

TEZĂ DE DOCTORAT

ROBOTIZAREA OPERAȚIILOR DE ANALIZĂ
METALOGRAFICĂ

BIBLIOTECA CENTRALĂ
UNIVERSITATEA "POLITEHNICA"
TIMIȘOARA



00186212

CONDUCĂTOR ȘTIINȚIFIC

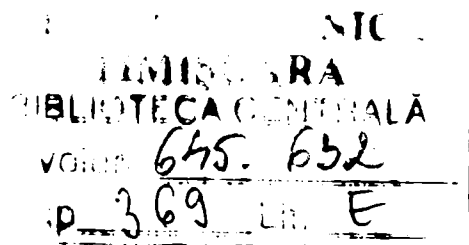
Prof. Dr. Ing.Dr.H.C.Mult.

FRANCISC VILIAM KOVÁCS

DOCTORAND

Șef lucrări ing.

TIBERIU VESSELENYI



2005

Prefață.

Analiza metalografică este un domeniu deosebit de interesant al științelor tehnice. Aceleași elemente chimice pot da naștere unor structuri complexe, diferite ca proprietăți mecanice și ca aspect vizual. Relevarea și identificarea acestor structuri presupune o vastă experiență și cunoaștere și nu de puține ori o gândire subtilă. Din această cauză robotizarea operațiilor de analiză metalografică este încă la începuturi.

Teza de doctorat abordează câteva din problemele majore ale acestui domeniu și oferă soluții în rezolvarea lor. Rămân însă întrebări deschise a căror soluționare poate constitui subiectul unor viitoare cercetări.

Tematica tezei de doctorat se axează în jurul a trei grupe de probleme. Prima dintre acestea tratează robotizarea operațiilor de analiză metalografică în ansamblu ei, prin proiectarea, realizarea practică și experimentarea unei celule flexibile concepute pentru acest domeniu. A doua problemă abordată este cea a șlefuirii eșantioanelor metalografice utilizând legi de reglare fuzzy și neuronale. Ultimul domeniu este cel al procesării imaginilor aplicate în analiza stării suprafețelor eșantioanelor și a microstructurilor.

Teza de doctorat este structurată pe 10 capitole și conține: 214 figur explicative, 5 tabele, 27 anexe și 218 referințe bibliografice.

Doresc să aduc mulțumiri Domnului Profesor Dr.Ing.Dr.H.C.Mult. Francisc Viliam Kovács pentru îndrumarea și sprijinul acordat pe perioada celor 8 ani de elaborare a tezei.

Deasemenea mulțumesc Domnului Profesor Dr.Ing. Teodor Maghiar pentru sprijinul acordat în dotarea laboratorului de Mecatronică ce a făcut posibilă realizarea părții experimentale.

Și nu în ultimul rând mulțumesc familiei și colegilor de la Universitatea din Oradea, Moga Ioan, Mudura Pavel, Barbás Tiberiu, Țarcă Radu, Blaga Florin, Rus Alexandru și Ungur Petru pentru sprijinul acordat.

Autorul

1.Introducere	1
2.Stadiul actual al analizei metalografice	3
2.1.Considerații generale.....	3
2.1.1.Spațiul de lucru al laboratorului.....	4
2.1.2.Funcționarea laboratorului metalografic.....	4
2.1.3.Principii de structurare a unui laborator de analiză metalografică.....	4
2.2.Tehnologia pregătirii epruvetelor metalografice.....	5
2.3.Mașini și echipamente de pregătire a epruvetelor.....	7
2.4.Analiza imaginilor microstructurilor.....	9
2.4.1.Considerații generale.....	9
2.4.2.Analiza macroscopică.....	10
2.4.3.Analiza microscopică.....	10
2.5.Concluzii.....	11
3.Stadiul actual în robotizarea pregătirii și analizei metalografice	14
3.1.Introducere.....	14
3.2.Elemente de comandă a roboților utilizând logica fuzzy și rețele neuronale artificiale.....	14
3.2.1.Considerații generale.....	15
3.2.2.Noțiuni privind logica fuzzy.....	24
3.2.3.Stadiul actual privind rețele neuronale artificiale.....	26
3.3.Procesarea computerizată a imaginilor.....	29
3.3.1.Considerații generale.....	29
3.3.2.Îmbunătățirea imaginii.....	35
3.3.3.Filtrare.....	36
3.3.4.Transformări integrale.....	36
3.3.5.Morfologie.....	37
3.3.6.Segmentare.....	38
3.3.7.Recunoaștere.....	39
3.3.8.Metode de caracterizare și recunoaștere a texturilor.....	40
3.3.9.Sinteza cercetărilor recente în domeniul analizei de imagini.....	44
4.Sisteme automatizate pentru laboratoare	59
4.1.Dezvoltarea sistemelor automatizate pentru laboratoare.....	59
4.2.Realizări în domeniul laboratoarelor automatizate.....	60
4.3.Concluzii.....	63
5.Celula flexibilă pentru laborator metalografic (CFLM1)	64
5.1.Considerații generale.....	64
5.2.Schema de amplasament a celulei flexibile.....	64
5.3.Descrierea funcțională a celulei flexibile.....	65
5.3.1.Schema de ansamblu a celulei.....	66
5.3.2.Schema fluxului epruvetelor în cadrul celulei flexibile.....	67

5.4.Componente și subsisteme ale celulei flexibile.....	68
5.4.1.Structura ierarhică a subsistemelor componente ale celulei flexibile.....	68
5.4.2.Subsistemul de transfer Ro1.....	69
5.4.3.Subsistem pentru determinarea stării suprafeței după șlefuire.....	69
5.4.4.Subsistemul de transfer Ro2 (AID V6).....	70
5.4.5.Subsistem pentru determinarea microstructurii.....	70
5.4.6.Subsistem pentru stocarea epruvetelor.....	71
5.5.Descrierea celulei flexibile sub aspect informațional.....	71
5.5.1.Structura informațională a celulei.....	71
5.5.2.Limbajul programelor de aplicație dezvoltat pentru CFLM1.....	73
5.5.2.1.Introducere.....	73
5.5.2.2.Conceptul programării orientate pe obiecte utilizat în cadrul programelor de aplicație CFLM1.....	73
5.5.2.2.1.Considerații generale.....	73
5.5.2.2.2.Clase de obiecte.....	74
5.5.2.3.Specificarea sintaxei limbajului programelor de aplicație sursă.....	81
5.5.2.3.1.Aspecte generale.....	81
5.5.2.3.2.Identificatori, simboluri speciale, separatori.....	81
5.5.2.3.3.Sintaxa și semantica limbajului.....	83
5.5.2.3.4.Exemplu de aplicare a sintaxei.....	84
5.5.2.4.Traducerea programelor de aplicație.....	85
5.5.2.5.Structura programelor de aplicație MATLAB.....	86
5.5.3.Prezentarea modulelor program de comandă a celulei flexibile.....	88
5.5.3.1.Descrierea programului de celulă pentru sistemul Sc1.....	88
5.5.3.2.Descrierea programului de celulă pentru sistemul SC2.....	95
5.6.Procese decizionale în cadrul CFLM1.....	104
5.6.1.Considerații generale.....	104
5.6.2.Tratarea deciziilor în subsistemul 1.....	105
5.6.3.Tratarea deciziilor în subsistemul 2.....	106
5.6.3.1.Considerații generale.....	106
5.6.3.2.Stabilirea ordinii de procesare a epruvetei.....	107
5.6.3.3.Selectarea reactivului.....	116
5.7.Modelarea celulei flexibile CFLM1.....	117
5.7.1.Considerații generale.....	117
5.7.2.Schema bloc și matricile de structură.....	117
5.7.3.Modelarea funcționării cu ajutorul rețelelor Petri.....	117
5.8.Aspecte privind realizarea CFLM1.....	121
6.Dezvoltarea structurii celulei flexibile, CFLM2.....	127
6.1.Necesitatea dezvoltării celulei flexibile.....	127
6.2.Accesul roboților la posturile comune.....	127
6.3.Execuția simultană a sarcinilor.....	128
6.4.Transmiterea informațiilor între sistemele de calcul ale celulei CFLM2.....	132

7.Şlefuirea eşantioanelor metalografice.....	135
7.1.Considerații generale.....	135
7.2.Determinarea matricii complianței structurale a unui dispozitiv de prehensiune a epruvetelor metalografice.....	135
7.3.Utilizarea regulatorului neuronal cu linearizarea recției în comanda sistemului automat de șlefuit eşantioane metalografice.....	140
7.3.1.Introducere.....	140
7.3.2.Considerații privind modelarea sistemului de șlefuire.....	146
7.3.3.Modelarea axei de translație pneumatice cu regulator neuronal.....	148
7.3.4.Utilizarea unui regulator fuzzy pentru comanda în forță/poziție a unei axe de translație cu acționare pneumatică pentru șlefuirea eşantioanelor.....	152
7.3.4.1.Considerații genarale.....	152
7.3.4.2.Modelarea sistemului de șlefuire cu regulator fuzzy.....	154
7.4.Realizarea sistemului experimental cu regulator fuzzy.....	159
7.5.Concluzii privind funcționarea sistemului cu reglare fuzzy după poziție și forță.....	160
8.Analiza automată a stării suprafeței.....	161
8.1.Necesitatea analizei rugozității.....	161
8.2.Caracterizarea rugozității suprafeței.....	162
8.3.Stabilirea metodelor de lucru.....	165
8.4.Considerații asupra relației dintre rugozitatea suprafeței și imaginea acesteia.....	166
8.5.Preprocesare.....	169
8.6.Algoritmi de analiză.....	174
8.6.1.Considerații generale.....	174
8.6.2.Metode fractale.....	174
8.6.3.Descrierea algoritmilor realizați și rezultate experimentale.....	181
8.6.3.1.Tipuri de algoritmi.....	181
8.6.3.2.Metoda numărării casetelor 2D.....	182
8.6.3.3.Metoda numărării casetelor 3D.....	188
8.6.3.4.Metoda spectrală 2D.....	194
9.Analiza computerizată a microstructurilor metalografice.....	201
9.1.Considerații generale.....	201
9.2.Tratarea sarcinilor de tip A.....	202
9.3.Tratarea sarcinilor de tip B.....	206
9.4.Tratarea sarcinilor de tip C.....	206
10. Concluzii și contribuții.....	215
10.1.Concluzii finale.....	215
10.2.Contribuții.....	216
10.2.1.Contribuții originale.....	216
10.2.2.Contribuții personale.....	216

