
Unitatea de învățare 1.

ELEMENTE DE INTELIGENȚĂ COMPUTAȚIONALĂ ÎN CONTROLUL PROCESELOR

Cuprins

1.1. Introducere. Soft computing.....	
1.2. Logica fuzzy.....	
1.3. Rețele neuronale.....	
1.4. Sisteme neuro-fuzzy.....	
1.5. Aplicații ale tehnicilor de inteligență computațională în automatică.....	



Introducere

În inginerie, experiența este definită atât prin reguli stricte, cât și sub forma unor aprecieri și evaluări calitative, mergând chiar până la ceea ce numim intuiție. Acestea sunt rezultatul mecanismelor naturale de învățare și adaptare ce caracterizează în fapt inteligența umană, la baza cărora stau modele, exemple, experimente, sau pe scurt experiențe de învățare.

Nivelul experienței, descris prin cantitatea și calitatea cunoștințelor acumulate, este direct determinat de numărul cât mai mare de experiențe de învățare utile. Simplu spus „repetarea este mama învățării”. Evident, experiența nu este înregistrată într-o formă rigidă, ci este se modifică cu fiecare nou caz studiat. Utilitatea unei experiențe de învățare nu poate fi evaluată în momentul învățării, dar devine certă pe măsură ce experiența crește. În aceste condiții, rolul unui supervisor experimentat în procesul de învățare devine important: acesta reduce timpul de învățare și ghidează către rezultatele corecte.

Din punct de vedere biologic, aceste mecanisme de acumulare a experienței, studiate pe larg în psihologie, sunt similare celor de formare a reflexelor naturale și a funcțiunilor sistemului nervos. Astfel, este rezonabilă încercarea de a copia mecanismele de învățare umane în cadrul unor modele matematice și a unor sisteme tehnice sau informatice. Această idee se concretizează în domeniul modern al rețelelor neuronale.

Pe de altă parte, experiență umană este în mod frecvent, dacă nu chiar întotdeauna, exprimată într-o manieră aproximativă, lingvistică. Adică, evaluările mărimilor sunt formulate prin atribute și descrieri calitative, iar relaționările între diverse mărimi și acțiuni sunt exprimate în forma unor reguli. Pentru ambele, precizia este

ignorată, cel puțin într-o primă etapă. Evident, și această caracteristică a raționamentului uman inspiră modele matematice și metode, acestea definind domeniul logicii fuzzy.



Competențele unității de învățare

- *Operarea cu concepte fundamentale de inteligență artificială în contextul aplicațiilor acestora în conducerea automată.*



Durata medie de parcurgere a primei unități de învățare este de 2 ore.

1.1.Introducere. Soft computing.

Proiectarea și implementarea sistemelor cu logică fuzzy în inginerie, în general, și în ingineria electrică și automatică, în particular, au fost tratate intens, din perspective teoretice și practice, de la definirea logicii fuzzy ca subiect matematic ([ZAD1965]) și de la introducerea ei în tehnică ([MAM1974]). Studiile teoretice au condus la expunerea multor strategii concrete de stabilire a bazelor de reguli fuzzy, de la cele empirice, simple, până la metodologii complexe bazate pe algoritmi specifici inteligenței artificiale, cum ar fi auto-instruirea, rețelele neuronale, algoritmi genetici ([JOV1999], [JAN2007], [PAY1998], [PRP1997]). Dintre acestea, ultimele amintite sunt preferate la ora actuală, datorită adaptabilității ridicate ale acestora la diverse situații, fără a implica etape de modelare matematică dificilă.

Referirea la domeniul inteligenței artificiale este rezonabilă în contextul în care noile strategii emulează două caracteristici umane, inteligente - *învățarea* și *adaptarea* – fie în timpul funcționării efective a sistemului, fie în etapa de proiectare a sistemului. Cele două concepte sunt implementate practic de algoritmii de antrenare a rețelelor neuronale, în diverse adaptări pentru variantele de structură sau model neuronal.

În contextul sistemelor cu logică fuzzy, învățarea reprezintă capacitatea de a extrage un set de reguli, fuzzy sau non-fuzzy, dintr-un proces de înregistrare și prelucrare de date numerice (pentru un sistem automat) sau de observații experimentale (pentru un proiectant). Un exemplu sugestiv este mecanismul de construcție a bazei de reguli pentru un regulator fuzzy descris în [NIL1995], bazat pe conceptul de auto-instruire din [ARI1990] și simplificat la condiții practice concrete în [BCC2009]. Adaptarea semnifică ajustarea unei baze de reguli standard sau brute la condițiile aplicației practice, cu scopul optimizării sistemului proiectat.

Această ajustare se poate face în baza unui algoritm de minimizare a erorii, cum ar fi antrenarea rețelelor neuronale, ca în [BCT2010b].

Fiecare tehnică inteligentă are proprietăți tipice care îi conferă aplicabilitatea pentru anumite probleme și prezintă dezavantaje în rezolvarea altora. De exemplu, rețelele neuronale au avantajul recunoașterii modelelor, însă nu pot explica modul în care au ajuns la o concluzie. Pe de altă parte, sistemele fuzzy pot raționa cu informații imprecise și își pot explica raționamentul, dar nu pot achiziționa automat regulile pe care le utilizează pentru luarea deciziilor. Aceste limitări au fost principalul motiv pentru care s-a încercat crearea de sisteme inteligente hibride, în care două sau mai multe tehnici pot fi combinate pentru a depăși dezavantajele tehnicilor individuale.

