

În Figura 29 sunt ilustrate comparativ variațiile erorii de învățare, descrise prin ecuația (3.5), pentru ambele reguli de corecție, pentru noua strategie, respectiv cele două strategii propuse de Aizenberg (prezentate în Capitolul 1.4.1).

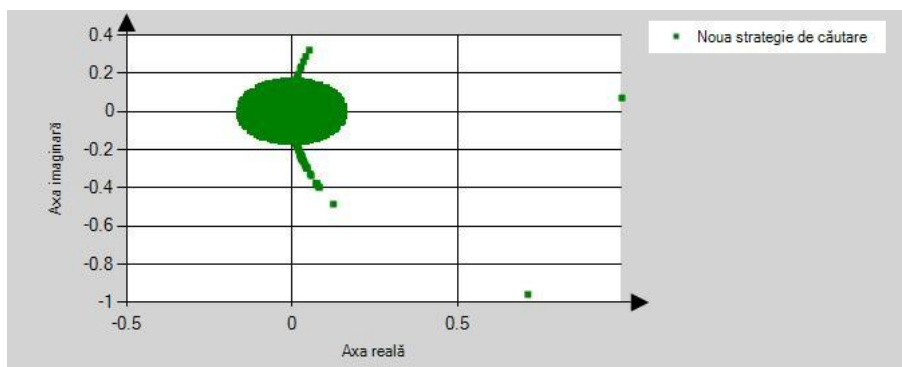
5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 57



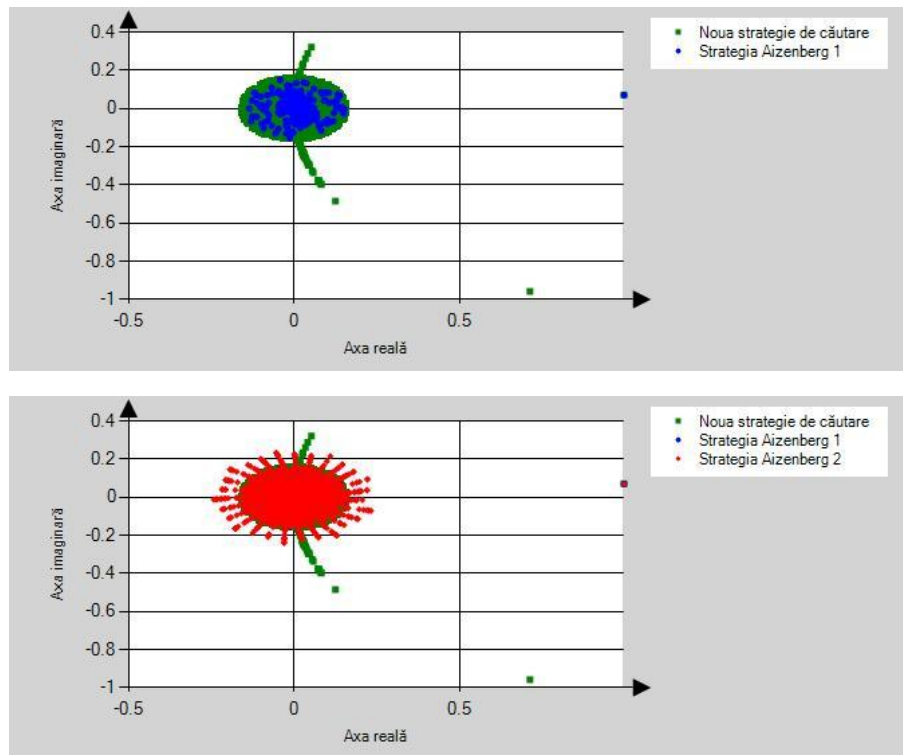
a.1)



a.2)
a) regula de corecție (1.4)



b.1)



b.2)
b) regula de corecție (1.6)

Figura 29: Variația erorii de învățare pentru a) regula de corecție (1.4) și a.1) noua strategie de căutare, respectiv a.2) strategiile lui Aizenberg, b) regula de corecție (1.6) și b.1) noua strategie de căutare, respectiv b.2) strategiile lui Aizenberg

În Figura 29 cu verde este reprezentată eroarea noii strategii propuse [48] iar cu albastru, respectiv roșu, eroarea strategiei 1, respectiv 2 din [38]. Toate cele trei variații ale erorii urmăresc evoluția diferenței ($\varepsilon^q - \varepsilon^s$) de-a lungul învățării. Atât în cazul regulii de corecție (1.4) cât și al regulii de corecție (1.6), spațiul erorilor pentru cele 3 strategii de căutare este asemănător, ponderile evoluând de asemenea asemănător, lucru ce conduce la rezultate apropiate în timpul învățării.

5.1.2.2. Concluzii preliminare

Ca și concluzii preliminare pentru rezultatele prezentate în acest subcapitol se pot formula următoarele:

1. Noua strategie de căutare propusă nu a putut învăța 100% baza de date IRIS.
2. S-a obținut o eficiență de 99.333% în 12 până la 2105 iterații, în funcție de regula de corecție folosită.
3. Eroarea medie pătratică a scăzut rapid încă de la primele iterații.
4. Ponderile MVN-P ating un punct de stabilitate între 200000 și 500000 de iterații în funcție de regula de corecție folosită.

5.1.3. Baza de date „Breast Cancer Wisconsin (Diagnostic)”

Baza de date „Breast Cancer Wisconsin (Diagnostic)” este o altă bază de date folosită în vederea validării noii strategii de căutare.

Această bază de date este compusă din 569 de instanțe, cu un număr de 9 intrări (n) notate cu X_1, \dots, X_9 , și 2 (k) valori posibile ale ieșirii (Y).

În Tabelul 10 este prezentată o parte restrânsă a bazei de date „Breast Cancer Wisconsin (Diagnostic)”; această bază de date poate fi descărcată integral de pe site-ul UCI Irvine Machine Learning Repository [30], unde se pot găsi de asemenea informații complete despre fiecare parametru în parte.

Tabelul 10: Baza de date „Breast Cancer Wisconsin (Diagnostic)”

| X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | X_7 | X_8 | X_9 | Y |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-----|
| 5 | 1 | 1 | 1 | 2 | 1 | 3 | 1 | 1 | 0 |
| 5 | 4 | 4 | 5 | 7 | 10 | 3 | 2 | 1 | 0 |
| 3 | 1 | 1 | 1 | 2 | 2 | 3 | 1 | 1 | 0 |
| 6 | 8 | 8 | 1 | 3 | 4 | 3 | 7 | 1 | 0 |
| 4 | 1 | 1 | 3 | 2 | 1 | 3 | 1 | 1 | 0 |
| 2 | 1 | 1 | 1 | 2 | 1 | 2 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 2 | 10 | 3 | 1 | 1 | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 7 | 8 | 7 | 2 | 4 | 8 | 3 | 8 | 2 | 1 |
| 9 | 5 | 8 | 1 | 2 | 3 | 2 | 1 | 5 | 1 |
| 5 | 3 | 3 | 4 | 2 | 4 | 3 | 4 | 1 | 1 |
| 10 | 3 | 6 | 2 | 3 | 5 | 4 | 10 | 2 | 1 |
| 5 | 5 | 5 | 8 | 10 | 8 | 7 | 3 | 7 | 1 |
| 10 | 5 | 5 | 6 | 8 | 8 | 7 | 1 | 1 | 1 |
| 10 | 6 | 6 | 3 | 4 | 5 | 3 | 6 | 1 | 1 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |

În continuare, sunt prezentate rezultatele obținute comparativ cu cele prezentate în literatură [38]. Analiza comparativă a rezultatelor este utilă în identificarea plusurilor și minusurilor unei strategii față de celelalte.

5.1.3.1. Rezultate comparative

În Tabelul 11 sunt listate rezultatele obținute în timpul validării noii strategii de căutare a sectorului corect în momentul învățării. Rezultatele folosite pentru comparare sunt cele mai bune rezultate raportate în literatură pentru MVN-P [38] până la momentul redactării acestei lucrări. De asemenea, rezultatele prezentate pentru noua strategie sunt cele mai bune obținute în urma rulării a 60 de teste pentru setul de date folosit. Cu fiecare test, valoarea I a fost incrementată cu 1, plecându-se de la valoarea 2.

60 Rezultate experimentale - 5

Regula folosită pentru corecția ponderilor neuronilor, pentru toate seturile de rezultate, este cea descrisă de ecuația (1.4); pentru această regulă s-au obținut în medie cele mai bune rezultate pentru toate strategiile considerate. De asemenea, pentru setul de rezultate obținute cu noua strategie de căutare, ca direcție de parcurgere a cercului trigonometric a fost ales sensul anti-orar (sensul de parcurgere care produce cele mai bune rezultate), iar pasul de căutare a fost setat la valoarea 1. Nici o altă prelucrare nu a fost aplicată setului de date.

Tabelul 11: Rezultate comparative pentru baza de date „Breast Cancer Wisconsin (Diagnostic)”

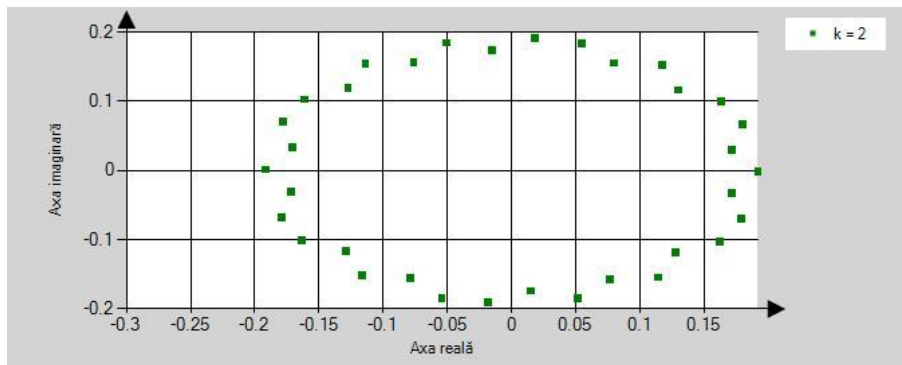
| Rezultate raportate în literatură | | | | | | Rezultate obținute cu noua strategie de căutare | | |
|-----------------------------------|------------|---------------|-------------|------------|---------------|---|------------|---------------|
| Strategia 1 | | | Strategia 2 | | | | | |
| Iterații | l-periodic | Eficiență (%) | Iterații | l-periodic | Eficiență (%) | Iterații | l-periodic | Eficiență (%) |
| 280 - 370 | 2 | 100 | 380 - 423 | 2 | 100 | 1184 | 19 | 95.168 |
| | | | | | | 710 | 29 | 91.654 |
| | | | | | | 260 | 59 | 90.776 |
| | | | | | | 1760 | 30 | 90.19 |
| | | | | | | 415 | 13 | 90.044 |
| | | | | | | 317 | 42 | 89.605 |
| | | | | | | 1279 | 21 | 89.165 |
| | | | | | | 573 | 23 | 88.873 |
| | | | | | | 712 | 2 | 88.726 |
| | | | | | | 651 | 20 | 88.726 |
| | | | | | | 536 | 18 | 88.287 |

În cazul acestei baze de date noua strategie s-a dovedit mai puțin eficientă. Doar cele mai bune 11 rezultate sunt prezentate dintr-un număr total de 60 de rulări independente. Per total o medie de 85.073% eficiență a învățării a fost obținută cu o medie de 744.333 iterații.

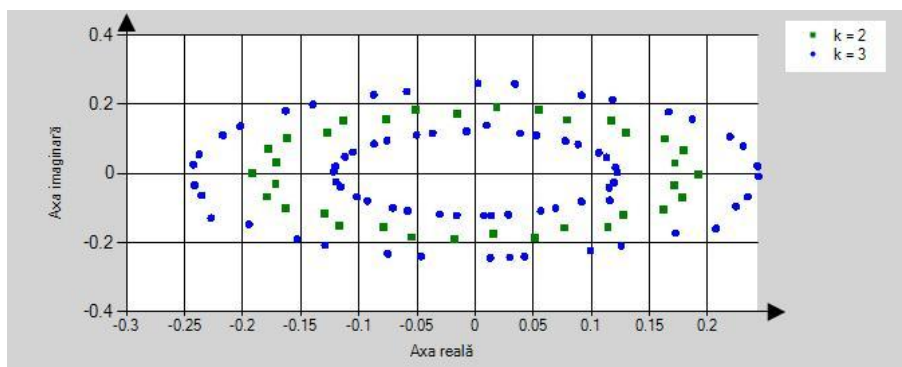
În vederea creșterii eficienței învățării, special pentru acest set de date, s-a adus o modificare procesului de învățare. Astfel, s-a introdus un sector suplimentar, sau cu alte cuvinte s-a introdus o clasă fictivă în spațiul posibilelor valori de ieșire. Cu această modificare, valoarea k a fost crescută de la 2 la 3. În urma rulării aceluiași set de teste, cu aceleași configurații, s-a obținut o eficiență în medie de 97.999% pentru o valoare medie de 732.433 iterații.

În Figura 30 sunt ilustrate comparativ variațiile erorii de învățare, descrise prin ecuația (3.5) pentru $k = 2$, respectiv $k = 3$.

5.1 - Rezultatele produse de noua strategie de căutare la nivelul MVN-P 61



a)



b)

Figura 30: Variația erorii de învățare în cazul bazei de date „Breast Cancer Wisconsin (Diagnostic)”, a) pentru $k = 2$, respectiv b) pentru $k = 3$

În Figura 30 cu verde este reprezentată variația erorii pentru $k = 2$, iar cu albastru variația erorii pentru $k = 3$, pentru noua strategie propusă [48].

Se poate observa faptul că în cazul $k = 3$ spațiul erorilor se restrânge de-a lungul procesului de învățare (diferența $(\varepsilon^q - \varepsilon^s)$ scade treptat) lucru ce conduce la o creștere a eficienței învățării. Introducerea unei clase fictive, în acest caz, a crescut eficiența cu aproximativ 10%.