1. Introducere.

În anul 1997, când s-a stabilit tema tezei de doctorat, cele mai avansate realizări, la acea vreme (în domeniul robotizării operațiilor de pregătire și analiză a eșantioanelor metalografice), se constituiau în microscopae echipate cu sisteme de captare a imaginii și câteva tipuri de mașini, specializate în șlefuirea eșantioanelor, fără a exista încercări de integrare a acestora într-un sistem robotizat. Pe parcursul elaborării tezei de doctorat s-a observat o intensificare a cercetărilor în domeniul automatizării și robotizării operațiilor specifice activității de testare în general în toate domeniile științifice (laboratoare biologice, chimice, știința materialelor). În domeniul metalografiei abia în anul 2004 au început să apară sisteme ce au în componența lor microscop, stocator și un robot de deservire a microscopului, fără a integra însă pregătirea eșantioanelor, ce rămâne în continuare în sarcina unor mașini specializate neintegrate în sistem, asistată de către operatori umani.

Deasemenea domeniul analizei imaginilor metalografice rămâne un domeniu deschis, încă nefinalizat, mai ales în ceea ce privește clasificarea, interpretarea și recunoașterea automată a microstructurilor.

Ideea unui laborator metalografic robotizat este susținută de nevoia crescândă pentru un control de calitate eficient al produselor fabricilor din România în contextul integrării europene și a unei concurențe din ce în ce mai mari pe piața mondială. Un laborator de pregătire și analiză automată a eșantioanelor metalografice ar îmbunătăți calitatea produselor la costuri suportabile și ar oferi servicii de analiză și control unui număr important de unități de producție atât în România cât și în țările vecine, deoarece tendința actuală este spre unități de producție mici și medii ce își pot permite cu greu investițiile necesare unor laboratoare proprii.

Structura procesului de producție în general, include procesele de testare și verificare a pieselor produse.

Testarea pieselor se poate referi la verificarea geometriei (formă, dimensiuni și calitatea suprafeței), sau la verificarea proprietăților microstructurale, mecanice sau chimice. În cadrul procesului de producție testarea poate fi realizată înainte sau după diferite faze (testare intermediară), sau după încheierea procesului (testare finală).

Integrarea calculatorului în procesul de producție s-a realizat treptat în diferitele faze ale acestuia apărând astfel noțiunile de concepție constructivă, fabricație și în final testare asistată de calculator (CAD – Computer Aided Design, CAM – Computer Aided Manufacturing, CAT – Computer Aided Testing). Avantajele testării integrate în sistemele moderne de producție (celule și sisteme flexibile) sunt similare cu cele ale integrării CAD și CAM dar mai ales faptul că se realizează cu ajutorul acesteia o calitate superioară a produselor. Rolul pe care îl ocupă CAT în sistemele CIM este descris în [KOVACS1]. Dezavantajele testării automate au efecte doar pe termen scurt prin investițiile ce trebuie realizate (proiectarea sau adăcarea unor echipamente, elaborarea unor metodologii noi de testare).

În cadrul procesului de producție analiza metalografică se situează înainte de realizarea prelucrărilor mecanice și după anumite faze ce pot induce transformări structurale (prelucrări intensive, tratamente termice).

Scopul tezei de doctorat constă, în ultimă instanță, în identificarea unor metode eficiente de robotizare a operațiilor de analiză metalografică. În acest sens se pot defini următoarele obiective :

- a. trecerea în revistă a stadiului actual al analizei metalografice cu identificarea posibilităților de robotizare;
- b. sinteza metodelor moderne de comandă a roboților industriali în vederea aplicării acestora la pregătirea și manipularea eșantioanelor metalografice;
- c. sinteza metodelor de procesare și interpretare a imaginilor suprafețelor prelucrate și a microstructurilor;
- d. trecerea în revistă a sistemelor automatizate pentru laboratoare;
- e. conceperea și realizarea unei celule flexibile experimentale pentru laboratoare metalografice;
- f. identificarea problemelor specifice în cadrul experimentelor cu celula flexibilă realizată și găsirea unor soluții eficiente de rezolvare a acestora;
- g. modelarea și experimentarea unor metode de conducere fuzzy și neuronale utilizate pentru șlefuire eșantioanelor metalografice;
- h. identificarea și rezolvarea unor probleme legate de achiziția, prelucrarea și interpretarea imaginilor în cadrul pregătirii și analizei metalografice.

În cadrul tezei de doctorat s-au tratat obiectivele propuse realizându-se o bază solidă de cunoștințe și metode ce pot fi utilizate în viitoare cercetări sau în aplicațiile practice din domeniu.

2. Stadiul actual al analizei metalografice.

2.1. Considerații generale

Pentru realizarea robotizării operațiilor de pregătire și analiză metalografică este necesar studiul structurii și organizării unui laborator clasic de acest tip, cât și a proceselor aferente, din care să rezulte problemele specifice legate de funcționarea acestuia.

Pentru ca elementele implicate în analiza metalografică să formeze un sistem, funcțiunile lor trebuie corelate folosind un subsistem de comandă adecvat, capabil să transmită informații și comenzi în concordanță cu stările sistemului și să asigure o derulare optimă a proceselor.

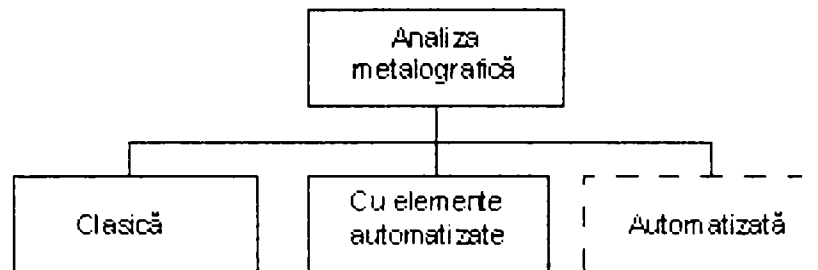


Fig.2.1. Clasificarea sistemelor de analiză metalografică

În figura 2.1. se prezintă clasificarea sistemelor de analiză metalografică. Prin sisteme „clasice” se înțeleg acele sisteme în care atât manipularea și prelucrarea eșantioanelor cât și analiza acestora se realizează manual. Aceste sisteme sunt cele mai răspândite momentan. Variantele moderne de sisteme conțin subsisteme de captură și analiză a imaginilor și subsisteme semiautomate de pregătire a eșantioanelor.

În realizarea studiului se iau în considerare specificațiile privind structura unui laborator metalografic date în bibliografia studiată [BUDAU], [MET]. Aceste specificații conțin elemente importante de care trebuie să se țină cont la proiectarea unor sisteme robotizate de analiză metalografică.

2.1.1. Spațiul de lucru al laboratorului.

Pregătirea eșantioanelor metalografice se poate realiza atât într-un spațiu restâns cât și într-un întreg atelier, depinzând de necesarul de spațiu al echipamentelor de lucru. Cerința de bază este ca echipamentele și utilajele să beneficieze de spațiu suficient pentru a nu se influența reciproc, într-un mod negativ. Astfel, utilajele ce produc zgomote și vibrații sau cele care poluează mediul vor trebui separate de aparatele mai sensibile. Sursele de zgomot și vibrații sunt de regulă asociate operațiilor de tăiere și șlefuire. Operațiile de șlefuire sunt de asemenea și surse de poluare cu particule abrazive. Majoritatea reactivilor sunt soluții acide care pot să ajungă în alte spații de lucru prin evaporare, cauzând corodarea aparatelor din

laborator.

2.1.2. Funcționarea laboratorului metalografic.

Principala funcție a laboratorului metalografic este pregătirea și analiza eșantioanelor, cu ajutorul unor echipamente și într-o manieră tehnologică specifică. Pe lângă acestea există o serie de funcții ce mai trebuie îndeplinite și care sunt amintite mai jos.

Recepționarea materialelor de analizat - materialele primite spre analiză vor trebui înregistrate și urmărite cu strictețe. Este indicat să se înregistreze starea inițială a fiecărui material, încă înainte de a se debita, mai ales în cazurile în care debitarea și eșantionarea acestora modifică starea inițială a eșantionului.

Identitatea eșantionului - trebuie urmărită în fiecare fază a pregătirii. Este indicat ca eșantioanele să fie marcate pentru o identificare sigură. Marcajul va fi înregistrat într-un document împreună cu celelalte caracteristici ale eșantionului.

Standarde și referințe – testele și analizele trebuiesc efectuate conform unor indicații și precizări ale unor standarde recunoscute, evitându-se astfel neîncrederea clienților laboratorului în corectitudinea rezultatelor.

Servicii adiționale – cu toate că principala funcție a laboratorului metalografic este pregătirea eșantioanelor, se pot executa la cerere și alte tipuri de testări cum ar fi : macroduritate (Rockwell, Brinell), control dimensional, determinarea ratei incluziunilor, determinarea mărimii grăunților.