Multitudinea de structuri hibride neuro-fuzzy este posibilă datorită faptului că cele două domenii ale matematicii – rețele neuronale și logica fuzzy – pot fi combinate la diverse niveluri de complexitate, în funcție de modul de combinare a modului simbolic fuzzy cu modulul conexionist sub-simbolic. Criteriul de clasificare al tipurilor este cel al rolului sau „poziției” celor două părți ([ABR2001]). Primul model neuro-fuzzy este introdus este cel propus de Jang ([JAN1992a], [JAN1992b]), cunoscut sub acronimul ANFIS.

Includerea unor caracteristici de inteligență artificială în etapa de proiectare are un avantaj semnificativ: foarte puține informații privind procesul reglat sunt necesare în proiectare. Necesitatea unui operator uman experimentat sau a unui model matematic este redusă sau chiar eliminată complet.

Soft computing

Termenul *soft computing*, introdus de L. Zadeh în 1965, desemnează un set de metode care pot „exploata toleranța la imprecizie, incertitudine și adevăr parțial pentru a dobândi flexibilitate, robustețe, costuri scăzute ale soluțiilor și o legătură mai bună cu realitatea”. Această descriere a metodelor le face fundamental diferite de soluțiile de implementare hardware clasice. Ținta finală a conceptului este simularea tehnicilor de proiectare și a raționamentelor minții umane, de unde și abordarea lor ca *tehnici de inteligență artificială*.

Soft computing-ul se caracterizează printr-un parteneriat al mai multor sub-domenii, dintre care cele mai importante sunt rețelele neuronale, algoritmi genetici, logica fuzzy și raționamentele probabilistice.

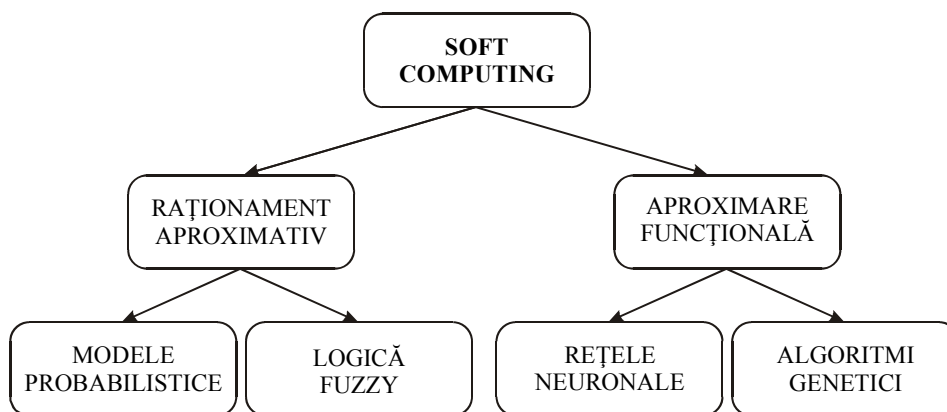


Figura 1.1. Descrierea termenului soft-computing introdus de L. Zadeh.

Având la bază modelul gândirii umane, soft computing-ul grupează aceste domenii, aflate nu într-o relație de concurență, ci una de complementaritate, în care fiecare partener contribuie cu avantajele și tehnicile proprii la soluționarea unor probleme imposibil de rezolvat în alt mod. Astfel, rețelele neuronale au capacitatea de a învăța și de a se adapta, iar logica fuzzy oferă posibilitatea aproximării, în timp ce algoritmi genetici pot realiza o căutare sistematizată a soluției optime. Situat între sistemele de inteligență artificială și computing-ul convențional, soft computing-ul reprezintă problema de rezolvat de o așa manieră încât starea curentă a sistemului poate fi măsurată și comparată cu starea ce se dorește a fi obținută. Starea sistemului stă la baza adaptării parametrilor, care încetul cu încetul converg către soluția optimă.

Logica fuzzy a devenit o parte importantă a noțiunii de soft-computing, ca direcție de abordare ce exploatează „toleranța la imprecizie, incertitudine și adevăr parțial”.

1.2. Logica fuzzy

Abordarea inginerescă „clasică”, strictă, a realității este una în special cantitativă, bazată pe modelări matematice exprimate în forme care inspiră exactitatea. Într-o astfel de abordare, aprecierea de ordin calitativ a mărimilor, valorilor, rezultatelor etc. este greu interpretabilă, deoarece acestea au valori stricte, bine precizate. Într-un context strict, modelele disponibile sunt exacte (de fapt cât mai exacte posibil, cu aproximări și supoziții), iar sistemele de comandă și control sunt dezvoltate în strânsă legătură cu acestea.

Aplicațiile practice în domeniul ingineriei electrice, bazate pe teoria mulțimilor fuzzy, între care și controlul fuzzy, includ un mecanism de evaluare numerică a aprecierilor calitative, proprii exprimării uzuale ale experienței. Abordarea problemelor se extinde, incluzând și aprecieri calitative ale rezultatelor.

În inginerie, experiența descrisă calitativ este rezultatul unor mecanisme naturale de învățare și adaptare, dintr-un set cât mai mare de exemple, experimente, aplicații etc. Experiența nu este înregistrată într-o formă rigidă, ci este se modifică cu fiecare nou caz studiat. Acest mecanism uzual este reprezentat matematic prin introducerea rețelelor neuronale artificiale, cu modele și algoritmi specifici.

Deși aparent independente, cele două caracteristici, aprecierea calitativă și reprezentarea neuronală distribuită, adaptabilă a experienței interferează des, în dezvoltări teoretice sau practice combinate, hibride, denumite sisteme neuro-fuzzy.