În capitolele următoare, în momentul analizei eficienței RN cu MVN-P pentru acest set de date, rezultatele obținute în urma modificării lui k vor fi considerate pentru comparare.

În Figura 31 este prezentată evoluția de-a lungul învățării a erorii medii pătratice; numărul maxim de iterații a fost setat la valoarea 150000.

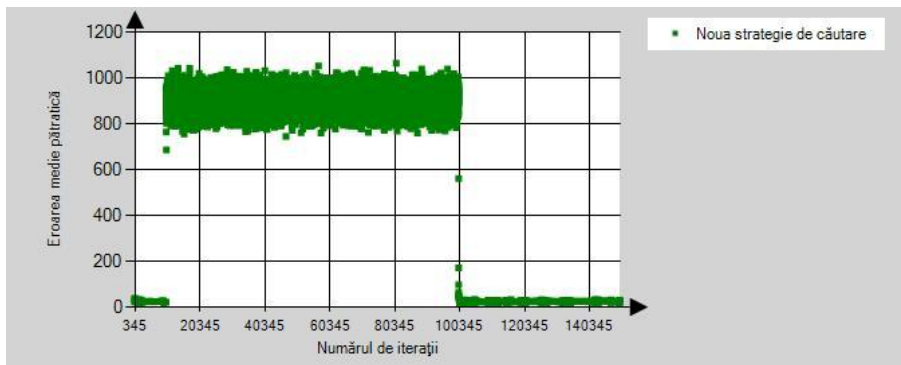


Figura 31: Variația erorii medii pătratice pentru noua strategie de învățare în cazul bazei de date „Breast Cancer Wisconsin (Diagnostic)”

Din Figura 31 se poate astfel observa scăderea rapidă a erorii încă din primele iterații. Mai apoi, eroarea crește semnificativ, lucru ce denotă o instabilitate a ponderilor MVN-P. Această instabilitate conduce la păstrarea unei valori relativ mari a erorii medii de-a lungul procesului de învățare. Spre sfârșitul învățării, în jurul valorii de 100000 de iterații, ponderile tind spre stabilizare, consecință fiind scăderea și păstrarea erorii medii pătratice la o valoare medie scăzută.

5.1.3.2. Concluzii preliminare

Ca și concluzii preliminare pentru rezultatele prezentate în acest subcapitol se pot formula următoarele:

1. Noua strategie de căutare propusă nu a putut învăța 100% baza de date „Breast Cancer Wisconsin (Diagnostic)”.
2. S-a obținut o eficiență medie de 85.073% într-un număr mediu de 744.333 iterații.
3. Crescând numărul de clase ale ieșirii cu o clasă fictivă s-a obținut o eficiență medie de 97.999% într-un număr mediu de 732.433 iterații.
4. Eroarea medie pătratică a scăzut rapid încă de la primele iterații.
5. Ponderile MVN-P ating un punct de stabilitate în jurul valorii de 100000 de iterații.

5.2. Rezultatele produse în urma integrării MVN-P într-o rețea neuronală

În vederea creșterii eficienței MVN-P s-a urmărit integrarea acestuia într-o RN ca neuron pe stratul de ieșire. La momentul redactării acestei lucrări nici o altă încercare de acest fel nu a fost raportată în literatură, reprezentând o noutate în domeniu.

Pentru validarea acestui nou tip de arhitectură de RN, s-au folosit aceleași baze de date ca și în cazul validării noii strategii de căutare. Rezultatele obținute pentru MLMVN-P au fost comparate cu cele obținute pentru un singur MVN-P cu noua strategie de căutare în vederea confirmării, sau nu, a creșterii eficienței învățării și de asemenea pentru determinarea validității acestui tip de RN.

Aceleași configurații (constanta de învățare $C_m (= 1+0i)$, periodicitatea posibilelor soluții l ($2 \leq l \leq 61$), pasul de căutare (având valoarea 1), direcția de căutare (sensul anti-orar), numărul maxim de iterații (150000), respectiv regulile de corecție a ponderilor) ca și în cazul unui singur MVN-P au fost folosite; de asemenea, noua strategie de căutare a fost folosită pentru neuronul de pe stratul de ieșire.

Algoritmul de învățare este prezentat în detaliu în Capitolul 3.2, cu precizarea că spre deosebire de cazul unui singur MVN-P, unde ponderile au fost inițializate cu valoarea $0+0i$, ponderile tuturor neuronilor din RN au fost inițializate cu valoarea $1+0i$.

Pe parcursul validării s-au folosit diverse arhitecturi pentru MLMVN-P, variindu-se atât numărul de straturi ascunse, cât și numărul de neuroni de pe acestea (numiți neuroni ascunși). Ca și notație a unei posibile arhitecturi, pentru mai ușoară înțelegere a cititorului, 3-1-1-1 denotă o RN cu un strat de intrare pe care sunt 3 neuroni, două straturi ascunse cu câte un neuron pe fiecare strat și un strat de ieșire cu un singur neuron; în continuare, această notație va fi folosită pentru simplificare.

5.2.1. Suma modulo k a n valori

5.2.1.1. Rezultate comparative

În Tabelul 12 sunt listate rezultatele obținute în timpul validării MLMVN-P [49]. Rezultatele folosite pentru comparare sunt cele mai bune rezultate obținute pentru un singur MVN-P cu noua regulă de căutare [48]. Doar cele mai bune rezultate obținute, dintre toate arhitecturile MLMVN-P folosite, sunt prezentate (în Anexa 2 se pot găsi rezultatele complete pentru fiecare arhitectură în parte) dintr-un număr total de 60 rulări pentru fiecare set de date în parte. Cu fiecare test, valoarea l a fost incrementată cu 1, plecându-se de la valoarea 2.

Tabelul 12: Rezultate comparative pentru suma modulo k a n valori

| Setul de date | | Rezultate obținute cu un singur MVN-P | | | Rezultate obținute cu MLMVN-P | | | |
|---------------|---|---------------------------------------|------------|---------------|-------------------------------|----------|------------|---------------|
| n | k | Iterații | l-periodic | Eficiență (%) | Arhitectura MLMVN-P | Iterații | l-periodic | Eficiență (%) |
| 2 | 3 | 25 | 49 | 100 | 2-1-1 | 1 | 61 | 100 |
| 2 | 4 | 44 | 7 | 100 | 2-1-1-1-1 | 4 | 36 | 100 |

64 Rezultate experimentale - 5

| | | | | | | | | |
|---|---|--------|----|--------|-----------|------|----|-----|
| 2 | 5 | 144 | 8 | 100 | 2-1-1-1-1 | 16 | 36 | 100 |
| 2 | 6 | 363 | 8 | 100 | 2-1-1-1 | 9 | 56 | 100 |
| 3 | 3 | 54 | 11 | 100 | 3-1-1-1-1 | 3 | 12 | 100 |
| 3 | 4 | 165 | 18 | 100 | 3-2-2-1 | 43 | 28 | 100 |
| 3 | 5 | 3082 | 14 | 100 | 3-1-1-1-1 | 225 | 44 | 100 |
| 3 | 6 | 10014 | 60 | 100 | 3-2-1 | 807 | 60 | 100 |
| 4 | 3 | 83 | 11 | 100 | 4-1-1-1 | 3 | 59 | 100 |
| 4 | 4 | 3317 | 23 | 91.406 | 4-2-1-1 | 316 | 22 | 100 |
| 4 | 5 | 127966 | 24 | 65.440 | 4-1-1-1-1 | 454 | 37 | 100 |
| 4 | 6 | 19857 | 19 | 50.694 | 4-2-2-1 | 2230 | 59 | 100 |
| 5 | 3 | 114392 | 20 | 100 | 5-1-1-1 | 11 | 57 | 100 |
| 5 | 4 | 140021 | 45 | 68.652 | 5-1-1-1-1 | 771 | 43 | 100 |
| 6 | 3 | 119119 | 20 | 92.181 | 6-1-1-1-1 | 2 | 40 | 100 |

Pentru fiecare set de date au fost folosite aceleași arhitecturi de RN. Alegerea acestor arhitecturi (număr de straturi ascunse, respectiv neuroni ascunși) de RN a avut ca și scop pe de o parte, creșterea eficienței învățării, iar pe de altă parte, păstrarea unei structuri relativ simple a MLMVN-P, care să conducă la o durată totală de învățare apropiată cu cea obținută în cazul unui singur MVN-P [50]. Ambele scopuri au fost atinse, iar MLMVN-P și-a dovedit eficiența atât ca și eroare a învățării cât și ca durată de învățare [50]. În Tabelul 12 se poate observa scăderea semnificativă a numărului de iterații necesare învățării cât și creșterea eficienței (până la 100%) pentru seturile de date ce nu au putut fi învățate complet folosind un singur MVN-P. Datorită arhitecturilor simple folosite (cu un număr redus de neuroni ascunși) calculele adiționale introduse pot fi considerate neglijabile; astfel putem considera o echivalență, ca durată de execuție, între o iterație pentru un singur MVN-P, respectiv o iterație pentru MLMVN-P, rezultând astfel a scădere a timpului total de învățare.

Tabelul 13 prezintă analiza statistică (realizată în OriginPro 8 SR0) a celor două populații obținute pentru un singur MVN-P, respectiv MLMVN-P, pentru 3 seturi de date alese aleator; aici, prin populație se înțelege un set de date constituit din totalitatea sectoarelor returnate ca și sectoare corecte în timpul învățării. În cazul MLMVN-P, cele mai bune rezultate obținute au fost alese pentru comparare.

Se poate observa diferența statistică majoră între cele două populații, de unde și rezultatele semnificativ diferite.

Tabelul 13: Analiza statistică descriptivă a populațiilor obținute pentru un singur MVN-P, respectiv pentru MLMVN-P

a) $n = 3, k = 6, l = 60$

a.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|-------------|------------|
| 1 MVN-P | 64800 | 7.16510802 | 19.01448605 | 0.07469595 |
| MLMVN-P | 64800 | 9.16189814 | 21.19065845 | 0.08324476 |
| Diferența | | -1.99679012 | | |

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|-------------|------------|
| 1 MVN-P | 64800 | 5.74853395 | 17.19416038 | 0.06754504 |
| MLMVN-P | 64800 | 8.72719135 | 20.85333119 | 0.08191962 |
| Diferența | | -2.97865740 | | |

5.2 - Rezultatele produse în urma integrării MVN-P într-o rețea neuronală 65

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|-------------|------------|
| 1 MVN-P | 44712 | 5.12477634 | 16.25918903 | 0.07689300 |
| MLMVN-P | 44712 | 8.48819108 | 20.60105627 | 0.09742657 |
| Diferența | | -3.36341474 | | |

a.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|------------|------------|
| 1 MVN-P | 64800 | 2.10279320 | 1.67758970 | 0.00659019 |
| MLMVN-P | 64800 | 2.16171296 | 1.66452260 | 0.00653886 |
| Diferența | | -0.05891975 | | |

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|------------|------------|------------|
| 1 MVN-P | 64800 | 2.08458333 | 1.69245905 | 0.00664860 |
| MLMVN-P | 64800 | 1.98069444 | 1.65275481 | 0.00649263 |
| Diferența | | 0.10388888 | | |

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|------------|------------|------------|
| 1 MVN-P | 44712 | 2.09751297 | 1.70369328 | 0.00805711 |
| MLMVN-P | 44712 | 1.99838969 | 1.65109885 | 0.00780838 |
| Diferența | | 0.09912327 | | |

b) $n = 4, k = 4, l = 22$

b.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|------------|------------|
| 1 MVN-P | 64768 | 1.59101716 | 4.77019158 | 0.01874371 |
| MLMVN-P | 64768 | 5.54057559 | 9.06174053 | 0.03560667 |
| Diferența | | -3.94955842 | | |

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|------------|------------|
| 1 MVN-P | 16128 | 1.30611359 | 3.99978077 | 0.03149531 |
| MLMVN-P | 16128 | 3.38461061 | 3.38461061 | 0.05841110 |
| Diferența | | -2.07849702 | | |

b.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|------------|------------|
| 1 MVN-P | 64768 | 1.42294034 | 1.09618480 | 0.00430728 |
| MLMVN-P | 64768 | 1.50080286 | 1.19078736 | 0.00467901 |
| Diferența | | -0.07786252 | | |

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|------------|------------|
| 1 MVN-P | 16128 | 1.42751736 | 1.09308972 | 0.00860727 |
| MLMVN-P | 16128 | 1.50254216 | 1.10703024 | 0.00871704 |
| Diferența | | -0.07502480 | | |