Raportarea rezultatelor – produsul final al laboratoarelor metalografice este raportul cuprinzând rezultatele testelor efectuate.

2.1.3. Principii de structurare ale unui laborator de analiză metalografică.

Structurarea laboratorului metalografic trebuie să țină cont de următoarele elemente: fluxul tehnologic, selectarea echipamentelor, metode de operare și personalul ce deservește laboratorul.

Fluxul tehnologic – descrie succesiunea de operații executate asupra eșantionului de la debitare până la analiza microscopică. Fluxul tehnologic se bazează pe un amplasament corect al echipamentelor, ținând seama și de efectele de poluare menționate anterior. În figura 2.2 este arătat amplasamentul utilajelor și al mobilierului de laborator pentru un laborator de dimensiuni mici.

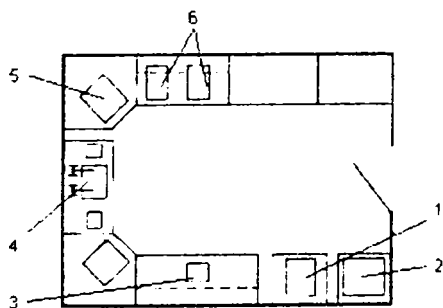


Fig.2.2. Planul de amplasament al unui laborator metalografic de capacitate mică: 1 - microscop; 2 - accesoriile microscopului; 3 - masă de lucru pentru prepararea reactivilor și atacul eșantioanelor cu reactivi; 4 - unitate de spălare; 5 - dulap cu materiale; 6 - echipamente de șlefuit eșantioane.

Amplasamentul din figura 2.2 arată că echipamentele sunt dispuse de-a lungul pereților pentru a lăsa un spațiu cât mai larg în mijlocul încăperii. Acest amplasament se pretează pentru încăperile de dimensiuni mici.

Selectarea echipamentelor – se realizează ținând cont de o serie de factori dintre care cei mai importanți sunt : încărcarea utilajului, dimensiunile eșantioanelor, costul echipamentului, factori ergonomici, posibilitatea unor îmbunătățiri sau achiziții ulterioare.

În afară de acești factori este bine ca înaintea achiziționării unor echipamente să se clarifice încadrarea parametrilor laboratorului în categoriile enumerate mai jos:

Sector de activitate : materii prime, turnătorie, tratamente termice, electronică, geologie, nemetale, educație.

Materialul eșantionului : materiale feroase și aliajele lor, oțeluri aliate, materiale neferoase și aliajele lor, titan, zirconiu, carburi și materiale extradure, materiale compozite, metale nobile, materiale toxice, nemetale.

Numărul de eșantioane prelucrate pe zi : de la 0 la 100; de la 101 la 200.

Forma și dimensiunile eșantioanelor : format adecvat pentru montură standard (cilindric 25, 40, 50 mm în diametru), format ce necesită monturi speciale (inclusiv de formă rectangulară), format fără montură.

Starea suprafeței la recepție : neprelucrată, tăiere grosolană, tăiere cu ferăstrău alternativ, tăiere cu disc abraziv, tăiere cu disc diamantat, prelucrat prin așchiere, rectificare de degroșare, tăiere cu fir, deformat plastic.

În cadrul modalităților de operare se iau în considerare caracteristicile și specificul funcționării laboratorului, dintre care cele mai importante sunt :

- *Scopul testării* : control de calitate, servicii pentru terți, cercetare – dezvoltare, testarea unui produs comercial, educațional, testare curentă în fluxul de producție;
- *Personal* – numărul de persoane care lucrează în cadrul laboratorului și calificarea acestora;
- *Tipul testării* : examinare vizuală, macrofotografie, microfotografie, microdurate, macrodurate, măsurarea mărimii grăunților, control dimensional, contorizare incluziuni, analiză nedistructivă.

2.2. Tehnologia pregătirii eșantioanelor metalografice.

Pentru a asigura serviciile cerute, laboratorul metalografic va trebui să fie organizat și echipat în mod adecvat și să dispună de un personal calificat [BUE1].

În figura 2.3 sunt prezentați factorii ce contribuie la definirea structurii unui laborator metalografic. În unele cazuri laboratorul este folosit doar pentru analiza cantitativă a microstructurii unui număr mic de eșantioane pentru care este necesar un spațiu mic și un număr restâns de echipamente și personal. Pe de altă parte însă cerințele unei întreprinderi mari impun examinarea unui număr mare de eșantioane și efectuarea unor analize complexe, caz în care sunt necesare echipamente complexe și

un personal cu înaltă calificare.

Buna funcționare a unui laborator metalografic este asigurată în primul rând de către personalul acestuia, care pregătește eșantioanele și execută analiza metalografică. Calitatea rezultatelor analizelor depind în bună măsură de calificarea și experiența personalului laboratorului.

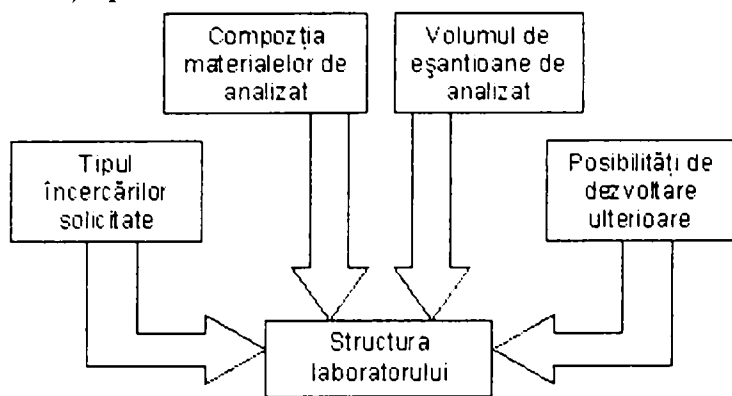


Fig.2.3. Factori ce determină structura laboratorului.

Pregătirea eșantioanelor metalografice constă în obținerea unor suprafețe finisate, fără deformații, care, supuse analizei să permită evidențierea microstructurii.

Echipamentele folosite în acest scop pot să varieze în complexitate și capacitate de prelucrare în funcție de tipul analizei, volumul eșantioanelor de prelucrat și materialul eșantionului.



Fig.2.4. Itinerarul operațiilor metalografice

Așa cum se arată în figura 2.4. operațiile ce pot fi executate într-un laborator metalografic de capacitate medie sunt următoarele :

- *debitare* – tăierea eșantioanelor la o mărime convenabilă ce se poate executa cu o sculă abrazivă convențională sau prin tăiere cu freză disc cu un regim de așchiere neintensiv.
- *rectificare (șlefuire) de degroșare* – realizată în scopul obținerii unei suprafețe plane, executată cu hârtie sau disc abraziv.
- *realizarea monturii* – pentru ca eșantionul să poată fi poziționat corespunzător, în unele cazuri, se presează într-un material moale (plastic) cu ajutorul unei prese. În acest scop se pot folosi și rășini sintetice. Montajul astfel obținut are un gabarit mai mare și este mai ușor de manipulat și de fixat. Prin realizarea montajului se previne și rotunjirea muchiilor eșantionului la prelucrările de finisare ulterioare.
- *rectificare (șlefuire) de finisare* - se realizează cu ajutorul hârtiilor abrazive cu diferite granulații montate pe frontalul unor discuri rotative. Folosirea hârtiilor abrazive este indicată deoarece prelucrarea se realizează la temperaturi relativ scăzute, evitându-se astfel posibile modificări ale microstructurii. În

unele cazuri se practică și lepuirea suprafețelor cu ajutorul unor paste abrazive.

- *atacul cu reactivi* – microstructura materialului nu este vizibilă în mod normal după finisare. Din această cauză suprafața eșantionului este supusă acțiunii unor reactivi chimici care atacă în mod diferențiat diferite componente ale microstructurii realizând un relief vizibil. De remarcat că unele tipuri de analiză cum ar fi evidențierea incluziunilor sau studiul suprafețelor de rupere nu necesită atac cu reactivi.

- *analiza macroscopică* – se realizează prin examinarea suprafețelor eșantioanelor cu ochiul liber sau cu ajutorul unor lupe. Pentru acest tip de analiză nu este necesar atacul eșantioanelor cu reactivi.

- *analiza microscopică* – se realizează cu ajutorul microscopelor optice și reprezintă de fapt scopul final al tuturor celorlalte operații.

În unele cazuri anumite operații prezentate în figura 2.4. pot să lipsească (rectificarea de degroșare sau realizarea monturii).

Experiența acumulată în domeniul pregătirii eșantioanelor metalografice a dus la stabilirea unor itinerarii tehnologice bine definite și la realizarea unor echipamente corespunzătoare.

O caracteristică importantă a acestor echipamente este aceea că ele sunt destinate unui scop precis : obținerea unei suprafețe plane superfinisate adecvate relevării microstructurii eșantionului. Diversitatea procedeeleor de prelucrare și atac cu reactivi, constă în modul diferit în care trebuie tratate materiale diferite. Astfel, gradul de finisare al suprafeței este mai mic în cazul unor teste simple cum ar fi determinarea mărimii incluziunilor nemetalice pe un anumit eșantion, și mai mare pentru determinarea procentului fazelor structurale în cazul unor structuri complexe.

Gabaritul eșantioanelor este relativ mic și se alege în așa fel încât suprafața de analizat să dispună de o arie de câțiva cm². Eșantioanele cu gabarit mic care sunt greu de manipulat și a căror prindere, în vederea prelucrării, este dificilă, sunt înglobate într-o montură cu ajutorul unor prese, sau într-o rășină epoxidică.