Teoria matematică a mulțimilor și logicii fuzzy a cunoscut o dezvoltare exponențială pe parcursul celor 45 de ani de la introducerea ei ([ZAD1965]), motivată atât de atractivitatea raționamentelor – fiind asemănătoare mecanismelor raționale umane uzuale, cât și aplicării cu succes într-o serie largă de domenii ingineresti. Expunerile teoretice ale logicii fuzzy, cu toate aspectele ei, interferează strâns cu alte capitole ale matematicii, dar și cu domenii ale ingineriei, și utilizează instrumente abstracte sau practice specifice.

Dintre lucrările importante de amintit pentru o introducere mai detaliată a aparatului teoretic privind logica fuzzy, în general, și a aplicațiilor ingineresti fuzzy și hibride neuro-fuzzy, în

particular (printre care și controlul în logică fuzzy), se pot aminti monografiile: Mamdani – [MAM1985], Driankov ș.a. – [DHR1993], Kosko – [KOS1996], Pedrycz și Gomide – [PEG1998], Passino și Yurkovich – [PAY1998], Zimmermann – [ZIM2001], Jantzen – [JAN2007], dar și multe altele. Pe plan internațional, numărul lucrărilor în domeniu este mare, cele amintite fiind doar exemplele frecvent citate. În limba română se pot aminti: Negoită și Ralescu – [NER1974], Preitl și Precup – [PRP1997].

Aplicații ale logicii fuzzy

Aplicațiile curente în inginerie și mai exact în control automat apelează doar o parte restrânsă a teoriei mulțimilor fuzzy, care se dovedește relativ ușor accesibilă. Prezentarea noțiunilor teoretice preliminare reprezintă o enumerare a elementelor de bază considerate necesare pentru dezvoltarea ulterioară a temei propuse.

Într-o listă a aplicațiilor industriale ale sistemelor cu logică fuzzy ar putea fi amintite:

- echipamente de aer condiționat (reducerea consumului de putere, îmbunătățirea stabilității temperaturii) – Mitsubishi;
- sisteme automate de transmisie, ABS - Daimler-Benz;
- echipamente de uz casnic, reglatoare pentru motoare de viteză redusă, monitorizare trafic – Siemens;
- senzori ieftini (care evaluează imprecizia de achiziție cu sisteme fuzzy);
- aparate electrocasnice - Matsushita;
- camere video cu imagine stabilizată – Panasonic;
- sisteme de reglare pentru motoare cu viteză redusă – Rockwell International Corp.;
- reglatoare de temperatură – Omron;
- cameră video cu autofocalizare, Fotocopiatore – Canon;
- televizoare (control de contrast și luminozitate) - Sanyo, Mitsubishi;
- compresie de date – Mitsubishi;
- acționări electrice de curent continuu și alternativ - SGS Thomson;
- controlul cu DSP a motorului asincron - Texas Instruments;
- sistem de aer condiționat cu microcontroller de 8-biți – Mitsubishi;
- microprocesoare de 4-biți pentru aplicațiile de mașini de spalat;
- camere video cu microprocesoare de dimensiune redusă - Sanzo-Fischer;
- cuptoare de ciment;
- tratamentul de spălare a deșeurilor – INFORM;
- sisteme expert descrise în limbaj natural;
- sisteme de susținere a deciziilor bancare - Zimmerman & INFORM Corporation;
- reactoare de polimerizare (controlul temperaturii);
- coloane de distilare (producția de etilenă);
- instalații ce utilizează energie solară (compensată cu energie electrică);
- macarale comandate prin reglatoare fuzzy;
- vehicule subacvatice autonome – Florida Atlantic University;
- navigația adaptivă terenului și vehicule terestre autonome – NASA (Sojourner);
- aplicații software de asistență în luarea deciziilor;
- aparatură medicală de supraveghere și control (restrâns).

Tabelul 1.1. Principalele arii de cercetare în domeniul logicii fuzzy.

<ul style="list-style-type: none"> • Fundamentele teoriei fuzzy - Algoritmi de inferență fuzzy - Tehnici fundamentale pentru dezvoltarea calculatoarelor fuzzy
<ul style="list-style-type: none"> • Procesare fuzzy a informației - Sisteme de decizie fuzzy - Prelucrarea imaginii - Sisteme expert fuzzy de tip shell - Sisteme de diagnoză fuzzy pentru centrale electrice - Sisteme de comunicare în limbaj natural pentru roboți - Evaluare intelectuală și înțelegere semantică
<ul style="list-style-type: none"> • Controlul fuzzy - Studiul teoriei controlului fuzzy - Dezvoltarea și producția uneltelor pentru structurile de bază ale sistemelor de control fuzzy - Bazele de dezvoltare a calculatoarelor fuzzy
<ul style="list-style-type: none"> • Calculatoare fuzzy - Arhitectura fuzzy - Software fuzzy - Hardware fuzzy
<ul style="list-style-type: none"> • Tehnici de recunoaștere - Procesarea în timp real a imaginilor - Recunoașterea caracterelor
<ul style="list-style-type: none"> • Modelarea proceselor cu informație obținută de la senzori - Interfața umană pentru procesele din domotica - Interfața umană pentru sistemele rapide și/sau instabile
<ul style="list-style-type: none"> • Tehnici de procesare inteligentă a informației - Sisteme inteligente de evaluare a informației - Tehnici fuzzy de regăsire a informației - Procesarea fuzzy în mecanismul de asociere
<ul style="list-style-type: none"> • Studiul comportamentului uman și a fenomenelor sociale - Estimarea fiabilității a sistemelor pe scară largă - Dezvoltarea tehnicilor fuzzy și a sistemelor pentru managementul problemelor sociale
<ul style="list-style-type: none"> • Studiul fenomenelor naturale - Tehnici fundamentale pentru predicția cutremurelor - Dezvoltarea unor simulatoare pentru predicția poluării mediului - Modelarea dezvoltării platformelor industriale