66 Rezultate experimentale - 5

c) $n = 5, k = 3, l = 57$

c.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

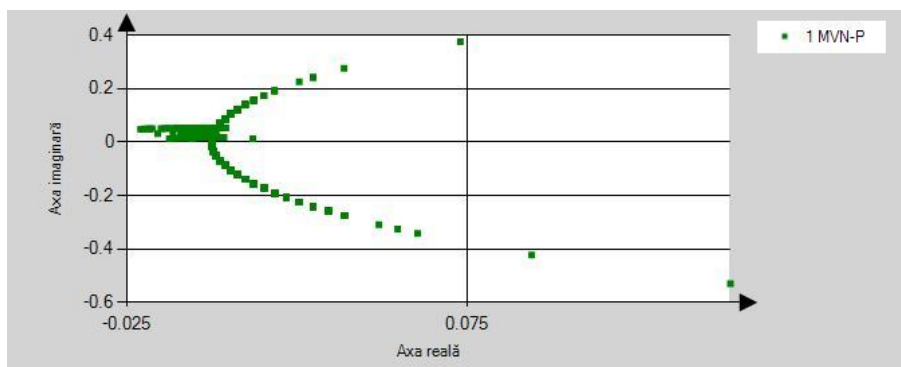
| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|------|-------------|-------------|------------|
| 1 MVN-P | 2673 | 7.78563411 | 18.08581266 | 0.34981509 |
| MLMVN-P | 2673 | 9.42798353 | 19.99828198 | 0.38680600 |
| Diferența | | -1.64234942 | | |

c.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

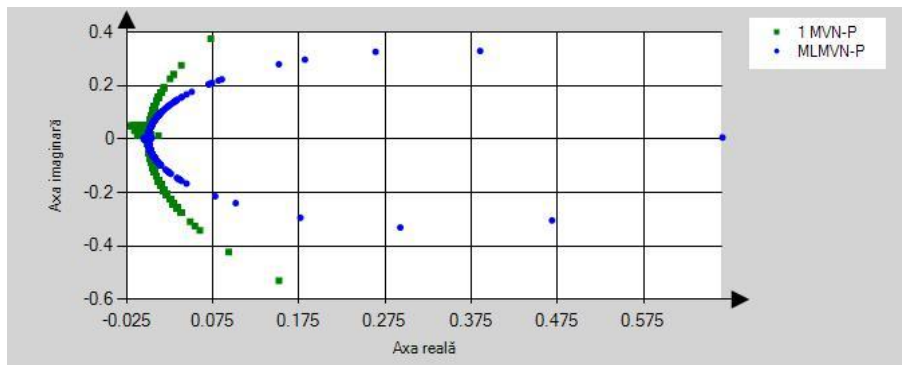
| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|------|------------|------------|------------|
| 1 MVN-P | 2673 | 1.00785634 | 0.81454659 | 0.01575492 |
| MLMVN-P | 2673 | 1.00037411 | 0.81596156 | 0.01578229 |
| Diferența | | 0.00748222 | | |

Se poate observa din Tabelul 13 o diferență statistică (identificată prin parametrul "Diferența" din tabel) majoră între valorile medii statistice (identificate prin parametrul "Mean") a populațiilor domeniilor returnate (Tabelul 13 a.1), b.1) și c.1)). În cazul populațiilor sectoarelor returnate (Tabelul 13 a.2), b.2) și c.2)) această diferență este mai puțin semnificativă, lucru de așteptat datorită folosirii aceleiași strategii de căutare atât pentru cazul MVN-P, respectiv pentru cazul MLMVN-P. Integrarea MVN-P într-o RN a modificat spațiul domeniilor returnate, lucru ce denotă valori total diferite, pentru valorile de ieșire a MVN-P, descrise de ecuația (1.2), respectiv pentru valorile de ieșire MLMVN-P, descrise de ecuația (3.1). În cazul MLMVN-P, diferențele sunt cauzate de straturile ascunse, respectiv neuronii ascunși introduși. Folosirea aceleiași strategii de căutare, la nivelul stratului de ieșire, (format dintr-un singur MVN-P) conduce la rezultate similare ale spațiilor sectoarelor returnate.

În Figura 32, 33, respectiv 34 sunt ilustrate variațiile erorii de învățare, descrise prin ecuația (3.5), pentru aceleași seturi de date, analizate statistic, de mai sus.



a)

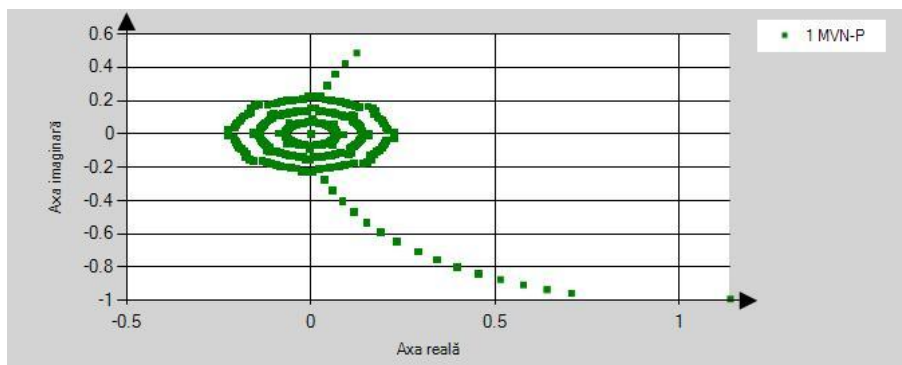


b)

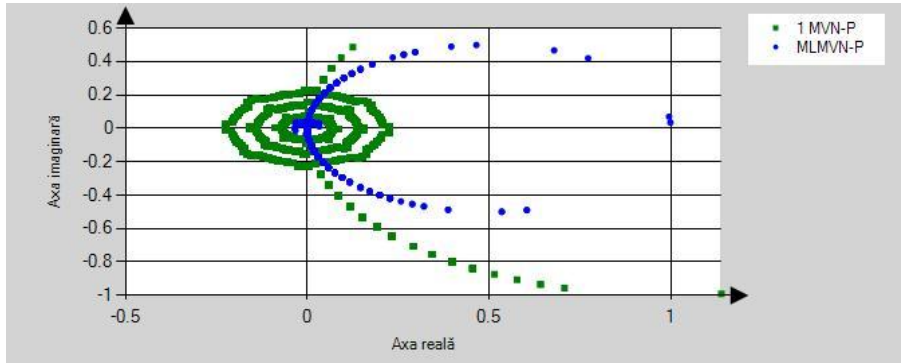
Figura 32: Variația erorii de învățare pentru a) un singur MVN-P, respectiv b) MLMVN-P considerând cazul $n = 3$, $k = 6$, $l = 60$

În Figura 32 cu verde este reprezentată variația erorii pentru un singur MVN-P cu strategia de căutare din [48], iar cu albastru variația erorii pentru MLMVN-P [49] cu strategia de căutare din [48]. În cazul erorii MLMVN-P se poate observa o variație similară a erorii cu cea a erorii MVN-P. Diferența constă în restrângerea spațiului erorii în cazul MLMVN-P. Această eroare este direct proporțională cu diferența ($\varepsilon^q - \varepsilon^s$) și denotă o îmbunătățire a procesului de învățare prin participarea neuronilor ascunși în calculul valorii de ieșire (sectorul s) a neuronului de pe stratul de ieșire. Astfel, sectorul s (cel obținut) este mai apropiat de sectorul q (cel dorit), lucru ce conduce la minimizarea diferenței ($\varepsilon^q - \varepsilon^s$) și creșterea eficienței de învățare.

Concluzii similare se pot formula și pentru distribuțiile erorilor din Figura 33 și Figura 34. Și în cazul acestor seturi de date, spațiul erorilor este mai restrâns pentru MLMVN-P. Se poate observa, din figurile 32, 33 și 34, aceeași specificitate a variației erorii pentru diferite valori ale parametrului l, respectiv ale parametrului k.

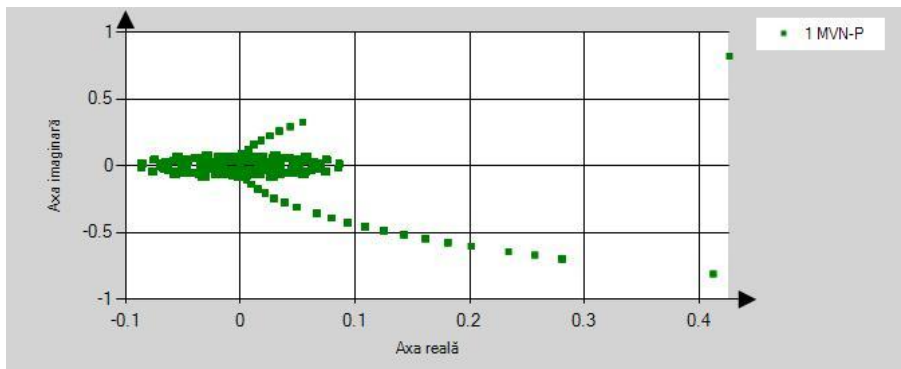


a)

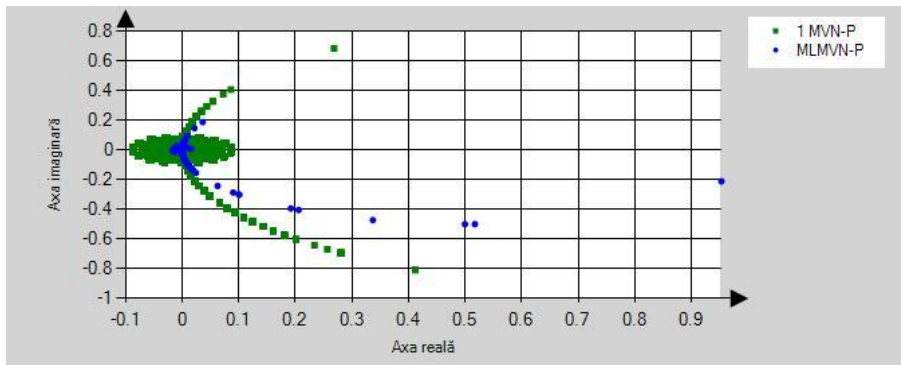


b)

Figura 33: Variația erorii de învățare pentru a) un singur MVN-P, respectiv b) MLMVN-P considerând cazul $n = 4, k = 4, l = 22$



a)



b)

Figura 34: Variația erorii de învățare pentru a) un singur MVN-P, respectiv b) MLMVN-P considerând cazul $n = 5, k = 3, l = 57$

Aceleași observații, ca cele făcute în cazul figurilor 31, 32 și 33, sunt valabile pentru toate seturile de date considerate (vezi Tabelul 12). Restrângerea spațiului erorii de ieșire, descrisă de ecuația (3.5) este un aspect general valabil și specific MLMVN-P [49] și este cauzată de influența majoră a neuronilor ascunși în calculul valorii de ieșire a MLMVN-P.

5.2.1.2. Concluzii preliminare

Ca și concluzii preliminare pentru rezultatele prezentate în acest subcapitol se pot formula următoarele:

1. Integrarea MVN-P ca și neuron pe stratul de ieșire a dus la creșterea eficienței învățării.
2. Eficiența a crescut atât din punct de vedere al erorii de învățare cât și al duratei totale de învățare.
3. În cazul MLMVN-P, spațiul erorilor este mai restrâns decât cel al erorilor unui singur MVN-P.

5.2.2. Baza de date IRIS

5.2.2.1. Rezultate comparative

În Tabelul 14 sunt listate rezultatele obținute în timpul validării MLMVN-P. Rezultatele folosite pentru comparare sunt cele mai bune rezultate obținute pentru un singur MVN-P cu noua regulă de căutare [48]. Doar cele mai bune rezultate obținute pentru MLMVN-P sunt prezentate.

Tabelul 14: Rezultate comparative pentru baza de date IRIS

| Rezultate obținute cu un singur MVN-P | | | | Rezultate obținute cu MLMVN-P | | | | |
|---------------------------------------|----------|------------|---------------|-------------------------------|---------------------|----------|------------|---------------|
| Regula de corecție | Iterații | I-periodic | Eficiență (%) | Regula de corecție | Arhitectura MLMNV-P | Iterații | I-periodic | Eficiență (%) |
| (1.4) | 1211 | 5 | 99.333 | (1.4) | 4-1-1 | 3091 | 23 | 99.333 |
| | | | | | 4-2-1 | 664 | 23 | 99.333 |
| | | | | | 4-3-1 | 1173 | 21 | 99.333 |
| | | | | | 4-1-1-1 | 879 | 33 | 99.333 |
| | | | | | 4-1-2-1 | 302 | 39 | 98.667 |
| | | | | | 4-2-1-1 | 302 | 39 | 98.667 |
| | | | | | 4-2-2-1 | 3793 | 13 | 99.333 |
| | | | | | 4-1-1-1-1 | 253 | 36 | 99.333 |
| (1.6) | 12 | 27 | 99.333 | (1.6) | 4-1-1 | 27 | 2 | 99.333 |
| | | | | | 4-2-1 | 289 | 3 | 99.333 |
| | | | | | 4-3-1 | 904 | 22 | 99.333 |
| | | | | | 4-1-1-1 | 40 | 42 | 99.333 |
| | | | | | 4-1-2-1 | 20 | 31 | 99.333 |
| | | | | | 4-2-1-1 | 100 | 12 | 99.333 |
| | | | | | 4-2-2-1 | 80 | 10 | 99.333 |

70 Rezultate experimentale - 5

| | | | | | | | | |
|--|--|--|--|--|-----------|-----|----|--------|
| | | | | | 4-1-1-1-1 | 460 | 53 | 99.333 |
|--|--|--|--|--|-----------|-----|----|--------|

Se poate observa o îmbunătățire a eficienței învățării în cazul MLMVN-P și regula de corecție (1.4), îmbunătățire ce se referă mai degrabă la durata totală de învățare decât la eroarea de învățare. În cazul regulii de corecție (1.6) eficiența MLMVN-P a fost mai scăzută, numărul de iterații crescând.

Tabelul 15 prezintă analiza statistică (realizată în OriginPro 8 SR0) a celor două populații obținute pentru un singur MVN-P, respectiv MLMVN-P, pentru ambele reguli de corecție. Pentru analiză, în cazul MLMVN-P s-a luat în considerare arhitectura care a produs cele mai bune rezultate.