Acestă practică este avantajoasă și datorită faptului că astfel se evită rotunjirea muchiilor eșantionului, deseori o zonă importantă de analiză fiind chiar zona din apropierea muchiilor (cazul studierii straturilor superficiale).

2.3. Mașini și echipamente de pregătire a eșantioanelor.

Ca și în alte domenii industriale, echipamentele tehnologice folosite pentru prelucrarea eșantioanelor au trecut prin mai multe faze de dezvoltare ajungând în ultimul timp la variante ce conțin elemente de automatizare. Totuși construcția acestor echipamente, în cele mai multe cazuri, nu este de natură să facă posibilă integrarea în sisteme flexibile, fără o adecvare prealabilă, deoarece lipsesc elementele de comandă și interfață care să permită schimbul de informații la nivel de sistem.

Echipamentele de pregătire a eșantioanelor sunt unele dintre cele mai dezvoltate la ora actuală, marea lor majoritate fiind produse de către firma Buehler,

una dintre cele mai importante companii în domeniu. Alte firme cum ar fi Struer's, Leco sau Mecamet produc echipamente similare cu cele prezentate. În figura 2.5. se prezintă o clasificare a acestor aparate. Aparatele de șlefuit eșantioane metalografice sunt produse într-o gamă largă, acoperind cerințele diverselor laboratoare.

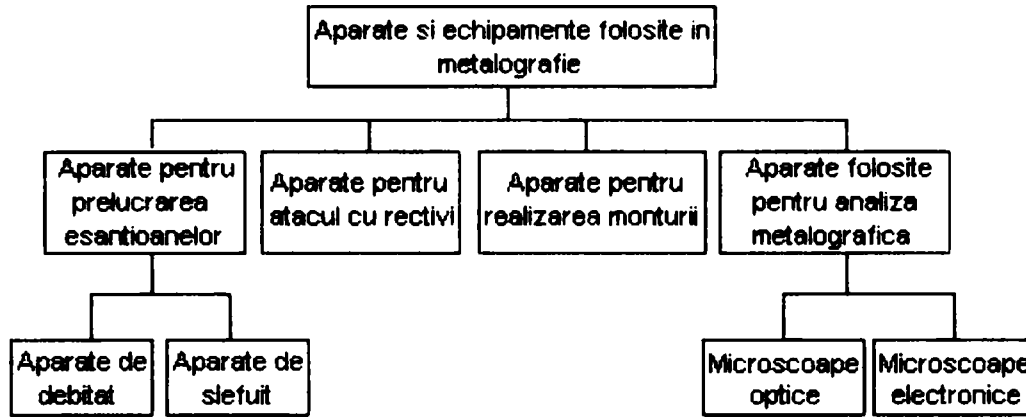


Fig. 2.5. Clasificarea aparatelor folosite în metalografie

Un aparat mai evoluat atât din punctul de vedere al automatizării cât și al integrării mai multor operații pe un singur echipament, este prezentat în figura 2.6.a.

Acest echipament automatizat îndeplinește funcțiunile unui aparat de șlefuit, a unui aparat de curățat (debavurat) cu ultrasunete și a unui uscător, toate operațiile realizându-se într-o succesiune automată. În figura 2.6.b. este prezentat un aparat pentru prelucrarea eșantioanelor de gabarit mărit ce folosește discuri cu un diametru de 305 respectiv 250 mm. Aparatul este dotat cu afișare digitală, sistemul de comandă fiind capabil să programeze diverse cicluri de lucru.

Aparatul prezentat în figura 2.6.c. este dotat cu discuri de diametre de 200 respectiv 250 mm. Viteza de rotație a discurilor este programabilă. Corpul aparatului este realizat din materiale minerale (granit) pentru a atenua vibrațiile. Funcționarea aparatelor se poate extinde prin montarea unor capete de fixare a eșantioanelor.

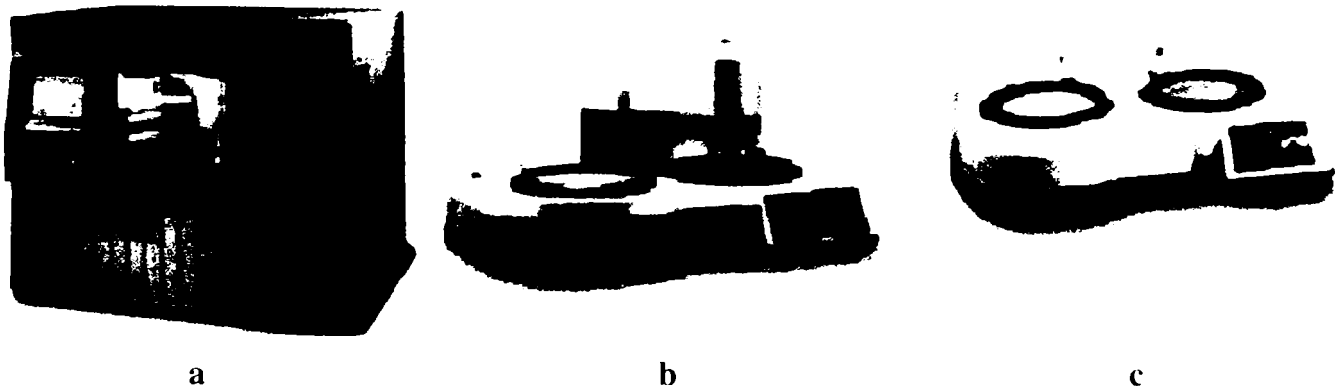


Fig.2.6.a. Aparatul BUEHLERVANGUARD®, b Aparatul PHOENIX 4000 c Aparatul ALPHA

Realizarea inserării eșantioanelor în monturi din materiale plastice este realizată cu ajutorul unor prese de mici dimensiuni. Operarea presei se realizează cu ajutorul unui panou de comandă cu afișare digitală a valorilor programate. Se pot programa

valori ale temperaturilor (între 150-180°C) și a presiunilor (80-300 bari).

Atacul cu reactivi este destinat punerii în evidență a microstructurii. Modul în care se realizează atacul cu reactivi influențează în mare măsură calitatea analizei metalografice. În laboratoarele convenționale atacul este realizat manual. Din punctul de vedere al realizării atacului acesta se poate efectua prin picurare (acoperirea suprafeței cu reactiv) sau prin imersie. Durata atacului este de asemenea un factor important în pregătirea unor eșantioane. Atacul clasic cu reactivi are ca rezultat corodarea diferențiată a unor componente ale microstructurii sau a marginii grăunților în comparație cu miezul acestora, lăsând în urma lui un relief, care prin iluminare relevă porțiuni luminoase și umbrite ce corespund într-o anumită măsură aranjării spațiale a componentelor microstructurii. Mecanismul de corodare diferențiată este explicat printr-un proces de electroliză naturală în care există diferențe de potențial între grăunții anumitor faze, între miezul și marginea grăunților, între impurități și matricea de bază. La eșantioanele ce conțin o singură fază pot exista diferențe de potențial între grăunți cu orientări cristaline diferite dar acestea sunt mai dificil de pus în evidență. Cu cât diferențele de potențial sunt mai mari cu atât atacul este mai pregnant, deci pentru diferențe mici de potențial durata atacului este mai mare. Ca reactivi sunt folosite soluțiile a diferiți acizi în apă sau în alcool. Natura reactivului trebuie să fie în concordanță cu microstructura atacată pentru ca punerea ei în evidență să fie corectă.

2.4. Analiza imaginilor microstructurilor.

2.4.1. Considerații generale.

Prin analiză metalografică se înțelege studiul structurii (macroscopice sau microscopice) unui eșantion prelevat dintr-un material metalic. În figura 2.9. este dată o clasificare a metodelor de analiză a eșantioanelor metalografice.

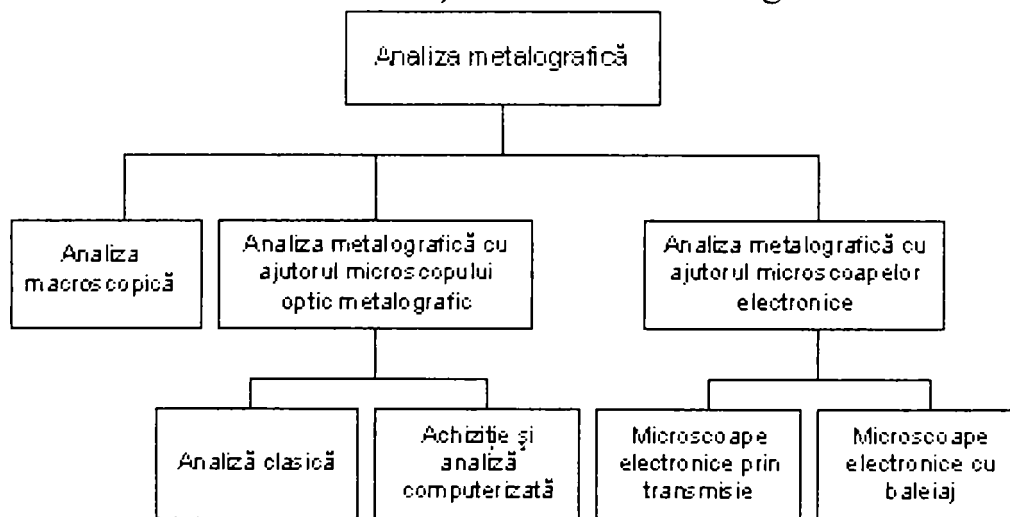


Fig.2.9. Clasificarea metodelor de analiză metalografică

Se definește ca analiză “clasică” acea analiză care se efectuează în totalitate de către un expert uman. Diferențele dintre metodele prezentate se referă în primul rând la măririle cu care se obțin imaginile, la modalitatea prin care se realizează aceste mărimi, cât și la modalitatea de preparare a eșantioanelor în vederea analizei [BUDAU].