1.3. Rețele neurale

Încă de la începutul cercetărilor în inteligența artificială s-au conturat două direcții rivale ce constituie și acum modelele dominante:

- *Paradigma logico-simbolică* care presupune mecanisme de reprezentare simbolică a cunoștințelor și utilizarea unor modele logice pentru a deduce noi cunoștințe din faptele memorate în baza de cunoștințe a sistemului.
- *Paradigma conexionistă*, bazată pe calculul neuronal, care a generat realizări concrete cunoscute sub numele de rețele neuronale artificiale (pe scurt rețele neuronale sau rețele neurale).

Un aspect important, care este probabil principalul argument în dezvoltarea cu succes a modelelor conexioniste, respectiv a rețelelor neuronale, este capacitatea acestora de a învăța din exemple. În mod tradițional pentru a rezolva o problemă trebuie să elaborăm mai întâi un model (matematic, logic, lingvistic, etc) al acesteia pe baza căruia se stabilește sub forma unei succesiuni de operații algoritmul de rezolvare a problemei.

Există, însă, probleme practice de mare complexitate pentru care chiar construcția modelului prezintă dificultăți mari și stabilirea unui algoritm, fie el și unul aproximativ, este dificilă sau chiar imposibilă.

Caracteristic rețelelor neuronale este faptul că, pornind de la o mulțime de exemple, ele sunt capabile să sintetizeze în mod explicit un anumit model al problemei. Se poate spune că o rețea neuronală își construiește singură algoritmul pentru rezolvarea unei probleme, dacă are la dispoziție o mulțime reprezentativă de cazuri particulare (exemple de instruire). Așadar rețeaua neuronală extrage de fapt informația prezentă în mulțimea de instruire. Această etapă (fază) de funcționare a unei rețele neuronale se numește de instruire sau antrenare. În faza de lucru – sau de referință – rețeaua va folosi informația achiziționată în etapa de instruire pentru a trata situații de aceeași natură cu cele conținute în mulțimea de instruire.

Capacitatea rețelei neuronale de a rezolva probleme practice complexe, utilizând o mulțime de exemple de instruire conferă acestora un potențial de aplicabilitate extrem de larg. Rețelele neuronale sunt foarte potrivite pentru a rezolva aceleași tipuri de probleme ca și creierul.

Rețelele neuronale artificiale (RN) reprezintă modele simplificate ale sistemului nervos central. Ele au abilitatea de a răspunde la stimuli de intrare și de a se adapta la mediu. Modelele RN oferă cea mai promițătoare procedură pentru construcția sistemelor de calcul inteligente. Procesarea paralelă și distribuită într-o rețea neuronală artificială este cea mai bună cale pentru implementarea acestor sisteme.

Cu o desfășurare doar puțin mai lungă, dezvoltarea modelelor neuronale, a structurilor de rețele și a algoritmilor a fost de asemenea amplă. Structurile și metodele matematice propuse simulează pe cele naturale, specifice inteligenței umane, de unde sunt inspirate.

Primele studii făcute asupra creierului prezintă drept constituenți de bază ai creierului, neuronii, prima contribuție majoră în termeni de modelare matematică a rețelelor de neuroni fiind realizată în 1943 de McCulloch și Pitts. Conceptul de învățare în creier a fost propus de D. Hebb în 1949, care arăta modificarea continuă a ponderilor conexiunilor neuronale pe măsură ce organismul învață diferite task-uri funcționale.

În 1958, F. Rosenblatt a propus un model neuronal numit *perceptron*, care poate învăța să clasifice anumite mulțimi de pattern-uri, similare sau distincte. Rețeaua este formată dintr-un singur strat de neuroni, fiecare neuron având caracteristicile propuse de McCulloch și Pitts. Legăturile sinaptice din rețea erau modificate în funcție de o lege de învățare ('adaptare'), care minimizează eroarea dintre răspunsul dorit și cel real al rețelei. La puțin timp după apariția lucrării lui Rosenblatt asupra rețelei de tip perceptron, Widrow și Hoff au propus o rețea cu un singur strat numită *Adaline*, care diferă de perceptron din punctul de vedere al funcției de activare și al legii de învățare.

Rezultatele obținute de Minski și Papert referitoare la faptul că perceptronul și rețeaua ADALINE nu pot realiza operații logice simple cum ar fi XOR, au condus la o diminuare a interesului pentru acest domeniu. Deși unii cercetători, ca Amari sau Grossberg, au continuat studiile asupra modelării și comportării neuronilor biologici și au dezvoltat arhitecturi și metode matematice pentru task-uri de extragere de caracteristici și clasificare de pattern-uri, cercetările au luat o amploare deosebită doar odată cu apariția lucrărilor lui Hopfield. În aceste lucrări sunt descrise elemente de procesare de tip neuron cu conexiuni simetrice (rețele Hopfield). S-a arătat că acest tip de rețele poate fi utilizat ca *Memorii Adresabile prin Conținut* (Content Addressable Memories - CAM). Deși rețeaua dezvoltată de Hopfield este puțin plauzibilă privită din perspectivă biologică, ea s-a dovedit utilă în rezolvarea anumitor probleme, cum ar fi problema de optimizare, a "comis-voiajorului".