Tabelul 15: Analiza statistică descriptivă a populațiilor obținute pentru un singur MVN-P, respectiv pentru MLMVN-P

a) $n = 4, k = 3, l = 36$, regula de corecție (1.4)

a.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|------------|------------|
| 1 MVN-P | 37950 | 0.28252964 | 2.23005988 | 0.01144749 |
| MLMVN-P | 37950 | 0.41596837 | 3.71657731 | 0.01907819 |
| Diferența | | -0.13343873 | | |

a.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|------------|------------|------------|
| 1 MVN-P | 37950 | 1.00050065 | 0.82253611 | 0.00422230 |
| MLMVN-P | 37950 | 0.97992094 | 0.82371596 | 0.00422835 |
| Diferența | | 0.02057971 | | |

b) $n = 4, k = 3, l = 27$, regula de corecție (1.6)

b.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|------|-------------|-------------|------------|
| 1 MVN-P | 1800 | 0.20222222 | 2.28468462 | 0.05385053 |
| MLMVN-P | 1800 | 18.05722222 | 11.14189411 | 0.26261696 |
| Diferența | | -17.855 | | |

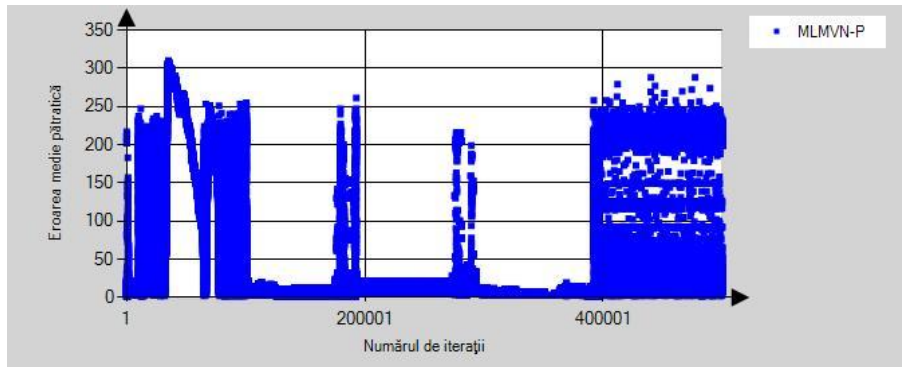
b.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|------|-------------|------------|------------|
| 1 MVN-P | 1800 | 0.96666667 | 0.81536113 | 0.01921824 |
| MLMVN-P | 1800 | 1.07222222 | 0.82303250 | 0.01939906 |
| Diferența | | -0.10555555 | | |

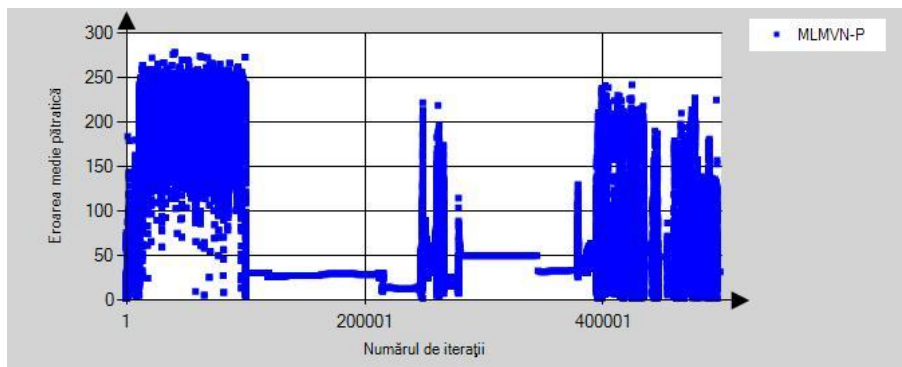
Din nou se poate observa, ca și în cazul datelor din Tabelul 13, o diferență semnificativă între populațiile domeniilor returnate și o diferență mai puțin semnificativă între populațiile sectoarelor returnate. Și în acest caz influența neuronilor ascunși este semnificativă și se observă din spațiul domeniilor returnate, în vreme ce folosirea aceleiași tehnici de căutare, la nivelul MVN-P de pe stratul de ieșire, ca și în cazul unui singur MVN-P, conduce la sectoare similare returnate.

5.2 - Rezultatele produse în urma integrării MVN-P într-o rețea neuronală 71

În Figura 35 este prezentată evoluția de-a lungul învățării a erorii medii pătratice obținută pentru cele mai bune rezultate (vezi Tabelul 14); numărul maxim de iterații a fost setat la valoarea 500000.



a)



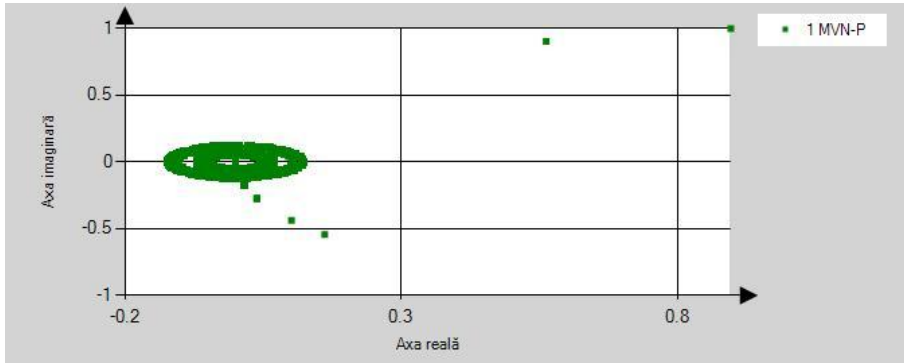
b)

Figura 35: Variația erorii medii pătratice pentru MLMVN-P în cazul bazei de date IRIS și a) regula de corecție (1.4), respectiv b) regula de corecție (1.6)

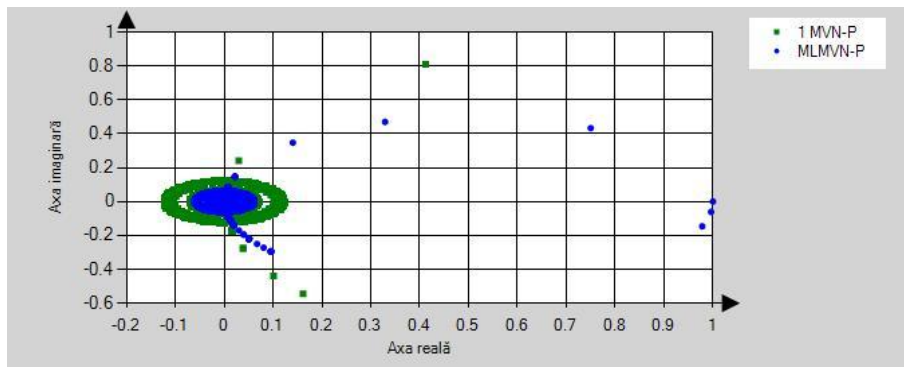
Din Figura 35 a) și b) se poate astfel observa scăderea rapidă a erorii medii pătratice încă din primele iterații pentru ambele reguli de corecție, lucru remarcabil și în cazul unui singur MVN-P (Figura 28) [48]. Mai apoi, eroarea crește semnificativ pentru ambele cazuri, lucru ce denotă o instabilitate a ponderilor MVN-P. Pentru ambele reguli de corecție se poate identifica, din Figura 35, un interval în care eroarea scade semnificativ și se păstrează la o valoare constantă, ceea ce indică o stabilizare temporară a ponderilor. Spre sfârșitul învățării ponderile se destabilizează treptat pentru ambele reguli de corecție, eroarea medie ajungând și păstrându-se la o valoare relativ ridicată.

În figura 36 se prezintă comparativ, distribuția erorii descrisă de ecuația (3.5) pentru cazul unui singur MVN-P (verde), respectiv pentru cazul MLMVN-P (albastru), pentru ambele reguli de corecție considerate și configurațiile (valoarea parametrului l) din Tabelul 15.

72 Rezultate experimentale - 5

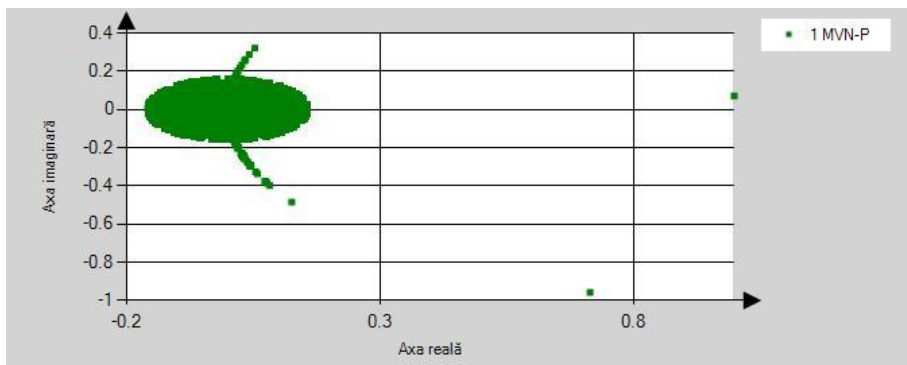


a.1)

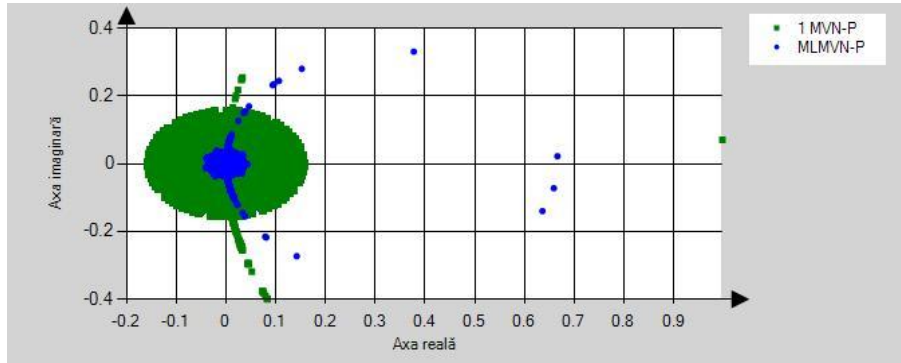


a.2)

a) regula de corecție (1.4), $l = 36$



b.1)



b.2)
b) regula de corecție (1.6), $l = 27$

Figura 36: Variația erorii de învățare pentru a) regula de corecție (1.4) și a.1) un singur MVN-P, respectiv a.2) MLMVN-P, b) regula de corecție (1.6) și b.1) un singur MVN-P, respectiv b.2) MLMVN-P

Se poate remarca în Figura 36 a.2) o valoare a erorii mai scăzută în cazul MLMVN-P pentru regula de corecție (1.4). Acest lucru poate fi observat prin faptul că spațiul erorilor MLMVN-P este mai restrâns și inclus în spațiul erorilor pentru un singur MVN-P (lucru observat și în figurile 32, 33 și 34). Datorită acestui aspect, procesul de învățare este îmbunătățit în cazul regulii de corecție (1.4), iar numărul de iterații redus pentru obținerea aceleiași eficiențe (eroare de învățare). Pentru regula de corecție (1.6) (Figura 36 b.2)) eficiențele (aici prin eficiență ne referim strict la numărul de iterații) MVN-P și MLMVN-P sunt apropiate pentru anumite arhitecturi considerate, dar nu putem afirma că integrarea MVN-P a condus la rezultate îmbunătățite. Pentru acest caz, spațiul erorii pentru MLMVN-P nu este inclus în totalitate în spațiul erorii MVN-P, ca pentru situațiile observate în figurile 32, 33, 34 și 36 a.2).

5.2.2.2. Concluzii preliminare

Ca și concluzii preliminare pentru rezultatele prezentate în acest subcapitol se pot formula următoarele:

1. Integrarea MVN-P ca și neuron pe stratul de ieșire a dus la creșterea eficienței învățării în cazul regulii de corecție (1.4).
2. Eficiența a crescut doar din punct de vedere al duratei totale de învățare (numărul de iterații a fost redus) și nu din punct de vedere al erorii de învățare.
3. În cazul regulii de corecție (1.6) și a MLMVN-P, ponderile MVN-P nu ating un punct de stabilizare, astfel că eroarea medie păstrează o valoare relativ ridicată spre sfârșitul învățării.

5.2.3. Baza de date "Breast Cancer Wisconsin (Diagnostic)"

5.2.3.1. Rezultate comparative

În Tabelul 16 sunt listate rezultatele obținute în timpul validării MLMVN-P. Rezultatele folosite pentru comparare sunt cele mai bune rezultate obținute pentru un singur MVN-P cu noua regulă de căutare [48]. Doar cele mai bune rezultate obținute pentru MLMVN-P sunt prezentate.

Tabelul 16: Rezultate comparative pentru baza de date „Breast Cancer Wisconsin (Diagnostic)”

| Rezultate obținute cu un singur MVN-P | | | Rezultate obținute cu MLMVN-P | | | |
|---------------------------------------|------------|---------------|-------------------------------|----------|------------|---------------|
| Iterații | l-periodic | Eficiență (%) | Arhitectura MLMNV-P | Iterații | l-periodic | Eficiență (%) |
| 107 | 17 | 98.097 | 9-1-1 | 281 | 23 | 97.950 |
| | | | 9-2-1 | 1602 | 6 | 98.097 |
| | | | 9-3-1 | 1032 | 7 | 98.097 |
| | | | 9-1-1-1 | 488 | 53 | 98.097 |
| | | | 9-1-2-1 | 818 | 41 | 98.097 |
| | | | 9-2-1-1 | 306 | 30 | 98.097 |
| | | | 9-2-2-1 | 198 | 8 | 98.097 |
| | | | 9-1-1-1-1 | 42 | 30 | 97.950 |

În cazul bazei de date „Breast Cancer Wisconsin (Diagnostic)” integrarea MVN-P nu a condus la o îmbunătățire a eficienței învățării. S-au obținut aceleași valori ale erorii de învățare, însă într-un număr mai mare de iterații.

Tabelul 17 prezintă analiza statistică (realizată în OriginPro 8 SR0) a celor două populații obținute pentru un singur MVN-P, respectiv MLMVN-P. Pentru analiză, în cazul MLMVN-P s-a luat în considerare arhitectura care a produs cele mai bune rezultate.