2.4.2. Analiza macroscopică [BUDAU].

Acest tip de analiză are ca scop cercetarea structurii cu ochiul liber sau cu lupa, la o mărire de cel mult 50x. Pregătirea probei se realizează în mai multe etape după cum urmează : debitare, șlefuire, atac cu reactivi (dacă este cazul). Prin analiză macroscopică se realizează studiul suprafețelor de rupere sau a suprafețelor în diferite secțiuni ale piesei. În cazul studiului suprafețelor de rupere (fractologie) prin analiza macroscopică se primesc informații privind : dimensiunea grăunților cristalini, particularități ale proceselor de turnare sau deformare plastică (viteza și uniformitatea răcirii, temperatura de turnare) și tratamente termice aplicate.

2.4.3. Analiza microscopică [BUDAU].

Materialele metalice sunt corpuri opace, pentru analiza cărora se vor folosi microscopie optice în lumină reflectată.

Principiul de funcționare al unui microscop optic este arătat în figura 2.10.

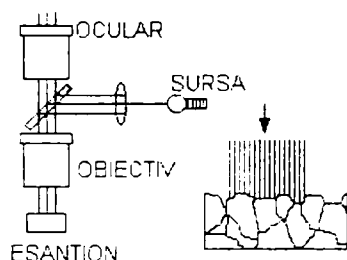


Fig.2.10. Principiul de funcționare al microscopului optic în lumină reflectată

Microscopul optic conține trei sisteme distincte : sistemul optic, sistemul de iluminare, sistemul mecanic.

Sistemul optic al microscopului este format din : obiective, prisme și oglinzi.

Obiectivele și ocularele sunt formate din lentile care în ansamblu se comportă ca un sistem pozitiv și convergent. În principiu, obiectivul se compune dintr-o lentilă frontală plan convexă (determină puterea de mărire) și o serie de lentile secundare (ajută la eliminarea defectelor lentilei frontale). Ocularele măresc imaginea reală dată de către obiectiv și corectează aberațiile acestuia. Ocularele se realizează în mai multe tipuri dintre care cele mai cunoscute sunt : *obișnuite* (de tip Huygens) – necorectate se folosesc pentru obiective acromatice; *de compensație - corectate* pentru obiective apocromatice; *de proiecție - corectate* – pentru obiective specifice fotomicrografierii.

Sistemul de iluminare al microscopului poate fi cu incandescență sau cu arc electric. Există mai multe tipuri de iluminări : în câmp luminos; în câmp întunecat; în

lumină polarizată; cu echivalente auxiliare.

Sistemul mecanic de reglare se compune din: stativ, tub vizual, ecran fotografic, măsută port obiect deplasabilă după două direcții perpendiculare în plan, șuruburi micrometrice.

În cele ce urmează sunt prezentate caracteristicile microscopului optic metalografic.

Apertura (deschiderea numerică) "A" sau puterea de strângere a razelor de lumină de către lentilă, care arată capacitatea maximă a obiectivelor de a reda cele mai mici detalii.

Puterea de mărire "M" este un parametru important al microscopului, definit prin produsul dintre mărirea obiectivului și cea a ocularului :

$$M = M_{ob} \times M_{oc} = \frac{L}{F_{ob}} \times \frac{l}{F_{oc}} = \frac{L \times 250}{F_{ob} \times F_{oc}}$$

unde : L – lungimea optică a tubului microscopului, $l = 250$ mm distanța vederii normale, F_{ob} – distanța focală a obiectivului, F_{oc} – distanța focală a ocularului.

În cadrul fotografierii se utilizează o relație de calcul aproximativ :

$$M_f = \frac{D \cdot M_{oc} \cdot M_{ob}}{250}$$

unde D este distanța de proiecție.

În cazul unui microscop comercial puterea de mărire este : cu ulei de cedru $M = 1000:1$; uscat $M = 750:1$.

Adâncimea de pătrundere (puterea de separare verticală), este capacitatea obiectivului de a reda clar imaginea unor puncte aflate în diferite plane. Ea este invers proporțională cu apertura, ceea ce înseamnă că la mărimi mari trebuie folosite atacuri slabe deoarece cu cât atacul cu reactivi este mai puternic diferența dintre planele unor faze diferite ale structurii este mai mare.

Microscoapele optice au un domeniu larg de utilizare în aplicații industriale și de cercetare, în evidențierea unor caracteristici, cum ar fi : incluziuni nemetalice în oțeluri; granulația structurală; natura, mărimea, forma și distribuția contituenților structurali. Eșantioanele pregătite în prealabil după metodele descrise în paragraful anterior, sunt amplasate în câmpul vizual al microscopului, după care se realizează analiza acestora. De regulă analiza este realizată de către un personal calificat, specialist metalograf [BUE]. Specialistul poate realiza și fotografii ale microstructurii dacă acestea se cer ca anexe la buletinele de analiză. În marea majoritate a cazurilor poziționarea eșantionului și reglarea distanței focale a microscopului se realizează manual.

2.5. Concluzii.

Deși cele mai multe dintre sistemele de analiză metalografică sunt de tip clasic, în ultimul deceniu, odată cu dezvoltarea elementelor de automatizare, se folosesc din ce în ce mai multe echipamente și aparate ce înglobează asemenea elemente, în scopul

ușurării muncii operatorului uman. Totuși în aceste sisteme factorul de decizie primordial rămâne omul. În figura 2.11. se arată stadiul actual al automatizării sistemelor de analiză metalografică. Se observă că majoritatea operațiilor de pregătire se realizează de către operatorul uman, neexistând echipamente sau aparate care să lucreze la un nivel de automatizare corespunzător unei celule flexibile [GAL1]. În ceea ce privește analiza propriu-zisă a microstructurii s-au realizat pași importanți în automatizarea acestora, dar operațiile arătate sunt permanent supervizate de către operatorul uman care participă la toate fazele de decizie.

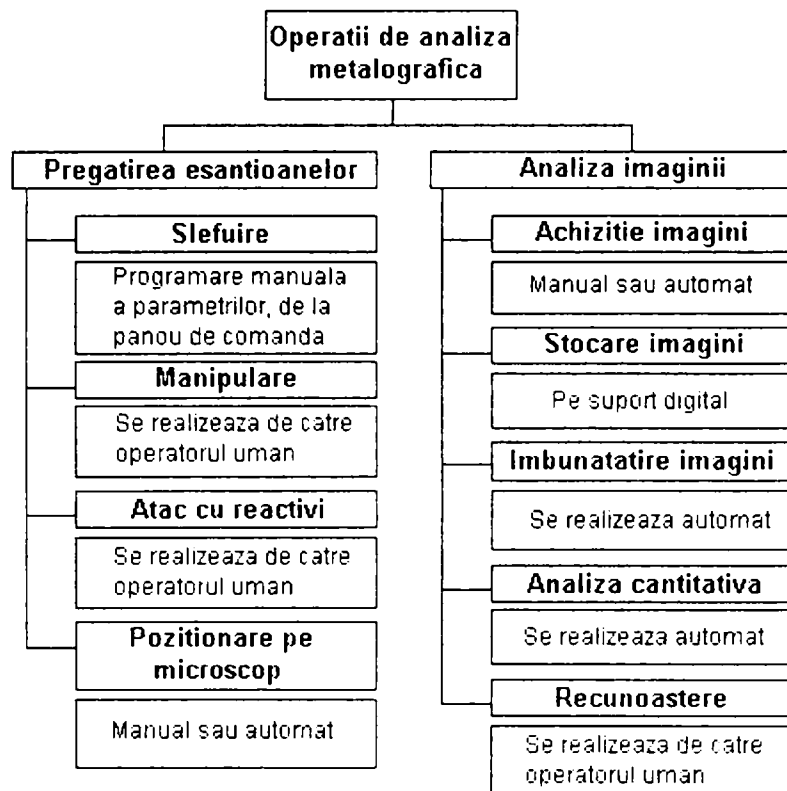


Fig.2.11. Stadiul actual al automatizării sistemelor de analiză metalografică.

În comparație cu echipamentele și mașinile de prelucrare din componența sistemelor flexibile, acestor aparate le lipsesc o serie de elemente ca: traductori de forță pentru monitorizarea apăsării eșantionului pe discul abraziv, traductori de viteză de rotație și avans, traductori de poziție; echipamente pentru preluarea semnalelor de la traductori și realizarea unor bucle de reglare; module de dialog (schimb de informații) cu alte aparate sau sisteme de calcul. Aceste componente sunt absolut necesare pentru realizarea unei automatizări flexibile a aparatelor de pregătire a eșantioanelor metalografice. În principiu, achiziția și prelucrarea imaginilor metalografice parcurge etape similare cu cele realizate în vederea oricărei aplicații de acest gen (recunoașterea formelor, conducere vizuală, recunoașterea caracterelor).

Particularități importante derivă din faptul că în acest caz se lucrează cu imagini achiziționate prin intermediul unui microscop și că aceste imagini nu reprezintă obiecte ci structuri abstracte (texturi).

În cadrul tezei de doctorat s-au abordat, pe lângă alte subiecte, două aspecte importante legate de pregătirea și analiza metalografică și anume modalitatea de

realizare a șlefuirii eșantioanelor, procesarea și interpretarea automată a imaginilor microstructurilor. Noțiunile de bază utilizate pentru rezolvarea acestor probleme sunt prezentate în capitolul 3.