În 1986, Rumelhart a dezvoltat un algoritm de învățare numit "*error back-propagation*" (propagarea înapoi a erorii), care putea fi folosit pentru antrenarea perceptronilor cu mai multe straturi de neuroni (arhitecturi multistrat). În acest fel arhitecturile multistrat au devenit capabile să formeze regiuni de decizie complexe, reușind astfel să rezolve probleme considerate nerezolvabile de către perceptron (problema XOR). Imediat au început să apară aplicații ale acestor rețele în diferite domenii, ca inginerie, economie sau științe exacte.

Ca urmare a cercetărilor întreprinse de Rumelhart, au apărut o serie de alte arhitecturi și algoritmi de învățare, multe dintre acestea fiind dezvoltate înainte ca rezultatele acestui grup să fie publicate. Dintre acestea fac parte: *Cerebellar Model Articulate Controller (CMAC)* dezvoltată de Albus, *rețelele cu auto-organizare* (Self-Organizing Networks) dezvoltate de Kohonen, *mașina Boltzmann* – Hinton și Ackley –, *Teoria Rezonanței Adaptive* (ART - Adaptive Resonance Theory) dezvoltată de Grossberg. În anii următori ani au apărut o serie de aplicații ale rețelelor neuronale artificiale: clasificare de pattern-uri, sisteme de conducere cu auto-adaptare pentru sisteme dinamice ș.a. În multe din aceste aplicații, rețelele neuronale au fost utilizate drept clasificatori în cazuri în care pattern-urile utilizate ca înțrări au natură statică. Studiile asupra unor task-uri mult mai complexe, cum ar fi predicție, recunoaștere de voce sau control adaptiv, au arătat că rețelele neuronale artificiale oferă o soluție viabilă în cadrul acestor domenii.

Arhitectura RN joacă un rol important în orice aplicație. De exemplu, o rețea de tip Hopfield este mult mai potrivită ca o memorie adresabilă prin conținut decât ca o rețea pentru predicția semnalelor. O rețea recurentă cu conexiuni asimetrice poate fi mult mai potrivită decât o rețea feed-forward într-o aplicație de control sau identificare de sistem. Din această cauză, alegerea arhitecturii este dependentă de aplicație, și în consecință de alegerea algoritmului de învățare (algoritmul de învățare nu este o entitate separată, ci trebuie văzut ca parte a rețelei).

În ultimii ani algoritmul back-propagation a devenit de departe cel mai popular algoritm de antrenare pentru rețelele neuronale artificiale. Însă, așa cum a arătat Hinton, în ciuda rezultatelor obținute în problemele de complexitate redusă, acest algoritm s-a dovedit inadecvat pentru task-urile complexe. Timpul necesar procesului de învățare cu algoritmul back-propagation crește odată cu complexitatea procesului, și este de specificat faptul că *cel mai popular algoritm de antrenare pentru rețelele neuronale nu este plauzibil biologic* (nu există nici o evidență a faptului că sinapsele biologice pot fi utilizate în direcție inversă pentru propagarea erorii sau că neuronii pot “propaga” derivata erorii).

Creșterea interesului privind modelele conexiuniste a fost determinată de dezvoltarea unor noi arhitecturi, a unor noi algoritmi de învățare, de paralelismul intrinsec al unor astfel de structuri de procesare precum și de rezultatele obținute în domeniul implementărilor VLSI și a proiectării asistate de calculator care au permis construirea unor mașini de calcul cu un înalt grad de paralelism.

În domeniul reglării automate modelele conexiuniste oferă o nouă modalitate de abordare a metodelor de identificare și control atât pentru sisteme dinamice liniare, cât și pentru sisteme neliniare. Într-un număr de probleme practice aceste tipuri de modele au demonstrat robustețe și adaptabilitate în prezenta incertitudinilor parametrice, pentru rejecția zgomotului de măsură sau în cazul perturbațiilor din mediul de lucru. Pot fi menționate aici și alte domenii de aplicabilitate: procesare de semnale și predicție, recunoaștere de voce, detecție de defecte. *Din punct de vedere sistemic, se poate considera că o rețea neuronală reprezintă o clasă de aplicații neliniare parametrizată corespunzător.*

În ultimii ani s-au făcut cercetări în direcția obținerii unor algoritmi de antrenare supervizată corespunzători problemelor de aproximare funcțională, fiind demonstrat faptul că rețelele multistrat cu cel puțin un strat ascuns de neuroni și funcții de activare sigmoideale pot aproxima o clasă largă de neliniarități. *Din punctul de vedere a sistemelor de control, o rețea neuronală feed-forward reprezintă o aplicație statică neliniară, rețelele neuronale recurente reprezentând sisteme feedback neliniare.* La fel ca în cazul oricărei proceduri clasice de estimare de parametri, parametrii rețelei trebuie să fie adaptați astfel încât să modeleze o funcție sau un sistem.

Dintre lucrările importante de amintit pentru o introducere mai detaliată a aparatului teoretic privind rețelele neuronale, în general, și a aplicațiilor hibride neuro-fuzzy, în particular, se pot aminti monografiile: Kosko – [KOS1991], Nie și Linkens – [NIL1995], Kasabov – [KAS1998], Nguyen et al. – [NPW2003]. Pe plan internațional, numărul lucrărilor în domeniu este mare, cele amintite fiind doar exemple frecvent citate sau utile pentru înțelegerea noțiunilor de bază. În limba română se pot aminti lucrările: Dumitrescu și Costin – [DUC1996], Comnac – [COM1999], Matcovschi și Păstrăvanu – [MAP2008].