Tabelul 17: Analiza statistică descriptivă a populațiilor obținute pentru un singur MVN-P, respectiv pentru MLMVN-P

a) $n = 9, k = 3, l = 17$

a.1) Populația tuturor domeniilor returnate ($0 \leq \text{domeniu} < l$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|------------|------------|
| 1 MVN-P | 64885 | 0.12679355 | 1.41801115 | 0.00556682 |
| MLMVN-P | 64885 | 2.06899899 | 5.20067976 | 0.02041681 |
| Diferența | | -1.94220544 | | |

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|------|-------------|------------|------------|
| 1 MVN-P | 8196 | 0.07027818 | 1.05813446 | 0.01168799 |
| MLMVN-P | 8196 | 0.96876525 | 3.80757415 | 0.04205788 |
| Diferența | | -0.89848706 | | |

5.2 - Rezultatele produse în urma integrării MVN-P într-o rețea neuronală 75

a.2) Populația tuturor sectoarelor returnate ($0 \leq \text{sector} < k$)

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|-------|-------------|------------|------------|
| 1 MVN-P | 64885 | 0.37771441 | 0.50193778 | 0.00197050 |
| MLMVN-P | 64885 | 0.68767819 | 0.75950097 | 0.00298164 |
| Diferența | | -0.30996378 | | |

| Structura folosită pentru învățare | N | Mean | SD | SEM |
|------------------------------------|------|-------------|------------|------------|
| 1 MVN-P | 8196 | 0.36517813 | 0.49054785 | 0.00541851 |
| MLMVN-P | 8196 | 0.56100536 | 0.70839964 | 0.00782487 |
| Diferența | | -0.19582723 | | |

Din nou se poate observa, ca și în cazul datelor din Tabelul 13 și Tabelul 15, o diferență semnificativă între populațiile domeniilor returnate (cauzată de influența neuronilor ascunși) și o diferență mai puțin semnificativă între populațiile sectoarelor returnate (cauzată de folosirea aceleiași tehnici de căutare, la nivelul MVN-P de pe stratul de ieșire).

În Figura 37 este prezentată evoluția de-a lungul învățării a erorii medii pătratice obținută pentru cea mai eficientă arhitectură (vezi Tabelul 16); numărul maxim de iterații a fost setat la valoarea 150000.

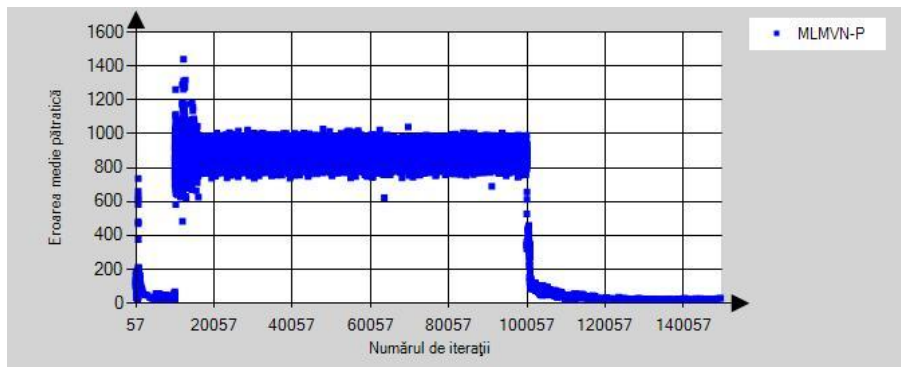
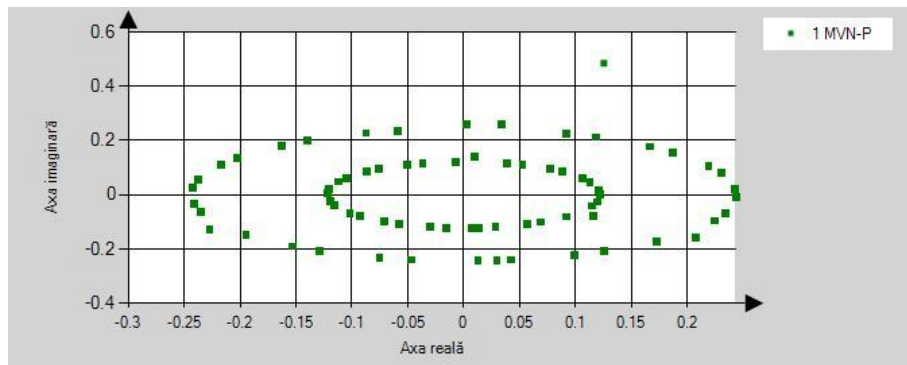


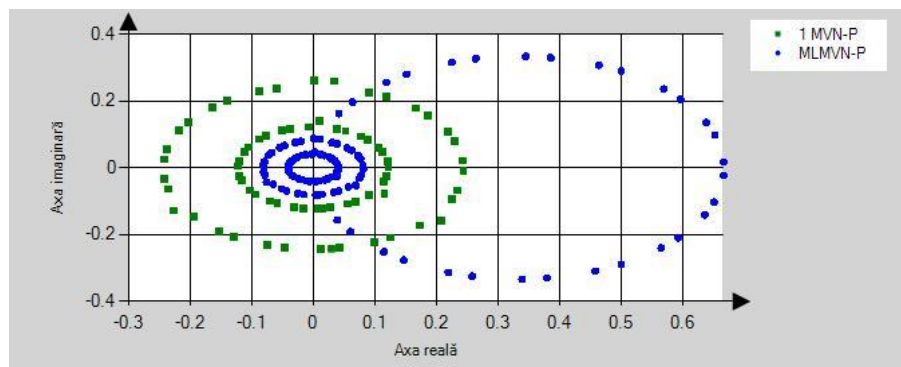
Figura 37: Variația erorii medii pătratice pentru noua strategie de învățare în cazul bazei de date „Breast Cancer Wisconsin (Diagnostic)”

Din Figura 37 se poate astfel observa o evoluție a erorii medii pătratice în cazul MLMVN-P asemănătoare cu evoluția erorii pentru un singur MVN-P (Figura 31). Eroarea scade brusc la început, mai apoi variază într-o plajă largă de valori, până la 100000 de iterații, moment din care începe să se stabilizeze, având o valoare constantă până spre sfârșitul învățării.

În figura 38 se prezintă comparativ, distribuția erorii descrisă de ecuația (3.5) pentru cazul unui singur MVN-P (verde), respectiv pentru cazul MLMVN-P (albastru) și configurațiile (valoarea parametrului l) din Tabelul 17.



a)



b)

Figura 38: Variația erorii de învățare pentru a) un singur MVN-P, respectiv b) MLMVN-P.

În Figura 38 b) se pot remarca două aspecte importante: pe de-o parte spațiul erorii MLMVN-P este parțial inclus în spațiul erorii MVN-P (ca și în situațiile observate în figurile 32, 33, 34 și 36 a.2)), iar pe de altă parte spațiul erorii MLMVN-P variază parțial în afara spațiului erorii MVN-P, lucru similar cu situația observată în Figura 36 b.2). Ca și în cazul bazei de date IRIS, această variație conduce la o creștere a numărului de iterații necesare, în cazul MLMVN-P, pentru obținerea aceleiași eficiențe ca și în cazul MVN-P.

5.2.3.2. Concluzii preliminare

Ca și concluzii preliminare pentru rezultatele prezentate în acest subcapitol se pot formula următoarele:

1. Integrarea MVN-P ca și neuron pe stratul de ieșire nu a condus la creșterea eficienței învățării.
2. S-au obținut valori asemănătoare ale erorii de învățare într-un număr mai mare de iterații.

3. Variația erorii medii pătratice denotă o evoluție asemănătoare a ponderilor MVN-P ca și în cazul unui singur MVN-P.
4. În cazul MLMVN-P, variația erorii, de la începutul învățării, este cauza principală a creșterii numărului necesar de iterații pentru obținerea aceleiași eficiențe ca și în cazul MVN-P.

5.3. Concluzii

În Capitolul 5 au fost prezentate rezultatele obținute în urma aplicării conceptelor teoretice din Capitolul 3, respectiv Capitolul 4. În cazul noii strategii de căutare a sectorului corect, rezultatele obținute au fost prezentate comparativ cu cele din literatură pentru ambele strategii de căutare din [38]. În cazul MLMVN-P rezultatele obținute au fost prezentate comparativ cu cele obținute pentru un singur MVN-P, în ambele implementări (un singur MVN-P, respectiv MLMVN-P) fiind folosită noua strategie de căutare propusă.

Noua strategie de căutare a produs rezultate bune, în anumite cazuri semnificativ mai bune decât cele două strategii din [38]. În cazul seturilor de date "suma modulo k a n valori" cu noua strategie de căutare, numărul de iterații a scăzut, cu câteva excepții evidențiate. În cazul bazelor de date "IRIS" și "Breast Cancer Wisconsin (Diagnostic)" cu noua strategie, învățarea nu a produs o eficiență de 100%. Cu toate acestea, o eficiență de peste 95% a fost obținută pentru ambele seturi de date, această valoare încadrându-se în marja de eroare dorită (5-10%). De asemenea s-a analizat, comparativ, eficiența noii strategii de căutare pentru sensul orar, respectiv anti-orar, ajungându-se la concluzia că sensul anti-orar conduce la o eficiență mai mare a învățării.

Pentru MLMVN-P rezultatele indică o creștere a eficienței învățării (durata de învățare a scăzut, respectiv rata erorii a scăzut la 0) pentru toate seturile de date "suma modulo k a n valori". Pentru bazele de date "IRIS" și "Breast Cancer Wisconsin (Diagnostic)" MLMVN-P nu a condus la o îmbunătățire semnificativă a procesului de învățare, rata erorii păstrându-se la aceeași valoare. Cu toate acestea s-a observat o variație a erorii mai mică în timpul învățării.

Ca și concluzie pentru MLMVN-P, integrarea MVN-P într-o rețea neuronală (cu folosirea unor funcții specifice de activare, respectiv corecție a ponderilor) a fost realizată cu succes, ducând la creșterea suplimentară, în majoritatea situațiilor, a eficienței MVN-P, obiectivul stabilit în Capitolul 4 fiind atins.

6. CONCLUZII

În acest capitol se prezintă concluziile finale ale tezei de față, identificându-se contribuțiile proprii, cât și viitoare posibile direcții de dezvoltare ale conceptelor propuse.

6.1. Îndeplinirea obiectivelor propuse

Obiectivul principal al studiului doctoral de față l-a constituit creșterea eficienței neuronului cu ponderi numere complexe și funcție periodică de activare (MVN-P).

MVN-P a apărut ca o evoluție a MVN prin introducerea unui parametru de periodicitate a posibilelor sectoare/soluții. MVN-P și-a demonstrat eficiența în varii domenii ca prelucrare de imagini, clasificare sau predicție, rezultatele obținute putând fi consultate în partea bibliografică. Având în vedere eficiența crescută a MVN-P, cât și noutatea acestui tip de neuron, s-a dorit analiza în detaliu a acestui concept pentru a fi identificate eventuale posibile îmbunătățiri în domeniu.

În vederea îndeplinirii acestui obiectiv, s-a analizat în detaliu eficiența procesului actual de antrenare a MVN-P. S-au analizat două strategii de antrenare prezentate în literatură, identificându-se avantajele, respectiv dezavantajele fiecăreia în parte. După finalizarea acestei analize s-a propus o modificare a strategiei de învățare curente prin definirea unei noi tehnici de căutare a sectorului corect. Pentru noua tehnică de căutare propusă s-a dorit reducerea timpului de învățare cu păstrarea, în același timp, a unei erori reduse.

Pentru validarea noii strategii de căutare s-au realizat studii comparative (ale erorii învățării, cât și a populațiilor statistice obținute pentru diverse configurații), rezultatele obținute fiind prezentate, în detaliu, în Capitolul 5.1. Aceste rezultate indică faptul că noua strategie propusă este validă, iar folosirea ei a dus la îmbunătățirea, chiar semnificativă în unele cazuri, a eficienței MVN-P.

Pentru a crește și mai mult eficiența MVN-P, următorul pas a fost integrarea acestuia ca neuron pe stratul de ieșire a unei RN multi-strat (MLMVN-P). Pe stratul de intrare, respectiv straturile ascunse au fost folosiți MVN. Regula de corecție, a ponderilor neuronilor, folosită a fost bazată pe învățarea supervizată, adaptată specificității MVN, respectiv MVN-P.

În vederea analizei eficienței MLMVN-P s-au realizat, de asemenea, studii comparative (ale erorii învățării, cât și a populațiilor statistice obținute pentru diverse configurații), rezultatele fiind prezentate în Capitolul 5.2. Și în acest caz, rezultatele indică o creștere suplimentară a eficienței învățării, spațiul erorilor fiind mai restrâns în cazul MLMVN-P decât în cazul MVN-P, în majoritatea situațiilor analizate. De asemenea, ca o precizare suplimentară, eficiența a fost îmbunătățită atât din punct de vedere al erorii, cât și a duratei de învățare, datorită folosirii unor arhitecturi simple de RN.

6.2. Contribuțiile proprii

Ca și contribuții proprii se pot identifica următoarele:

1. Propunerea unei tehnici noi de căutare a sectorului/soluției corecte [48]. Noua tehnică se bazează pe căutarea anti-orară, iar dacă sectorul corect nu este găsit după un număr de încercări (direct proporțional cu numărul de intrări) sectorul returnat ca și corect este sectorul 0 (echivalent clasei 0); tehnica de căutare propusă ajută la eficientizarea procesului de învățare (numărul de iterații este redus).
2. Identificarea plajei optime de valori a parametrului "periodicitate" pentru posibilele sectoare/soluții pentru o serie de seturi de date [50].
3. Integrarea MVN-P într-o RN cu structură configurabilă [50] – acest lucru reprezintă o noutate în domeniu până la data redactării acestei teze de doctorat.
4. Analiza eficienței MLMVN-P comparativ cu cea a unui singur MVN-P [50].
5. Identificarea necesității folosirii tehnicii de învățare cu exemple negative [45] pentru cazul MVN-P, respectiv MLMVN-P, în vederea creșterii suplimentare a eficienței învățării.
6. Identificarea necesității inserării, unde este nevoie, a unui sector suplimentar (ieșire fictivă), pentru creșterea eficienței învățării; un astfel de procedeu și rezultatele obținute sunt prezentate în Capitolul 5.1.3.