3. Stadiul actual în robotizarea pregătirii și analizei metalografice.

3.1. Introducere.

Pregătirea eșantioanelor metalografice implică realizarea șlefuirii și a atacului chimic cu reactivi. Atacul chimic se poate realiza asistat de robot sau independent de robot, dar în ambele cazuri procesarea decurge fără a implica o solicitare dinamică majoră a robotului, acesta îndeplinind doar sarcini de manipulare (aducerea eșantionului la postul de atac chimic, menținerea eșantionului într-o anumită poziție pe timpul procesării). Procesările de spălare și uscare, cât și cele de verificare a calității suprafeței se pot realiza în mod similar cu atacul chimic cu reactiv.

În cazul șlefuirii apar însă solicitări dinamice ale robotului iar comanda apăsării eșantionului pe discul de șlefuit trebuie realizată într-o manieră care implică conducerea în forță – poziție. Acest tip de sarcină se încadrează în categoria sarcinilor de contact („contact task”). Pentru abordarea acestui tip de probleme, cu caracter dinamic, neliniar, se utilizează din ce în ce mai mult abordări bazate pe logica fuzzy și rețele neuronale artificiale.

A doua problemă abordată în cadrul tezei de doctorat este cea a procesării și interpretării computerizate a imaginilor metalografice. Există o mare diversitate de metode și algoritmi în domeniul procesării imaginilor în general, fiind necesar un studiu pentru stabilirea metodelor ce se pot utiliza în cazul metalografiei.

Pentru rezolvarea acestor probleme, în acest capitol se prezintă o sinteză a realizărilor în stadiul actual.

3.2. Elemente de comandă a roboților utilizând logica fuzzy și rețele neuronale artificiale.

3.2.1. Considerații generale.

Cele mai dificile probleme în ce privește comanda roboților sunt legate de complexitatea dinamică a acestora, datorată unor factori ca de exemplu posibilități reduse de cunoaștere precisă a parametrilor cinematici și inerțiali, flexibilitatea elementelor mecanice, dinamica neliniară a acționărilor, frecarea vâscoasă și uscată și nu în ultimul rând dinamica mediului.

Există strategii de comandă clasice (de exemplu cele care folosesc regulatoare PID), ce pot satisface un număr mare de aplicații, dar lărgirea domeniului aplicațiilor face ca problemele ce trebuiesc soluționate să fie din ce în ce mai complexe. După cum practica a demonstrat, pentru comanda proceselor ce au un puternic caracter neliniar, s-au impus cu precădere două categorii de strategii și anume cele bazate pe logica fuzzy și cele bazate pe rețele neuronale artificiale.

Elemente de cinematică folosite în conducerea roboților

Abordarea consacrată a elaborării modelelor matematice ale roboților este bine cunoscută fiind tratată în mai multe lucrări [KOVACS1], [DRAGULESCU].

Modelarea matematică a unui robot se poate face și sub forma dată în [LEVINE1], obținând expresii ușor de utilizat în cadrul noțiunilor specifice conducerii roboților.

În acest context se definește vectorul $q = (q_1, q_2, \dots, q_n)^T$ ca fiind o *configurație*. Mulțimea tuturor configurațiilor posibile se definește ca *spațiul coordonatelor cuplelor cinematice conducătoare*, și se notează cu T^n .

Mulțimea tuturor situațiilor pe care le poate atinge efectorul final se definește ca *spațiul situațiilor*. Coordonatele originii sistemului legat de efectorul final se definesc prin vectorul $p \in M^3$, iar matricea de orientare se notează cu R . Mulțimea tuturor matricilor de orientare se notează $SO(3)$. Ca urmare *spațiul situațiilor* se poate nota $SE(3) = M^3 \times SO(3)$.

Cu aceste notații problema cinematică directă este de a determina o funcție ce realizează corespondența :

$$X_0 = \begin{bmatrix} p(q) \\ R(q) \end{bmatrix} = f_0(q) : T^n \rightarrow SE(3) \quad (3.1)$$

iar problema cinematică inversă este de a determina inversa funcției f_0 .

Relația dintre componentele vitezei efectorului final și cele ale cuplelor cinematice se pot exprima prin :

$$V = \begin{pmatrix} v \\ \omega \end{pmatrix} = J_0(q)\dot{q} \quad (3.2)$$

în care $J_0(q)$ este Jacobianul structurii cinematice considerate, iar $V^T = (v^T, \omega^T)$, reprezintă vitezele liniare și unghiulare ale efectorului final.

Datorită complexității cinematice și dinamice, problema conducerii roboților se descompune în trei faze separate : planificarea mișcărilor, generarea traiectoriei și urmărirea traiectoriei.

Schema de conducere principală a unui robot este prezentată în figura 3.1.

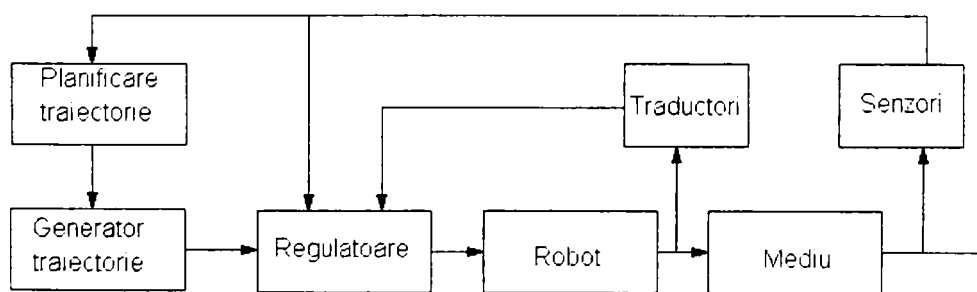


Fig.3.1.Schema de conducere a unui robot.

Elemente de dinamică folosite în conducerea roboților

Dinamica unui robot poate fi descrisă în mod convenabil folosind expresii energetice (energie cinetică, energie potențială) care reflectă caracteristici dinamice

globale. Noțiunea de coordonată generalizată utilizată în mecanica teoretică, poate fi folosită pentru reprezentarea coordonatelor cuplelor cinematice conducătoare.

Energia cinetică a unui manipulator, se exprimă sub forma :

$$K = \frac{1}{2} \sum_{i,j=1}^n d_{ij}(q) \dot{q}_i \dot{q}_j = \frac{1}{2} \dot{q}^T D(q) \dot{q} \quad (3.3)$$

în care $D(q)$ este matricea de inerție a robotului.

Dacă se definește $P: C \rightarrow R$, ca o funcție continuă și derivabilă, numită funcție a energiei potențiale, atunci această funcție va reprezenta energia potențială datorată numai gravitației, în cazul robotului rigid, și încă a energiei potențiale elastice, în cazul robotului flexibil. Se definește funcția $L = K - P$, numită Lagrangian, cu ajutorul căreia se poate scrie ecuația lui Lagrange ce descrie dinamica robotului :

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_k} - \frac{\partial L}{\partial q_k} = \tau_k; k = 1, \dots, n \quad (3.4)$$

în care τ_1, \dots, τ_n , reprezintă forțe generalizate.

Sub formă matricială, ecuația lui Lagrange se scrie :

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau \quad (3.5)$$

în care $g(q)$, reprezintă componenta matriceală datorată gravitației.

Pe mai departe, mărimile folosite în relațiile din acest paragraf sunt considerate matrici. Relația (3.5), descrie dinamica unui robot în concepția în care elementele acestuia sunt rigide. În afară de acestea reprezentarea poate să cuprindă și dinamica acționărilor, flexibilitatea cuplelor și a elementelor de legătură, frecarea, cât și zgomotele ce afectează semnalele date de către senzori și traductoare. Mai mult, în momentul în care manipulatorul intră în contact mecanic cu mediul, trebuie luată în considerare descrierea dinamică a mediului cât și a forțelor de cuplare. Cu cât se ține seama de o caracterizare mai precisă a fenomenelor dinamice, cu atât modelul matematic va fi mai apropiat de realitate dar în același timp mai complex. O complexitate mare implică legi de reglare complexe, calcule complicate și timp de execuție ridicat. Din această cauză cheia proiectării unui sistem de conducere pentru un robot este de a lua în considerare numai acele componente care sunt majore în comportamentul dinamic, adică sistemul de conducere să fie insensibil sau robust față de componentele ne semnificative [LEVINE].

În continuare se prezintă modul în care se modifică ecuația dinamică, pentru factorii amintiți.

Influența inerției acționărilor și a frecării.

Inerția acționărilor este reprezentată sub forma unei matrici diagonale $n \times n$:

$$I = \text{diag}(I_1 r_1^2, \dots, I_n r_n^2) \quad (3.6)$$

Cu această expresie se redefinieste matricea de inerție ca : $M(q) = D(q) + I$ iar ecuația dinamică (3.5), se modifică :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + f(q, \dot{q}) = \tau \quad (3.7)$$

în care termenul $f(q, \dot{q})$ reprezintă expresia matriceală a forțelor de frecare.

Influența mediului.

Dacă se notează cu F_e matricea forțelor ce provin din mediu și acționează asupra robotului, expresia momentelor, potrivit principiului lucrului mecanic virtual, va fi :

$$\tau_e = J^T(q)F_e \quad (3.8)$$

în care $J(q)$ este Jacobianul manipulatorului.

Ecuția (3.5) se poate scrie :

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + J^T(q)F_e = \tau \quad (3.9)$$

Influența forțelor de interacțiune cu mediul este luată în considerare în cadrul “conducerii cu reglare în forță” a roboților.

Influența dinamicii acționărilor

În cazul în care se consideră ca elemente de acționare motoare de curent continuu, ecuația de echilibru în tensiuni a elementului de acționare este :

$$L \frac{di}{dt} + Ri = V - K_b \dot{q} \quad (3.10)$$

unde cu L s-a notat impedanța motorului, cu I și V s-au notat curentul respectiv tensiunea iar R și K_b sunt rezistența și inducția, considerate ca vectori.