Aplicații ale rețelelor neuronale artificiale

Rețelele neuronale artificiale și-au dovedit eficiența în taskuri de procesare, cum ar fi:

- recunoaștere de forme (vorbi, imagini vizuale);
- clasificări;
- compresii de date;
- modelare;
- rezolvarea problemelor combinatoriale;

- conducere adaptivă;
- fuziune de date;
- filtrare de zgomot.

Între domeniile care utilizează aplicații ale rețelelor neuronale se pot aminti

- aeronautica (sisteme de conducere, simulare componente, detecție de defecte, reconfigurare, identificare dinamici neliniare, detecția unor situații de eroare specifice prin utilizarea modelelor multiple, conducere adaptivă)
- domeniul financiar - bancar (evaluarea riscurilor creditelor), asigurări
- procese de producție (verificarea procesului de fabricație, analiză și proiectare de produs, analiza calității sudurii, predicția calității hârtiei, planificare și management, modelarea proceselor chimice, etc)
- domeniul medical (analiza celulelor care favorizează cancerul, analiza EEG și ECG, proiectare proteze, optimizarea timpului de transplant, reducerea costurilor spitalicești)
- domeniul militar (ghidare rachete, urmărire ținte, recunoaștere de fizionomie, tipuri noi de senzori, radare și procesare de semnal video incluzând compresie de date, extragere de caracteristici și rejectia zgomotelor, identificare de semnal și imagine)
- robotica (generare de traiectorie, comanda manipuletoarelor, sisteme de prelucrare de imagini, coordonare manipulare și contact, comanda mișcării în cazul roboților bipezi, parcare automată)
- telecomunicații (compresie de date și imagini, sisteme de informare automată, traducere în timp real a vorbirii)

Se aminti și detalia aplicațiile rețelelor neuronale multistrat antrenate prin metoda propagării înapoi a erorii:

- clasificare și diagnoză: clasificare celule pentru diagnoza cancerului; detecție defecte în cadrul sistemelor de comutare în telefonie mobilă.
- aplicații de conducere și optimizare: sisteme de ghidare pentru vehicule autonome; regulatoare pentru conducerea proceselor din industrie; produse de larg consum ce conțin regulatoare neuronale, simple sau în combinație cu logica fuzzy (instalații de aer condiționat, încălzitoare electrice și cu kerosen, mașini de fotocopiat, procesoare de texte).
- aplicații de predicție și prognoză: domeniul financiar (prognoze financiare), bănci (analiza gradului de încredere acordată creditelor); predicția evoluției fluxului de pasageri pe aeroporturi; sisteme energetice - prognoza cererii de energie a consumatorilor; fuziunea datelor de la senzori și predicția defectelor.
- recunoaștere de caractere, vorbire, recunoaștere de imagini și clasificare: analiza de semnale uni- și bi-dimensionale (electrocardiograme, electroencefalograme); analiza de grafice și monitorizare sisteme de alarmare; recunoaștere caractere scrise de mână, conversie text-vorbire.

1.4.Sisteme neuro-fuzzy

Fiecare tehnică inteligentă are proprietăți tipice care îi conferă aplicabilitatea pentru anumite probleme și prezintă dezavantaje în rezolvarea altora. De exemplu, rețelele neuronale au

avantajul recunoașterii modelelor, însă nu pot explica modul în care au ajuns la o concluzie. Pe de altă parte, sistemele fuzzy pot raționa cu informații imprecise și își pot explica raționamentul, dar nu pot achiziționa automat regulile pe care le utilizează pentru luarea deciziilor.

Aceste limitări au fost principalul motiv pentru care s-a încercat crearea de sisteme inteligente hibride, în care două sau mai multe tehnici pot fi combinate pentru a depăși dezavantajele tehnicilor individuale.

Modelarea cu reguli fuzzy, de tip *dacă-atunci*, este în general potrivită pentru modelarea calitativă. Totuși, în multe situații, datorită incompletitudinii cunoașterii, inerente unor situații reale, această abordare se dovedește insuficientă. De aceea se face apel la unele metode conexioniste, cum ar fi rețelele neuronale. Sistemele neuro-fuzzy sunt astfel capabile să învețe din exemple, să generalizeze pe baza cunoștințelor acumulate și, pe baza datelor primite, să sintetizeze cunoștințe în forma regulilor fuzzy.

În ultimii ani, sistemele hibride au suscitat un deosebit interes. Această abordare s-a dovedit încununată de succes în diferite domenii, precum diagnoza echipamentelor industriale, robotica sau controlul proceselor dinamice. Motivul principal al studierii sistemelor neuronale hibride este crearea de sisteme de reprezentare sub-simbolică, în special a rețelelor neuro-fuzzy. Din punctul de vedere al științelor cognitive, reprezentarea neuronală oferă avantajul omogenității, a capacității de învățare prin exemple și a generalizării, precum și efectuarea de sarcini distribuite, în medii cu date incomplete sau afectate de zgomote. Din punctul de vedere al sistemelor bazate pe cunoștințe, reprezentările simbolice au avantajul interpretării umane, al controlului explicit și al abstractizării cunoașterii.