6.3. Direcții viitoare de cercetare

Ca și direcții noi de dezvoltare se pot identifica următoarele:

1. Aplicarea conceptului de neuron block [47] în cazul MVN-P, respectiv MLMVN-P.
2. Aplicarea învățării nesupervizate, (e.x. învățarea hebbiană) pentru cazul MVN-P, respectiv MLMVN-P; în [44] un model de învățare hebbiană a fost aplicat pentru cazul MVN.
3. Aplicarea conceptului de cuaternion (numere complexe extinse) pentru cazul MVN-P.

Anexa 1 – Suma modulo k a n valori

Tabelul 18: Suma modulo 4 a 3 valori

| x_1 | x_2 | x_3 | $f(x_1, x_2, x_3)$ |
|-------|-------|-------|--------------------|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 2 | 2 |
| 0 | 0 | 3 | 3 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 2 |
| 0 | 1 | 2 | 3 |
| 0 | 1 | 3 | 0 |
| 0 | 2 | 0 | 2 |
| 0 | 2 | 1 | 3 |
| 0 | 2 | 2 | 0 |
| 0 | 2 | 3 | 1 |
| 0 | 3 | 0 | 3 |
| 0 | 3 | 1 | 0 |
| 0 | 3 | 2 | 1 |
| 0 | 3 | 3 | 2 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 2 |
| 1 | 0 | 2 | 3 |
| 1 | 0 | 3 | 0 |
| 1 | 1 | 0 | 2 |
| 1 | 1 | 1 | 3 |
| 1 | 1 | 2 | 0 |
| 1 | 1 | 3 | 1 |
| 1 | 2 | 0 | 3 |
| 1 | 2 | 1 | 0 |
| 1 | 2 | 2 | 1 |
| 1 | 2 | 3 | 2 |
| 1 | 3 | 0 | 0 |
| 1 | 3 | 1 | 1 |
| 1 | 3 | 2 | 2 |
| 1 | 3 | 3 | 3 |
| 2 | 0 | 0 | 2 |
| 2 | 0 | 1 | 3 |
| 2 | 0 | 2 | 0 |
| 2 | 0 | 3 | 1 |
| 2 | 1 | 0 | 3 |
| 2 | 1 | 1 | 0 |
| 2 | 1 | 2 | 1 |
| 2 | 1 | 3 | 2 |
| 2 | 2 | 0 | 0 |
| 2 | 2 | 1 | 1 |
| 2 | 2 | 2 | 2 |
| 2 | 2 | 3 | 3 |
| 2 | 3 | 0 | 1 |
| 2 | 3 | 1 | 2 |
| 2 | 3 | 2 | 3 |
| 2 | 3 | 3 | 0 |

Anexa 1 – Suma modulo k a n valori 81

| | | | |
|---|---|---|---|
| 3 | 0 | 0 | 3 |
| 3 | 0 | 1 | 0 |
| 3 | 0 | 2 | 1 |
| 3 | 0 | 3 | 2 |
| 3 | 1 | 0 | 0 |
| 3 | 1 | 1 | 1 |
| 3 | 1 | 2 | 2 |
| 3 | 1 | 3 | 3 |
| 3 | 2 | 0 | 1 |
| 3 | 2 | 1 | 2 |
| 3 | 2 | 2 | 3 |
| 3 | 2 | 3 | 0 |
| 3 | 3 | 0 | 2 |
| 3 | 3 | 1 | 3 |
| 3 | 3 | 2 | 0 |
| 3 | 3 | 3 | 1 |

Anexa 2 – MLMVN-P rezultate complete pentru "Suma modulo k a n valori"

Tabelul 19: Rezultate comparative complete pentru suma modulo k a n valori

| Setul de date | | Rezultate obținute cu un singur MVN-P | | | Rezultate obținute cu MLMVN-P | | | |
|---------------|---|---------------------------------------|------------|---------------|-------------------------------|----------|------------|---------------|
| n | k | Iterații | l-periodic | Eficiență (%) | Arhitectura MLMVN-P | Iterații | l-periodic | Eficiență (%) |
| 2 | 3 | 25 | 49 | 100 | 2-1-1 | 1 | 61 | 100 |
| | | | | | 2-2-1 | 9 | 10 | |
| | | | | | 2-3-1 | 23 | 13 | |
| | | | | | 2-1-1-1 | 1 | 36 | |
| | | | | | 2-1-2-1 | 2 | 38 | |
| | | | | | 2-2-1-1 | 1 | 51 | |
| | | | | | 2-2-2-1 | 2 | 58 | |
| | | | | | 2-1-1-1-1 | 2 | 26 | |
| 2 | 4 | 44 | 7 | 100 | 2-1-1 | 18 | 9 | 100 |
| | | | | | 2-2-1 | 29 | 17 | |
| | | | | | 2-3-1 | 45 | 8 | |
| | | | | | 2-1-1-1 | 12 | 13 | |
| | | | | | 2-1-2-1 | 40 | 44 | |
| | | | | | 2-2-1-1 | 40 | 44 | |
| | | | | | 2-2-2-1 | 165 | 8 | |
| | | | | | 2-1-1-1-1 | 4 | 36 | |
| 2 | 5 | 144 | 8 | 100 | 2-1-1 | 98 | 15 | 100 |
| | | | | | 2-2-1 | 95 | 17 | |
| | | | | | 2-3-1 | 173 | 25 | |
| | | | | | 2-1-1-1 | 18 | 34 | |
| | | | | | 2-1-2-1 | 165 | 34 | |
| | | | | | 2-2-1-1 | 165 | 34 | |
| | | | | | 2-2-2-1 | 35 | 14 | |
| | | | | | 2-1-1-1-1 | 16 | 36 | |
| 2 | 6 | 363 | 8 | 100 | 2-1-1 | 485 | 16 | 100 |
| | | | | | 2-2-1 | 237 | 21 | |
| | | | | | 2-3-1 | 398 | 11 | |
| | | | | | 2-1-1-1 | 9 | 56 | |
| | | | | | 2-1-2-1 | 189 | 12 | |
| | | | | | 2-2-1-1 | 189 | 12 | |
| | | | | | 2-2-2-1 | 763 | 11 | |
| | | | | | 2-1-1-1-1 | 9 | 57 | |
| 3 | 3 | 54 | 11 | 100 | 3-1-1 | 30 | 17 | 100 |
| | | | | | 3-2-1 | 29 | 30 | |
| | | | | | 3-3-1 | 31 | 19 | |
| | | | | | 3-1-1-1 | 7 | 28 | |

Anexa 2 – MLMVN-P rezultate complete pentru "Suma modulo k a n valori" 83

| | | | | | | | | |
|---|---|--------|----|--------|-----------|-------|----|-----|
| | | | | | 3-1-2-1 | 17 | 36 | |
| | | | | | 3-2-1-1 | 44 | 17 | |
| | | | | | 3-2-2-1 | 101 | 33 | |
| | | | | | 3-1-1-1-1 | 3 | 12 | |
| 3 | 4 | 165 | 18 | 100 | 3-1-1 | 523 | 24 | 100 |
| | | | | | 3-2-1 | 976 | 46 | |
| | | | | | 3-3-1 | 189 | 22 | |
| | | | | | 3-1-1-1 | 101 | 46 | |
| | | | | | 3-1-2-1 | 520 | 30 | |
| | | | | | 3-2-1-1 | 207 | 28 | |
| | | | | | 3-2-2-1 | 43 | 28 | |
| | | | | | 3-1-1-1-1 | 233 | 29 | |
| 3 | 5 | 3082 | 14 | 100 | 3-1-1 | 3141 | 42 | 100 |
| | | | | | 3-2-1 | 327 | 22 | |
| | | | | | 3-3-1 | 6267 | 28 | |
| | | | | | 3-1-1-1 | 928 | 17 | |
| | | | | | 3-1-2-1 | 258 | 25 | |
| | | | | | 3-2-1-1 | 1960 | 43 | |
| | | | | | 3-2-2-1 | 1484 | 36 | |
| | | | | | 3-1-1-1-1 | 225 | 44 | |
| 3 | 6 | 10014 | 60 | 100 | 3-1-1 | 1412 | 60 | 100 |
| | | | | | 3-2-1 | 807 | 60 | |
| | | | | | 3-3-1 | 10754 | 28 | |
| | | | | | 3-1-1-1 | 1010 | 57 | |
| | | | | | 3-1-2-1 | 2214 | 30 | |
| | | | | | 3-2-1-1 | 10057 | 28 | |
| | | | | | 3-2-2-1 | 2353 | 60 | |
| | | | | | 3-1-1-1-1 | 4010 | 15 | |
| 4 | 3 | 83 | 11 | 100 | 4-1-1 | 37 | 24 | 100 |
| | | | | | 4-2-1 | 247 | 16 | |
| | | | | | 4-3-1 | 320 | 60 | |
| | | | | | 4-1-1-1 | 3 | 59 | |
| | | | | | 4-1-2-1 | 89 | 59 | |
| | | | | | 4-2-1-1 | 45 | 35 | |
| | | | | | 4-2-2-1 | 147 | 30 | |
| | | | | | 4-1-1-1-1 | 3 | 38 | |
| 4 | 4 | 3317 | 23 | 91.406 | 4-1-1 | 2520 | 23 | 100 |
| | | | | | 4-2-1 | 879 | 47 | |
| | | | | | 4-3-1 | 2078 | 13 | |
| | | | | | 4-1-1-1 | 883 | 29 | |
| | | | | | 4-1-2-1 | 410 | 60 | |
| | | | | | 4-2-1-1 | 316 | 22 | |
| | | | | | 4-2-2-1 | 695 | 44 | |
| | | | | | 4-1-1-1-1 | 449 | 25 | |
| 4 | 5 | 127966 | 24 | 65.440 | 4-1-1 | 10549 | 37 | 100 |
| | | | | | 4-2-1 | 10443 | 35 | |
| | | | | | 4-3-1 | 15859 | 24 | |
| | | | | | 4-1-1-1 | 11142 | 33 | |

84 Anexa 2 – MLMVN-P rezultate complete pentru "Suma modulo k a n valori"

| | | | | | | | | | | | | | |
|---|---|--------|----|--------|-----------|-------|----|-----|--|-----------|-------|----|--------|
| | | | | | 4-1-2-1 | 5737 | 32 | | | | | | |
| | | | | | 4-2-1-1 | 721 | 35 | | | | | | |
| | | | | | 4-2-2-1 | 15859 | 24 | | | | | | |
| | | | | | 4-1-1-1-1 | 454 | 37 | | | | | | |
| 4 | 6 | 19857 | 19 | 50.694 | 4-1-1 | 25265 | 60 | 100 | | | | | |
| | | | | | 4-2-1 | 5264 | 60 | | | | | | |
| | | | | | 4-3-1 | 5951 | 58 | | | | | | |
| | | | | | 4-1-1-1 | 17249 | 59 | | | | | | |
| | | | | | | | | | | 4-1-2-1 | 4416 | 59 | |
| | | | | | | | | | | 4-2-1-1 | 23803 | 59 | 82.176 |
| | | | | | | | | | | 4-2-2-1 | 2230 | 59 | 100 |
| | | | | | | | | | | 4-1-1-1-1 | 6195 | 61 | 93.827 |
| 5 | 3 | 114392 | 20 | 100 | 5-1-1 | 428 | 23 | 100 | | | | | |
| | | | | | 5-2-1 | 335 | 15 | | | | | | |
| | | | | | 5-3-1 | 1023 | 19 | | | | | | |
| | | | | | 5-1-1-1 | 11 | 57 | | | | | | |
| | | | | | 5-1-2-1 | 371 | 36 | | | | | | |
| | | | | | 5-2-1-1 | 1022 | 37 | | | | | | |
| | | | | | 5-2-2-1 | 349 | 16 | | | | | | |
| | | | | | 5-1-1-1-1 | 430 | 30 | | | | | | |
| 5 | 4 | 140021 | 45 | 68.652 | 5-1-1 | 976 | 29 | 100 | | | | | |
| | | | | | 5-2-1 | 2732 | 29 | | | | | | |
| | | | | | 5-3-1 | 2655 | 45 | | | | | | |
| | | | | | 5-1-1-1 | 1032 | 61 | | | | | | |
| | | | | | 5-1-2-1 | 1386 | 50 | | | | | | |
| | | | | | 5-2-1-1 | 8582 | 43 | | | | | | |
| | | | | | 5-2-2-1 | 1166 | 59 | | | | | | |
| | | | | | 5-1-1-1-1 | 771 | 43 | | | | | | |
| 6 | 3 | 119119 | 20 | 92.181 | 6-1-1 | 231 | 22 | 100 | | | | | |
| | | | | | 6-2-1 | 1169 | 40 | | | | | | |
| | | | | | 6-3-1 | 2148 | 25 | | | | | | |
| | | | | | 6-1-1-1 | 360 | 60 | | | | | | |
| | | | | | 6-1-2-1 | 194 | 39 | | | | | | |
| | | | | | 6-2-1-1 | 371 | 60 | | | | | | |
| | | | | | 6-2-2-1 | 124 | 35 | | | | | | |
| | | | | | 6-1-1-1-1 | 2 | 40 | | | | | | |

Anexa 3 – Baza de date IRIS

Baza de date IRIS reprezintă cel mai cunoscut set de date folosit în domeniul RN pentru clasificare. Această bază de date conține 150 de instanțe, fiind compus din 3 clase a câte 50 de instanțe fiecare [30].

Fiecare instanță are 4 valori de intrare (numere reale) reprezentând în ordine lungimea sepalei în cm, lățimea sepalei în cm, lungimea petalei în cm, respectiv lățimea petalei în cm și o valoare de ieșire (număr întreg) reprezentând 3 clase: Iris Setosa (0), Iris Versicolour (1) și Iris Virginica (2) [30].

Tabelul 20 prezintă datele statistice ale bazei de date IRIS.