Cum momentul produs la arborele motorului electric este proporțional cu intensitatea, $\tau = K_m i$ din relațiile (3.5) și (3.10) se poate forma sistemul :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = K_m i \quad (3.11)$$

$$L \frac{di}{dt} + Ri = V - K_b \dot{q}$$

Influența flexibilității cuplelor

Flexibilitatea cuplei se datorează în primul rând arborelui motor și transmisiei.

Dacă se consideră cuplele ca fiind flexibile numărul gradelor de libertate se dublează deoarece se presupune că elementele de legătură dintre cuple se mișcă diferit față de arborele motorului de acționare, iar relația dintre aceste mișcări nu este corelată simplu numai prin raportul de transmisie. Acest lucru se reflectă și asupra calculelor de cinematică directă și inversă [SICILIANO1],[KARLIK1].

Dacă se notează cu q_1 deplasările unghiulare ale elementelor robotului și cu q_2 , deplasările unghiulare aferente arborilor motoarelor de acționare, și se consideră că între acestea există o legătură de tip elastic, ecuațiile dinamice ale manipulatorului se pot scrie :

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + K(q_1 - q_2) = 0 \quad (3.12)$$

$$I\ddot{q}_2 + K(q_1 - q_2) = \tau$$

în care I reprezintă inerția elementului de acționare iar K este matricea

diagonală a constantelor de elasticitate. Se observă în concluzie că ecuațiile dinamice ale roboților se pot scrie sub diverse forme în care sunt sau nu luați în considerare diferiți factori dinamici. Strategiile de conducere a roboților folosite în practică, în general, nu iau în considerare modele dinamice exhaustive, ci încearcă să rezolve problema conducerii pe baza unor modele simplificate. Decizia asupra modului în care se realizează simplificarea (care dintre factorii dinamici sunt importanți și care se pot neglija), este luată întotdeauna în urma analizei aplicației concrete (tipul robotului, mediul în care lucrează, ce sarcini trebuie să îndeplinească).

Strategii de conducere.

Metode ce folosesc regulatoare clasice.

În conducerea roboților un număr mare de aplicații folosesc sisteme de reglare clasice ca de exemplu PID (proporțional-derivativ-integrator), sau variante mai simple ale acestuia, ca de exemplu PD (proporțional-derivativ), după cum se arată în [LEVINE1],[ASTROM1].

În cazul unui regulator PD se consideră că în ecuația (3.7), se neglijează termenul frecării și ecuația dinamică devine :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = u \quad (3.13)$$

în care s-a notat $u = \tau - g(q)$.

Expresia legii reglării proporțional-derivative pentru cuple independente este:

$$u = K_p(q^d - q) - K_d\dot{q} \quad (3.14)$$

Expresia (3.14) este o lege de reglare neomogenă în raport cu cele două intrări : de tip P în raport cu referința q^d și de tip PD în raport cu reacția q iar K_p și K_d sunt matricile diagonale pozitive ale factorilor de amplificare proporționali respectiv derivativi. Se demonstrează folosind criteriul de stabilitate Lyapunov, că, cel puțin din punct de vedere teoretic, starea de echilibru $q = q^d : \dot{q} = 0$, este global asimptotic stabilă, dacă forțele de gravitație sunt compensate, adică $\tau = g(q) + u$. Există deasemenea încercări ce utilizează variante PID cu compensarea adaptivă a forțelor de gravitație pentru a nu fi necesară cunoașterea exactă a acestora.

Metodă bazată pe dinamica inversă (cu reglare după cuplu, "feedback linearization").

În cadrul acestui tip de strategie de reglare, se încearcă elaborarea unor legi care, în prima fază, să compenseze exact, toate cuplările neliniare în ecuația lui Lagrange (bucă internă), astfel încât în a doua fază algoritmul de reglare să acționeze asupra unui sistem liniar. Schema de principiu a sistemului este arătată în figura 3.2.

Pornind de la ecuația dinamicii manipulatorului dată de relația (3.7), omițând termenul ce exprimă frecarea, legea de reglare cu buclă neliniară (bucă internă), va avea expresia :

$$\tau = M(q)a_q + C(q, \dot{q})\dot{q} + g(q) \quad (3.15)$$

unde $a_q \in \mathbb{R}^n$, este necunoscută.

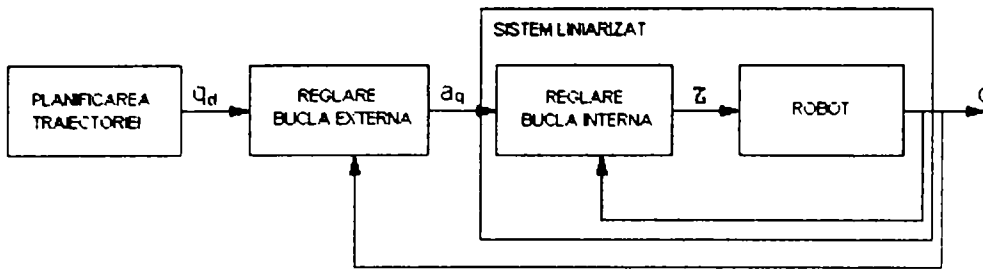


Fig. 3.2. Schema de principiu a unui sistem de conducere cu dinamică inversă.

Deoarece matricea de inerție $M(q)$ admite inversă, pentru orice q , sistemul în buclă închisă se reduce la, [LEVINE1]:

$$\ddot{q} = a_q \quad (3.16)$$

Pentru o traiectorie oarecare $q^d(t)$, pentru bucla de reglare externă, se alege o lege de reglare PD modificată:

$$a_q = \ddot{q}^d + K_p(q^d - q) + K_d(\dot{q}^d - \dot{q}) \quad (3.17)$$

Notând :

$$q_1 = q - q^d \quad (3.18)$$

se va obține sistemul cu buclă închisă, linear și decuplat dat de expresia:

$$\ddot{q}_1 + K_d\dot{q}_1 + K_p q_1 = 0 \quad (3.19)$$

Calculul termenilor neliniari ai buclei interne ar fi de preferat să se realizeze cu ajutorul unui procesor dedicat pentru a micșora durata efectuării calculelor.

Avantajele acestui tip de abordare rezidă în faptul că pentru bucla externă se pot adopta diferite legi de reglare fără a fi nevoie de modificarea buclei interne.

Metode robuste și adaptive

Dezavantajul metodei ce folosește dinamica inversă, constă în faptul că în legea de reglare a buclei interne nu pot fi luați în considerare o serie de factori : perturbații externe, zgomote, imprecizia calculelor și a modelării. Din această cauză sistemul nu va funcționa corect în anumite situații, când factorii perturbatori depășesc anumite limite. Metodele robuste și cele adaptive au ca scop asigurarea unei conduceri precise în prezența unor factori perturbatori semnificativi.

Diferența între cele două metode constă în faptul că în timp ce în sistemele de conducere adaptivă sunt înglobați algoritmi de estimare a parametrilor în timp real, în conducerea robustă nu se ține cont de modificările acestora.

Conducere robustă

Fiind dată ecuația dinamică :

$$M(q)a_q + C(q, \dot{q})\dot{q} + g(q) = \tau \quad (3.20)$$

intrarea de comandă se alege de forma :

$$\tau = \hat{M}(q)a_q + \hat{C}(q, \dot{q})\dot{q} + \hat{g}(q) \quad (3.21)$$

în care termenii notați cu “ ^ ” reprezintă valori calculate (nominale), și indică faptul că acestea sunt valori ideale pentru dinamica inversă, ce nu pot fi afectate de incertitudinile din sistem. Parametrul de reglare a_q poate fi folosit pentru compensarea perturbațiilor sistemului, în bucla de reglare exterioară. Dacă se definește :

$$a_q = \ddot{q}^d + K_d(\dot{q}^d - \dot{q}) + K_p(q^d - q) + \delta a \quad (3.22)$$

utilizând relațiile (3.18) și (3.22), se obține :

$$\ddot{q}_1 + K_d\dot{q}_1 + K_p q_1 = \delta a + \eta(q, \dot{q}, \delta a, t) \quad (3.23)$$

în care δa și η sunt funcții ce reprezintă incertitudinile din sistem.

Strategia de conducere în acest context se bazează pe determinarea unei toleranțe admise $\rho(q, \dot{q}, t) \geq 0$, astfel încât :

$$\|\eta\| \leq \rho(q, \dot{q}, t) \quad (3.24)$$

În consecință sistemul va fi insensibil la perturbații ce nu depășesc toleranța admisă.

Conducerea adaptivă

În definirea conducerii adaptive se consideră ecuațiile (3.21) și (3.22) ca și la conducerea robustă dar termenii din ecuația de reglare (3.23) nu sunt considerați fixați ca și în cazul anterior, ci sunt estimatele temporale ale parametrilor. Deci conducerea adaptivă se bazează, de fapt, pe estimarea mărimii de comandă, ținând cont în permanență de modificarea parametrilor sistemului.

Metoda conducerii prin optimizarea timpului de parcurgere a traiectoriei.

Se folosește în cazul în care există o corelație directă între viteza de execuție a mișcării robotului și ciclul procesului tehnologic. În principiu metoda constă în generarea unor traiectorii care să fie parcurse în timp minim de către efectorul final

Problema minimizării timpului înseamnă de fapt maximizarea vitezei, dar în mod evident vor apare limitări date de capacitatea acționărilor de a produce moment. Constrângerile impuse determină anumite puncte de schimbare a accelerației care poate lua ori valori maxime ori minime. Rezolvarea problemei se analizează în spațiul s, \dot{s} , în care s reprezintă coordonata curbilinie de-a lungul traiectoriei considerate.