Cele două abordări pot fi utilizate în mod complementar. Aceasta este premisa creării de sisteme hibride inteligente, care combină trăsături conexioniste (neuronale) și simbolice (fuzzy). În astfel de sisteme, mai întâi se introduc informații simbolice în rețeaua neuronală, apoi se folosesc exemplele de instruire pentru rafinarea cunoștințelor inițiale. În final, se procesează rezultatul pentru un set de intrări și, prin metode specifice, se extrag informații simbolice din rețeaua antrenată.

Din momentul în care sistemele fuzzy au fost puse în practică în aplicațiile industriale, proiectanții și-au dat seama de dificultățile care surveneau în mod inerent. Problema găsirii funcțiilor membre potrivite este de multe ori o chestiune de generare de valori și îndepărtarea celor eronate. Aceste soluții sunt greu de găsit, mai ales într-un proces industrial dinamic. De aceea s-a ivit ideea aplicării algoritmilor de învățare sistemelor fuzzy, ceea ce a determinat apariția controllerelor fuzzy adaptabile sau cu auto-organizare.

Sarcinile de lucru complexe pot da naștere la minime locale, de aceea paradigma învățării prin exemple se poate folosi mai mult pentru probleme simple. O rezolvare este împărțirea problemei în sub-probleme, rezolvate de diferite module neuro-fuzzy. În timp ce logica fuzzy asigură mecanismele de inferență în cazul incertitudinii, rețelele neuronale oferă avantajele învățării, adaptabilității, toleranței la defecte, paralelismului și generalizării.

Sistemele neuro-fuzzy moderne au caracteristici care le fac adecvate controlului unor procese industriale: sunt sisteme fuzzy instruite printr-un algoritm de învățare derivat în general din rețele neuronale, pot fi reprezentate printr-o arhitectură de rețea neuronală feed-forward multistrat și pot fi interpretate în termeni de reguli „dacă-atunci”. Totuși, într-un sistem real,

pot exista zeci de semnale de intrare. Un număr excesiv de intrări poate deteriora transparența modelului considerat și crește complexitatea calculelor. De aceea, este necesară selectarea intrărilor relevante și considerarea de priorități.

Din punct de vedere teoretic, sistemele fuzzy și rețelele neuronale sunt echivalente, deoarece sunt inter-convertibile:

- orice sistem fuzzy bazat pe reguli poate fi aproximat de o rețea neuronală;
- orice rețea neuronală poate fi aproximată de un sistem fuzzy bazat pe reguli.

În practică însă, fiecare tehnică are avantajele și dezavantajele sale (vezi tabelul 1.2).

Tabelul 1.2. Privire comparativă a avantajelor și dezavantajelor sistemelor fuzzy și rețelelor neuronale.

	Avantaje	Dezavantaje
Sisteme fuzzy	<ul style="list-style-type: none"> - Încorporează cunoștințe predefinite bazate pe reguli - Sunt interpretabile (reguli) - Oferă o implementare simplă - Cunoștințele sunt disponibile în orice moment 	<ul style="list-style-type: none"> - Pot învăța fără nici o informație inițială - Există mai multe paradigme de învățare, potrivite diverselor situații - Există o multitudine de algoritmi de învățare disponibili în literatura de specialitate
Rețele neuronale	<ul style="list-style-type: none"> - Nu pot învăța - Trebuie să dispună de reguli stabilite a-priori - Nu există metode formale pentru ajustarea regulilor 	<ul style="list-style-type: none"> - Sunt „cutii negre” sub-simbolice - Nu pot utiliza cunoștințe a-priori - Necesită algoritmi de învățare complecși - Nu există garanția că învățarea converge spre soluție - Prezintă dificultăți în extragerea cunoștințelor structurale

Sistemele fuzzy prezintă în principal avantajul explicării comportamentului pe baza unor reguli și deci performanțele lor pot fi ajustate prin modificarea acestora. Totuși, achiziționarea cunoștințelor este dificilă iar universul de discurs al fiecărei variabile de intrare trebuie divizat în mai multe intervale și de aceea, aplicațiile sistemelor fuzzy sunt restrânse la domeniile în care sunt disponibile cunoștințele experților iar numărul variabilelor de intrare este relativ mic.

Pentru rețelele neuronale, cunoștințele sunt achiziționate automat printr-un algoritm, dar procesul de învățare este relativ lent iar analiza rețelei antrenate este dificilă (o rețea neuronală este deseori considerată o „cutie neagră”, în care avem acces numai la intrări și la ieșiri, fără a ști precis ce se întâmplă în interior). Nu este posibilă nici extragerea cunoștințelor structurale (reguli) din rețeaua antrenată, nici integrarea de informații speciale despre problemă, pentru a simplifica procedura de învățare.

Utilizarea sistemelor hibride neuro-fuzzy minimizează dezavantajele și maximizează avantajele celor două tehnici. Rețelele neuronale sunt folosite pentru adaptarea funcțiilor de

apartenență ale sistemelor fuzzy în aplicațiile de control automat al proceselor dinamice. Deși logica fuzzy permite codarea cunoștințelor de tip expert prin termeni lingvistici, de obicei proiectarea și reglarea funcțiilor de apartenență care definesc cantitativ acești termeni este un demers greoi. Tehnicile de învățare ale rețelelor neuronale pot automatiza procesul și deci pot reduce substanțial costul și timpul de dezvoltare a aplicațiilor, conducând și la creșterea performanțelor.

Pentru a depăși problema achiziției cunoștințelor, rețelele neuronale pot fi proiectate să extragă automat reguli fuzzy din date numerice. Alte abordări presupun folosirea rețelelor neuronale pentru optimizarea unor anumiți parametri ai sistemelor fuzzy sau pentru preprocesarea datelor de intrare în sistemele fuzzy.