Tabelul 20: Datele statistice sumarizate ale bazei de date IRIS

| | Min | Max | Mean | SD | Corelația claselor |
|------------------|-----|-----|------|------|--------------------|
| lungimea sepalei | 4.3 | 7.9 | 5.84 | 0.83 | 0.7826 |
| lățimea sepalei | 2.0 | 4.4 | 3.05 | 0.43 | -0.4194 |
| lungimea petalei | 1.0 | 6.9 | 3.76 | 1.76 | 0.9490 |
| lățimea petalei | 0.1 | 2.5 | 1.20 | 0.76 | 0.9565 |

Parametrii prezentați în Tabelul 20 au următoarele semnificații:

- **Min:** valoare minimă a mărimii considerate;
- **Max:** valoare maximă a mărimii considerate;
- **Mean:** valoare medie a mărimii considerate;
- **SD:** deviația standard a mărimii considerate;
- **Corelația claselor:** corelația dintre mărimile considerate și clasele de flori.

Anexa 4 – Baza de date „Breast Cancer Wisconsin (Diagnostic)”

Baza de date „Breast Cancer Wisconsin (Diagnostic)” conține 699 de instanțe, fiind compusă din 2 clase a câte 458 de instanțe (65.5%), respectiv 241 (34.5%) [30].

Fiecare instanță are 10 valori de intrare (numere întregi) și o valoare de ieșire (număr întreg) reprezentând 2 clase: benign și malign [30].

Tabelul 21: Atributele bazei de date „Breast Cancer Wisconsin (Diagnostic)” [ENG]

| # | Attribute | Domain |
|----|-----------------------------|-------------------------------|
| 1 | Sample code number | id number |
| 2 | Clump Thickness | 1 - 10 |
| 3 | Uniformity of Cell Size | 1 - 10 |
| 4 | Uniformity of Cell Shape | 1 - 10 |
| 5 | Marginal Adhesion | 1 - 10 |
| 6 | Single Epithelial Cell Size | 1 - 10 |
| 7 | Bare Nuclei | 1 - 10 |
| 8 | Bland Chromatin | 1 - 10 |
| 9 | Normal Nucleoli | 1 - 10 |
| 10 | Mitoses | 1 - 10 |
| 11 | Class | 2 for benign, 4 for malignant |

Există 16 instanțe care au un atribut lipsă; acestea au fost eliminate în întregime în timpul învățării. De asemenea, atributul #1 nu a fost considerat ca intrare validă, iar clasele au fost codate astfel: 0 pentru benign, respectiv 1 pentru malign.

Anexa 5 – MVN-P simulator

Aplicația folosită pentru simularea unui MVN-P (Figura 39) a fost realizată în Microsoft Visual C# 2010. Pentru folosirea unui singur MVN-P se lasă nebifată opțiunea „Enable neural network”, se introduce în câmpul „Input Layer/Nb. of neurons” numărul de intrări ale problemei și se apasă butonul „Build NN”.

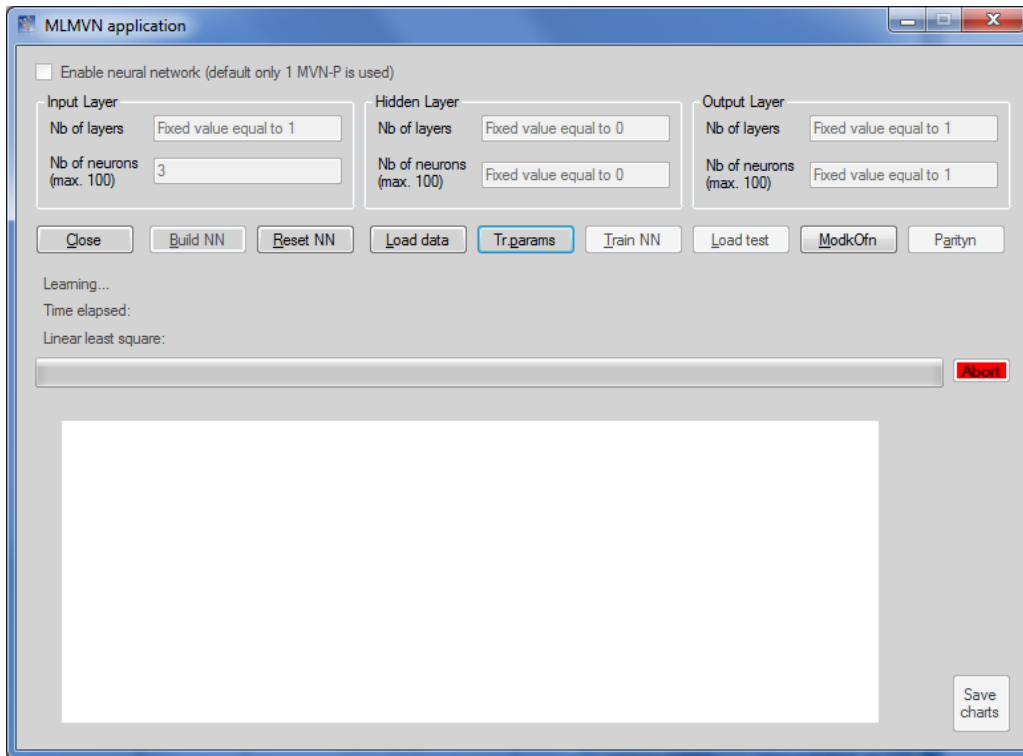


Figura 39: MVN-P simulator

Setul de date se încarcă folosind butonul „Load data”; fișierul cu setul de date trebuie să aibă extensia .txt și denumirea addnmodk, unde **n** este numărul de intrări, iar **k** numărul de ieșiri și trebuie să aibă următoarea structură:

| | | | | | | | | |
|--------|----------|----------|----------|----------|----------|----------|----------|----------|
| Input | X_{11} | X_{12} | X_{13} | X_{14} | X_{15} | X_{16} | X_{17} | X_{18} |
| Input | X_{21} | X_{22} | X_{23} | X_{24} | X_{25} | X_{26} | X_{27} | X_{28} |
| Input | X_{31} | X_{32} | X_{33} | X_{34} | X_{35} | X_{36} | X_{37} | X_{38} |
| Output | Y_1 | Y_2 | Y_3 | Y_4 | Y_5 | Y_6 | Y_7 | Y_8 |

, unde X_{11} reprezintă intrarea 1 a primei instanțe, X_{21} intrarea 2 a primei instanțe, X_{31} intrarea 3 a primei instanțe, Y_1 ieșirea (sau clasa) primei instanțe, ș.a.m.d.

În Figura 40 sunt prezentați parametrii folosiți în procesul de învățare.

Figura 40: Parametrii folosiți în timpul învățării MVN-P

În continuare sunt prezentați, pe scurt, fiecare parametru în parte:

- **„Use extended complex numbers”**: utilizat în vederea studierii aplicabilității cuaternionilor în cazul MVN-P (momentan această opțiune este dezactivată);
- **„Run automatic tests”**: utilizat în vederea rulării de teste automate pentru validarea diverselor configurații; odată activată această opțiune se poate introduce numărul de teste ce urmează a fi rulate;
- **„Generate random data structure”**: utilizat pentru generarea aleatoare a ordinii în care instanțele sunt prezentate în timpul învățării;
- **„Init weights randomly”**: utilizat pentru generarea aleatoare a ponderilor inițiale;
- **„Abort if efficiency is >”**: utilizat în cazul în care se dorește întreruperea învățării în momentul atingerii unei anumite eficiențe;
- **„Learning constant”**: reprezintă constanta de învățare (poate fi considerată egală cu $1+0i$ în majoritatea cazurilor);
- **„Nb of training epochs”**: reprezintă valoarea maximă a numărului de iterații permise;

- **„K-valued logic”**: reprezintă numărul de clase de ieșire;
- **„L-periodic value”**: reprezintă periodicitatea soluțiilor (sectoarelor) în timpul învățării;
- **„Corection rule”**: reprezintă regula de corecție folosită; poate fi una dintre regulile de corecție descrise de ecuațiile (1.4), (1.5), (1.6) sau (1.7);
- **„Neighbor step”**: reprezintă pasul de căutare al sectorului corect;
- **„Learning strategy”**: reprezintă strategia aleasă de căutare a sectorului corect; poate fi una din strategiile propuse de Aizenberg sau noua strategie de căutare propusă. **Notă:** strategiile propuse de Aizenberg au fost implementate după pseudo-codul prezentat în [38] și pot produce rezultate diferite; din acest motiv, în timpul analizei comparative, din Capitolul 5.1, rezultatele folosite sunt cele prezentate în articolele studiate și nu cele obținute folosind acest simulator;
- **„Nb. of learning sets”**: reprezintă numărul de instanțe folosite din întreg setul de date;
- **„Distance criteria”**: utilizat în vederea folosirii diverselor formule de calcul al distanței dintre 2 sectoare (momentan această opțiune este limitată la diferența dintre coordonatele reale, respectiv imaginare ale numerelor complexe $\varepsilon^q, \varepsilon^s$ descrise de ecuația (3.5));
- **„Counter-clockwise logic is enabled”**: utilizat pentru setarea direcției de căutare (orară sau anti-orară) pentru noua strategie propusă;

Anexa 6 – MLMVN-P simulator

Aplicația folosită pentru simularea unui MVN-P a fost modificată pentru simularea MLMVN-P prin bifarea opțiunii „Enable neural network” (Figura 41).

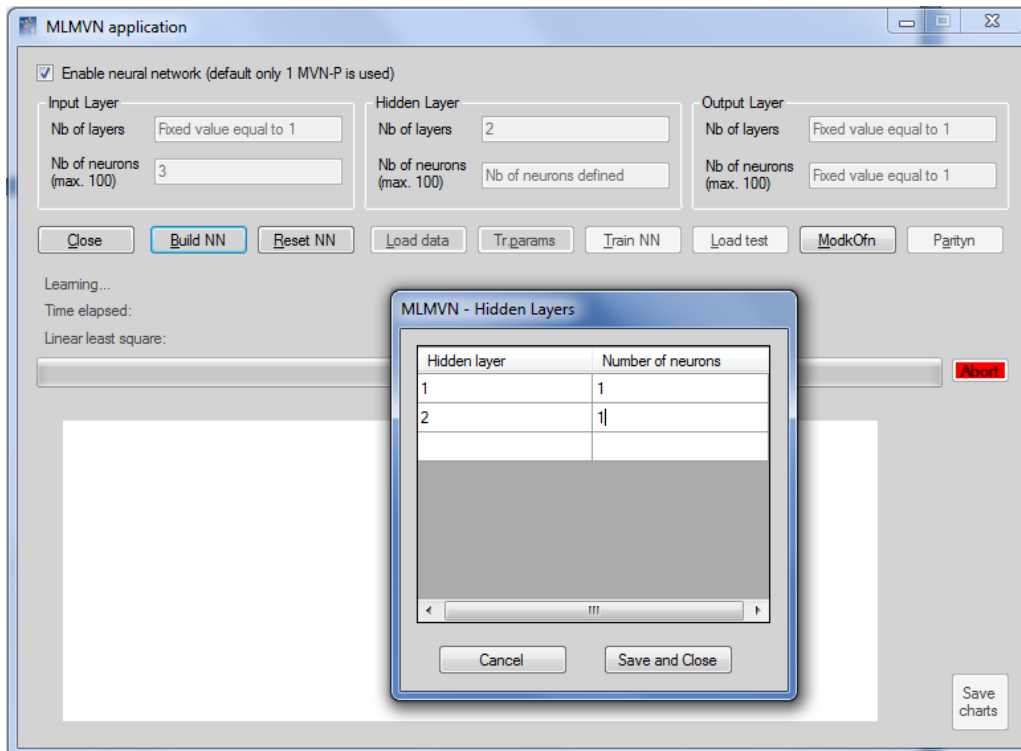


Figura 41: MLMVN-P simulator

Numărul de straturi ascunse se definește în câmpul „Hidden Layer/Nb of layers”, iar numărul de neuroni de pe fiecare strat ascuns este definit în fereastra „MLMVN – Hidden layer”; această fereastră apare automat la efectuarea unui click al mouse-ului în câmpul „Hidden Layer/Nb of neurons”.

Parametrii folosiți în procesul de învățare sunt aceiași cu cei prezentați în Figura 40.

Anexa 7 – Lista cu lucrările științifice comunicate/publicate

- [1] V.M. Lupea, „Multi-Valued Neuron with a periodic activation function (MVN-P) – New learning strategy”, Workshop-ul nr. 2 „Interdisciplinaritatea Și Managementul Cercetării În Studiile Doctorale”, Universitatea din Oradea, 7-8 Iunie 2012.
- [2] V.M. Lupea, "Multi-Valued Neuron with a periodic activation function — New learning strategy", Proc. of 8th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP 2012), Cluj-Napoca, Romania, 30 August -1 Septembrie 2012, pp. 79-82. (IEEE Xplore, Scopus, INSPEC).
- [3] V.M. Lupea, "Multi-Valued Neuron with a periodic activation function - as part of a multi-layered Neural Network", Proc. of 11th IEEE International Symposium on Applied Machine Intelligence and Informatics (SAMI 2013), Herl'any, Slovakia, 31 Jan. - 2 Feb. 2013, pp. 121-124. (IEEE Xplore, ISI Web of Knowledge, Scopus, INSPEC).
- [4] V.M. Lupea, „Multi-Valued Neuron with a periodic activation function – as part of a multi-layered Neural Network”, Workshop-ul nr. 3 Interdisciplinaritatea Și Managementul Cercetării În Studiile Doctorale”, Universitatea din Pitesti, 30-31 Mai 2013.
- [5] Lupea, Valentin Mircea, “Multi-Valued Neuron with a periodic activation function – influence of l-periodic parameter over the learning process” , Proc. of 14th IEEE International Symposium on Computational Intelligence and Informatics (CINTI 2013), vol., no., pp.185,188, Budapesta, Ungaria, 19-21 Nov. 2013 (IEEE Xplore).

Bibliografie

- [1] Alexander Bain, "Mind and Body: The Theories of Their Relation", New York: D. Appleton and Company, 1873.
- [2] William James, "The Principles of Psychology", New York: H. Holt and Company, 1890.
- [3] Hebb, D.O., "The Organization of Behavior", New York: Wiley & Sons, 1949.
- [4] F.Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain", *Psychological Review* 65, pp 386-408, 1958.
- [5] Minsky, M.L., Papert, S.: *Perceptrons: "An introduction to Computational Geometry"*, The MIT Press, Cambridge, MA, USA, 1969.
- [6] Kohonen, T., "Correlation Matrix Memories," *Computers, IEEE Transactions on* , vol.C-21, no.4, pp.353,359, April 1972 doi: 10.1109/TC.1972.5008975.
- [7] James A. Anderson, A simple neural network generating an interactive memory, *Mathematical Biosciences*, Volume 14, Issues 3-4, August 1972, Pages 197-220, ISSN 0025-5564.
- [8] Kohonen, Teuvo, "Self-Organized Formation of Topologically Correct Feature Maps". *Biological Cybernetics* 43 (1): 59-69, 1982.
- [9] J. J. HOPFIELD, "Neural networks and physical systems with emergent collective computational abilities", *Biophysics Proc. NatL Acad. Sci. USA* Vol. 79, pp. 2554-2558, 1982.
- [10] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, Learning representations by back-propagating errors. *Nature* 323:533-536, 1986.
- [11] B. Widrow. Adaline and madaline - 1963: Plenary speech. volume 1, pp 143-158, 1987.
- [12] Aizenberg, N.N.; Aizenberg, I.N., "CNN based on multi-valued neuron as a model of associative memory for grey scale images," *Cellular Neural Networks and their Applications*, 1992. CNNA-92 Proceedings., Second International Workshop on , vol., no., pp.36,41, 14-16 Oct 1992 doi: 10.1109/CNNA.1992.274330.
- [13] N.N.Aizenberg, I.N.Aizenberg., G.A.Krivosheev "Multi-Valued Neurons: Learning, Networks, Application to Image Recognition and Extrapolation of Temporal Series", *Lecture Notes in Computer Science*, Vol. 930, (JMra, F.Sandova1 - Eds.), Springer-Verlag, 1995, pp.389-395.
- [14] N.N.Aizenberg, I.N.Aizenberg, G.A.Krivosheek "Multi-Valued Neurons: Mathematical model, Networks, Application to Pattern Recognition", *Proc. of the 13 Int.Coi2f: on Pattern Recognition*, Vienna, August 25-30, 1996, Track D, IEEE Computer Soc. Press, pp. 185-189, 1996.
- [15] S. Jankowski, A.Loowski, M.Zurada "Complex-Valued Multistate Neural Associative Memory", *IEEE Trans. on Neural Networks*, Vol. 7, pp.1491-1496, 1996.
- [16] Jankowski, S.; Loowski, Andrzej; Zurada, J.M., "Layer interactions in a multivalued neural associative memory," *Circuits and Systems*, 1996. ISCAS '96., Connecting the World., 1996 IEEE International Symposium on , vol.3, no., pp.389,392 vol.3, 12-15 May 1996 doi: 10.1109/ISCAS.1996.541615.
- [17] Aizenberg, N.N.; Aizenberg, I.N.; Krivosheev, G.A., "Multi-valued and universal binary neurons: mathematical model, learning, networks, application

- to image processing and pattern recognition," *Pattern Recognition*, 1996., Proceedings of the 13th International Conference on , vol.4, no., pp.185,189 vol.4, 25-29 Aug 1996 doi: 10.1109/ICPR.1996.547258.
- [18] Donq-Liang Leea, Wen-June Wangb, "A multivalued bidirectional associative memory operating on a complex domain", *Neural Networks* 11 (1998) 1623–1635, PII: S0893-6080(98)00078-1.
- [19] I. Aizenberg, N. Aizenberg and J.Vandewalle "Multi-valued and universal binary neurons: theory, learning, applications", Kluwer Academic Publishers, Boston/Dordrecht/London (2000).
- [20] Aizenberg, I.; Aizenberg, N.; Butakov, C.; Farberov, E., "Image recognition on the neural network based on multi-valued neurons," *Pattern Recognition*, 2000. Proceedings. 15th International Conference on , vol.2, no., pp.989,992 vol.2, 2000 doi: 10.1109/ICPR.2000.906241.
- [21] John Reinitz, Igor Aizenberg, Ekaterina Myasnikova, Maria Samsonova, "Application of the Neural Networks Based on Multi-valued Neurons to Classification of the Images of Gene Expression Patterns", *Computational Intelligence. Theory and Applications Lecture Notes in Computer Science Volume 2206*, 2001, pp 291-304, DOI 10.1007/3-540-45493-4_33.
- [22] Aoki, H., "A complex-valued neuron to transform gray level images to phase information," *Neural Information Processing*, 2002. ICONIP '02. Proceedings of the 9th International Conference on , vol.3, no., pp.1084,1088 vol.3, 18-22 Nov. 2002 doi: 10.1109/ICONIP.2002.1202789.
- [23] Igor Aizenberg, Claudio Moraga, and Dmitriy Paliy, "A Feedforward Neural Network based on Multi-Valued Neurons", *Computational Intelligence, Theory and Applications. Advances in Soft Computing, XIV*, Springer, 2005, Aizenberg I. et. al.
- [24] Emad A. M. Andrews Shenouda "A Quantitative Comparison of Different MLP Activation Functions in Classification", J. Wang et al. (Eds.): *ISNN 2006, LNCS 3971*, pp. 849 – 857, 2006. © Springer-Verlag Berlin Heidelberg 2006.
- [25] Aizenberg Igor, Paliy Dmitriy, Moraga. Claudio and Astola Jaakko, „Blur Identification Using Neural Network for Image Restoration”, *Computational Intelligence, Theory and Applications*, vol. 38, pp. 441-455, 2006 doi: 10.1007/3-540-34783-6_45, Springer Berlin Heidelberg.
- [26] Igor Aizenberg, Pekka Ruusuvuori, Olli Yli-Harja and Jaakko T. Astola, "MULTILAYER NEURAL NETWORK BASED ON MULTI-VALUED NEURONS (MLMVN) APPLIED TO CLASSIFICATION OF MICRORARRAY GENE EXPRESSION DATA", Published in the Proceedings of the 4th TICSP Workshop on Computational Systems Biology (WCSB 2006), Tampere University of Technology, Finland, June 12-13, 2006, pp. 27-30.
- [27] Aizenberg, I.; Paliy, D.; Astola, J.T., "Multilayer Neural Network based on Multi-Valued Neurons and the Blur Identification Problem," *Neural Networks*, 2006. IJCNN '06. International Joint Conference on , vol., no., pp.473,480, 0-0 0, doi: 10.1109/IJCNN.2006.246719.
- [28] Igor Aizenberg, "Solving the Parity n Problem and Other Nonlinearly Separable Problems Using a Single Universal Binary Neuron", *Computational Intelligence, Theory and Applications*, vol. 38pp 457-471, DOI 10.1007/3-540-34783-6_46, 2006.
- [29] Igor Aizenberg, Claudio Moraga, "Multilayer Feedforward Neural Network Based on Multi-valued Neurons (MLMVN) and a Backpropagation Learning

- Algorithm", *Soft Computing*, Volume 11, Issue 2 , pp 169-183, 2007, DOI 10.1007/s00500-006-0075-5.
- [30] A. Asuncion and D. J. Newman. (2007). UCI Machine Learning Repository. School Inform. Comput. Sci., Univ. California, Irvine [Online]. Available: <http://www.ics.uci.edu/mlearn/MLRepository.html>.
- [31] Igor Aizenberg, Jacek M. Zurada, "Solving Selected Classification Problems in Bioinformatics Using Multilayer Neural Network Based on Multi-Valued Neurons (MLMVN)", *Artificial Neural Networks – ICANN 2007, 17th International Conference*, Porto, Portugal, September 9-13, 2007, Proceedings, Part I, pp 874-883, 2007, DOI 10.1007/978-3-540-74690-4_89.
- [32] Aizenberg, I.; Moraga, Claudio, "The Genetic Code as a Multiple-Valued Function and Its Implementation Using Multilayer Neural Network Based on Multi-Valued Neurons," *Multiple-Valued Logic*, 2007. ISMVL 2007. 37th International Symposium on , vol., no., pp.13,13, 13-16 May 2007, doi: 10.1109/ISMVL.2007.54.
- [33] Aizenberg, I.; Paliy, D.V.; Zurada, J.M.; Astola, J.T., "Blur Identification by Multilayer Neural Network Based on Multivalued Neurons," *Neural Networks, IEEE Transactions on* , vol.19, no.5, pp.883,898, May 2008, doi: 10.1109/TNN.2007.914158.
- [34] Zurada, J.M.; Aizenberg, I.; Mazurowski, M.A., "Learning in networks: Complex-valued neurons, pruning, and rule extraction," *Intelligent Systems, 2008. IS '08. 4th International IEEE Conference* , vol.1, no., pp.1-15,1-20, 6-8 Sept. 2008, doi: 10.1109/IS.2008.4670394.
- [35] Igor Aizenberg, "Solving the XOR and parity N problems using a single universal binary neuron", *Soft Computing* February 2008, Volume 12, Issue 3, pp 215-222, DOI 10.1007/s00500-007-0204-9.
- [36] Wei Zhou; Zurada, J.M., "A class of discrete-time recurrent neural networks with multivalued neurons," *Neural Networks, 2009. IJCNN 2009. International Joint Conference on* , vol., no., pp.3555,3558, 14-19 June 2009, doi: 10.1109/IJCNN.2009.5178721.
- [37] Bekir Karlik and A. Vehbi Olgac, "Performance Analysis of Various Activation Functions in Generalized MLP Architectures of Neural Networks", *International Journal of Artificial Intelligence And Expert Systems (IJAE)*, Volume 1, Issue 4, pp 111-122, 2010.
- [38] Aizenberg, I., "Periodic Activation Function and a Modified Learning Algorithm for the Multivalued Neuron," *Neural Networks, IEEE Transactions on* , vol.21, no.12, pp.1939,1949, Dec. 2010, doi: 10.1109/TNN.2010.2082561.
- [39] Aizenberg, I., "Learning of the Non-threshold Functions of Multiple-Valued Logic by a Single Multi-valued Neuron with a Periodic Activation Function," *Multiple-Valued Logic (ISMVL)*, 2010 40th IEEE International Symposium on , vol., no., pp.33,38, 26-28 May 2010, doi: 10.1109/ISMVL.2010.15.
- [40] Aizenberg, I.; Caudill, M.; Jackson, J.; Alexander, S., "Learning nonlinearly separable mod k addition problem using a single multi-valued neuron with a periodic activation function," *Neural Networks (IJCNN), The 2010 International Joint Conference on* , vol., no., pp.1,8, 18-23 July 2010, doi: 10.1109/IJCNN.2010.5596509.
- [41] Igor Aizenberg, "Multi-Valued Neuron with a Periodic Activation Function", *Complex-Valued Neural Networks with Multi-Valued Neurons Studies in*

-
- Computational Intelligence Volume 353, 2011, pp 173-206, DOI 10.1007/978-3-642-20353-4_5.
- [42] Aizenberg, I.; Jackson, J.; Alexander, S., "Classification of blurred textures using multilayer neural network based on multi-valued neurons," Neural Networks (IJCNN), The 2011 International Joint Conference on , vol., no., pp.1328,1335, July 31 2011-Aug. 5 2011, doi: 10.1109/IJCNN.2011.6033378.
- [43] Igor Aizenberg, Stefano Manetti, Antonio Luchetta, "A modified learning algorithm for the multilayer neural network with multi-valued neurons based on the complex QR decomposition", Soft Computing, Volume 16, Issue 4 , pp 563-575, 2012, DOI 10.1007/s00500-011-0755-7.
- [44] Igor Aizenberg, "Hebbian and error-correction learning for complex-valued neurons", Soft Computing Volume 17, Issue 2 , pp 265-273, 2013, DOI 10.1007/s00500-012-0891-8.
- [45] Cernazanu, C., Holban, S.: „Improving neural network performances – training with negative examples”, International Conference on Telecommunications and Networking/International Conference on Industrial Electronics, Technology and Automation, University of Bridgeport, Novel Algorithms and Techniques in Telecommunications, Automation and Industrial Electronics, pag. 49-53, 2008.
- [46] Cernazanu, C., Holban, S.: „Determining the optimal percent of negative examples used in training the multilayer perceptron neural networks”, International Conference on Neural Networks, pag. 114-119, Prague, 2009.
- [47] Cernazanu-Glavan Cosmin, "Contributii la antrenarea retelelor neuronale. Învatarea pe baza corectiei erorii cu exemple negative", teza de doctorat, iunie 2009.
- [48] Lupea, V.M., "Multi-Valued Neuron with a periodic activation function – New learning strategy," Intelligent Computer Communication and Processing (ICCP), 2012 IEEE International Conference on , vol., no., pp.79,82, Aug. 30 2012-Sept. 1 2012 doi: 10.1109/ICCP.2012.6356164.
- [49] Lupea, V.M., "Multi-Valued Neuron with a periodic activation function - as part of a multi-layered Neural Network," Applied Machine Intelligence and Informatics (SAMI), 2013 IEEE 11th International Symposium on , vol., no., pp.121,124, Jan. 31 2013-Feb. 2 2013 doi: 10.1109/SAMI.2013.6480958.
- [50] Lupea, Valentin Mircea, "Multi-valued neuron with a periodic activation function – Influence of l-periodic parameter over the learning process," Computational Intelligence and Informatics (CINTI), 2013 IEEE 14th International Symposium on , vol., no., pp.185,188, 19-21 Nov. 2013 doi: 10.1109/CINTI.2013.6705189.