1.5. Aplicații ale inteligenței computaționale în automatică

Așa cum am detaliat în secțiunile anterioare, tehnicile de inteligență computațională sunt intens utilizate în domeniul automaticii, în toate aspectele ce țin de proiectarea și funcționarea unui sistem automat. Pentru a sintetiza, vom enumera câteva probleme specifice automaticii în care logica fuzzy și/sau rețelele neuronale oferă metode și soluții viabile:

- reglarea fuzzy – se referă la utilizarea unor regulatoare ale căror funcționare implică utilizarea unei metode de inferență fuzzy;
- modelarea fuzzy – determinarea unui model exprimat prin reguli fuzzy, ce poate fi utilizat în proiectarea unui regulator fuzzy;
- control neuronal direct – utilizarea într-un sistem de reglare a unui regulator adaptiv care utilizează o arhitectură neuronală și un algoritm de instruire;
- control neuronal indirect – utilizarea unei structuri în care un regulator clasic este modificat în timpul funcționării de către un element suplimentar care realizează adaptarea pe baza unei arhitecturi neuronale și a unui algoritm de instruire;
- modelarea neuronală – utilizarea rețelelor neuronale pentru determinarea unui model al procesului sau pentru determinarea parametrilor acestuia.

Între acestea, reglarea fuzzy a dovedit o aplicabilitate extinsă și chiar avantaje în diverse situații. Unii autori subliniază o robustețe sporită a sistemelor de reglare fuzzy în cazul modificării valorilor unor parametri ai procesului, modificările acestea fiind posibile pentru sisteme care funcționează o perioadă mare de timp. De asemenea, caracteristica neliniară a unui sistem de reglare fuzzy poate oferi o îmbunătățire a performanțelor dinamice, în special în cazul proceselor lente.

Să ne reamintim ...



Inteligența computațională este parte integrantă a domeniului inteligenței artificiale și face referire la acele tehnici de calcul care sunt inspirate din caracteristicile raționamentului uman. Între tehnicile de inteligență computațională, cele mai frecvent utilizate în automatică sunt logica fuzzy și rețelele neuronale, laolaltă cu tehnici combinate neuro-fuzzy. Acestea, împreună cu algoritmi genetici și modelele probabilistice, definesc conceptul de soft computing, ale cărui obiective sunt atractive și eficiente inclusiv în automatică.



Test de evaluare a cunoștințelor

1. Descrieți conceptul de soft computing și scopul acestuia.
2. Dați exemple de aplicații ale raționamentului fuzzy în automatică și detaliați prin exemple concrete de utilizare a sistemelor cu logică fuzzy în aplicații, echipamente și sisteme.
3. Enumerați și detaliați câteva exemple de aplicații ale rețelelor neuronale.
4. Care sunt principalele tipuri de probleme specifice automatizării în care se pot utiliza tehnicile de inteligență computațională.
5. Care sunt avantajele și dezavantajele tehnicilor de inteligență artificială prezentate? Ce concept încearcă să valorifice avantajele și să reducă dezavantajele fiecărei tehnici.



Test de autoevaluare a cunoștințelor

1. Raționamentul fuzzy:
 - a) simulează funcționarea neuronilor biologici
 - b) simulează raționamentul inexact uzual
 - c) nu are legătură cu aplicațiile ingineresti
2. Între aplicațiile logicii fuzzy în automatică se pot aminti:
 - a) simularea raționamentului specific proiectantului
 - b) controlul fuzzy
 - c) modelarea fuzzy a proceselor

3. Între aplicațiile practice cu sisteme de reglare fuzzy se întâlnesc:
 - a) sisteme de control ale unor procese și echipamente industriale
 - b) echipamente electronice și electrocasnice
 - c) sisteme de producere a energiei
4. Logica fuzzy și rețelele neuronale:
 - a) sunt domenii concurente
 - b) sunt sub-domenii care definesc în mod exclusiv conceptul de soft computing
 - c) sunt sub-domenii ale conceptului de soft-computing
5. Sistemele neuro-fuzzy:
 - a) sunt sisteme care utilizează atât elemente de logică / raționament fuzzy, cât și arhitecturi și modele neuronale, în diverse combinații
 - b) sunt sisteme fuzzy care simulează funcționarea rețelelor neuronale biologice
 - c) sunt rețele neuronale cu funcționare inexactă
6. Rețelele neuronale artificiale:
 - a) au ca scop realizarea unor simulatoare ale rețelelor neuronale biologice
 - b) sunt inspirate din modul de funcționare al rețelelor neuronale biologice
 - c) utilizează algoritmi de instruire pentru a implementa abilitatea de a învăța specifică rețelelor neuronale biologice
7. Instruirea rețelelor neuronale poate fi :
 - a) supervizată
 - b) ne-supervizată
 - c) fuzzy
8. Algoritmul propagării înapoi a erorii este:
 - a) un algoritm de instruire a rețelelor neuronale multistrat
 - b) un algoritm de modelare fuzzy
 - c) un algoritm de control neuronal direct
9. Controlul neuronal direct constă în:
 - a) realizarea unui sistem de reglare bazat pe experiență
 - b) utilizarea unui model neuronal în proiectarea sistemului de reglare

c) utilizarea unui regulator principal bazat pe o rețea neuronală

10. Modelarea fuzzy constă în:

- a) obținerea unui model fuzzy al procesului
- b) determinarea într-o manieră aproximativă a modelelor proceselor
- c) realizarea unui model a rețelelor neuronale