Metoda de conducere prin folosirea informațiilor ciclurilor anterioare (repetitive and learning control).

În multe aplicații roboții execută sarcini repetitive. Se pornește cu o lege de reglare oarecare, aceasta poate fi îmbunătățită pe parcurs după realizarea mai multor cicluri. Dacă la fiecare ciclu se măsoară mărimile realizate și se compară cu cele dorite se găsesc valorile erorilor acestor mărimi. Completând legea de reglare cu

termeni de corecție care să minimizeze erorile pe măsură ce se realizează din ce în ce mai multe cicluri, legea de reglare se va îmbunătăți treptat ajungând să producă valori de comandă optime.

Conducere cu reacție în forță [LEVINE1],[BURDET1],[BAPTISTA1].

În cele prezentate până acum stabilirea erorii de conducere se baza cu precădere pe reacția în poziție a cuplelor cinematice conducătoare și compararea acestor mărimi cu cele de comandă. Într-un domeniu larg de aplicații se întâlnesc situații în care efortul final al robotului intră în contact cu mediul : asamblare, debavurare, polizare, șlefuire. În aceste cazuri, nu este oportun să se ia în considerare doar informațiile asupra poziției reale (position feedback) a robotului, ci devin importante și informațiile cu privire la forțe (force feedback).

Sistemele cu reacție în forță pot fi împărțite în două categorii : pasive și active. Sistemele cu reacție în forță pasive (compliance pasivă), nu sunt echipate cu senzori speciali pentru măsurarea forțelor și nici algoritmul de conducere nu este altul decât cel pentru reacție în poziție ci se bazează pe compliance elementelor constructive (acționări, elemente, cuple) sau a unor elemente compliant special concepute montate aproape de efortul final. Deși sistemele “pasive” au unele avantaje (costuri scăzute, rapiditate, robustețe), acestora le lipsește flexibilitatea (un anumit tip de element compliant este proiectat pentru un anumit tip de aplicații). Pentru a rezolva problema adaptării la situații diferite s-au găsit soluții de conducere cu reacție în forță, active. Acestea presupun echiparea robotului cu senzori de forță și un algoritm de conducere adecvat, dar și specificarea modului în care robotul să reacționeze la constrângerile externe.

Modelarea mediului în cadrul definirii unei conduceri cu reacție în forță se poate realiza considerând mediul rigid sau deformabil (soft environment). Mediul rigid este mai ușor de modelat, deoarece nu intervin deformații ale acestuia. Mediul compliant se poate lua în considerare dacă se definește interacțiunea robot-mediul, printr-o ecuație dinamică de forma :

$$F_c = M_c \Delta a + C_c \Delta V + K_c \Delta X \quad (3.25)$$

în care : - $\Delta V = V - V_e$ - viteza carteziană a deformației mediului;

- $\Delta a = \frac{d\Delta V}{dt}$ - accelerația deformației;

- $\Delta X = \int d\Delta V dt$ - deformația.

Un subiect de analiză mult mai profundă îl constituie sistemele cu reacție în forță active. Există două strategii distincte de abordare a problemei conducerii cu reacție în forță [BAPTISTA1].

Conducere hibridă forță – poziție (sau forță – viteză) (Raibert și Craig, 1981). Conducerea hibridă separă spațiul de conducere într-un subspațiu condus în forță (force-controlled), și un subspațiu condus în viteză (velocity-controlled).

Se demonstrează în [LEVINE1], că separarea spațiilor de conducere este

posibilă, pornind de la ecuația dinamică :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + J^T F_c = \tau \quad (3.26)$$

în care $F_c^T = (f^T m^T)$ este un vector cu 6 componente reprezentând forțe (f) și momente (m) ce apar în cazul contactului robotului cu mediul, J este matricea Jacobian a robotului

Notând cu V vectorul vitezelor robotului în spațiul cartezian, se ajunge la relația de reciprocitate cinematică :

$$V^T F_c = F_c^T V = 0 \quad (3.27)$$

Dacă se notează cu F_c forțele de reacție carteziene aparținând unui spațiu vectorial n_f dimensional, și se definește o bază a spațiului vectorial S_f se poate scrie :

$$\forall F_c, \exists \Phi = (\Phi_1, \dots, \Phi_{n_f}) \in R^{n_f} : F_c = S_f \Phi \quad (3.28)$$

în care Φ este un vector de scalari, F_c va fi subspațiul condus în forță.

Subspațiul condus în viteză va avea dimensiunea n_x cu proprietatea $n_x = 6 - n_f$ (pentru un robot cu 6 grade de mobilitate), și poate fi definit cu o bază S_x :

$$\forall V, \exists \chi = (\chi_1, \dots, \chi_{n_x}) \in R^{n_x} : V = S_x \chi \quad (3.29)$$

în care χ este un vector de scalari, iar V va fi subspațiul condus în viteze.

Specificarea referințelor se poate realiza astfel în două subspații separate, simplificând problema conducerii în forță a robotului. Dacă se notează cu F^d referințele momentane ale forțelor și cu V^d referințele momentane ale vitezelor, descrierea lor se poate realiza simplu prin specificarea vectorilor Φ^d și χ^d :

$$F^d = S_f \Phi^d, \quad V^d = S_x \chi^d \quad (3.30)$$

Dacă se notează cu S matricea de separare a subspațiului de forțe care se formează prin atribuirea de valori I pentru subspațiul de forțe și 0 pentru subspațiul de poziții, atunci matricea S se poate folosi pentru a reprezenta direcțiile reglate în forță iar $I - S$ (în care I este matricea identitate), reprezintă direcțiile reglate în poziție.

Schema de principiu a conducerii hibride forță – poziție este dată în figura 3.3. Notățiile din figura 3.3. au următoarea semnificație : f^d, x^d – forța și poziția de referință; q – poziția în coordonate c.c.c.; x, \dot{x} – viteza și poziția în coordonate carteziene.

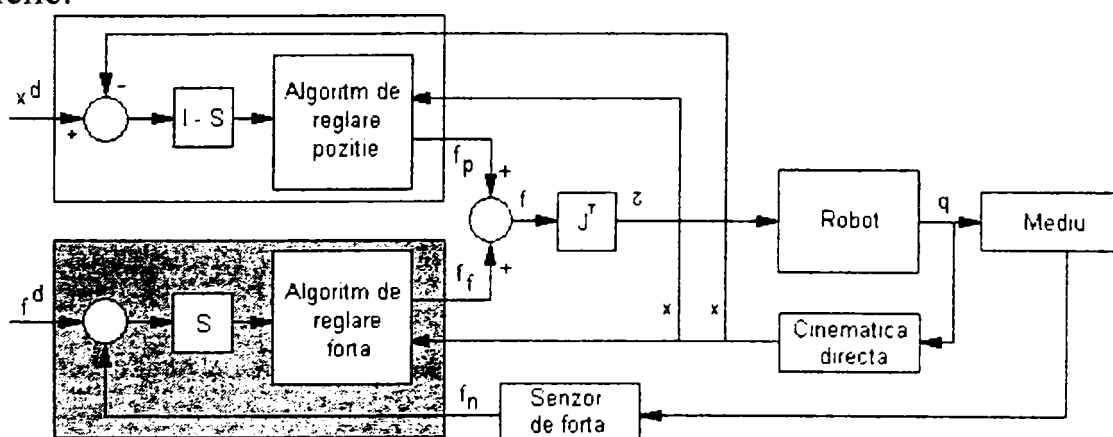


Fig. 3.3. Schema conducerii hibride forță - poziție

Conducerea prin impedanță (impedance control) (Hogan, 1985).

În acest caz nu este reglată explicit forța ci se specifică o relație între forță și deplasare a EF în contact cu mediul [BAPTISTA1]. Când contactul dintre EF și mediu este realizat, se presupune că mediul reacționează la acțiunea robotului, similar unui resort cu caracteristică liniară, k_e , și forța normală se poate scrie ca :

$$f_n = k_e \delta x \quad (3.31)$$

Ecuția de echilibru a contactului EF-mediul este :

$$k^d (x^d - x) = k_e (x - x_e) \quad (3.32)$$

în care indicele superior d reprezintă mărimi de referință și indicele e reprezintă mărimi ce corespund mediului.

Din relația (3.32) se poate exprima poziția x_{echil} și forța de echilibru f_{echil} (Singh și Popa, 1995):

$$x_{echil} = \frac{k^d x^d + k_e x_e}{k^d + k_e}; f_{echil} = \frac{k^d k_e}{k^d + k_e} (x^d - x_e) \quad (3.33)$$

Impedanța dorită se realizează prin calculul unei referințe de poziție virtuale :

$$x_v = x_e + f^d \frac{k^d + k_e}{k^d k_e}; f_n \neq 0 \quad (3.34.)$$

În funcție de valoarea lui f_n , x_v se poate exprima prin :

$$x_v = \begin{cases} x_e + \frac{f^d}{k^d}; f_n = 0 \\ x_e + f^d \frac{k^d \delta x + f_n}{k^d f_n}; f_n \neq 0 \end{cases} \quad (3.35)$$

Reglarea se va realiza în final în funcție de poziția de referință x_r :

$$x_r = \begin{cases} x^d, \text{ direcție } \parallel \text{ la suprafața de contact} \\ x_v, \text{ direcție } \perp \text{ la suprafața de contact} \end{cases} \quad (3.36)$$

Schema unui sistem cu conducere în impedanță este dată în figura 3.4.

Strategiile de control prezentate întâlnesc mari greutăți în implementare datorită volumului mare de calcule, a incompletitudinii modelelor folosite sau a factorilor de incertitudine ce caracterizează mediul în care lucrează robotul.

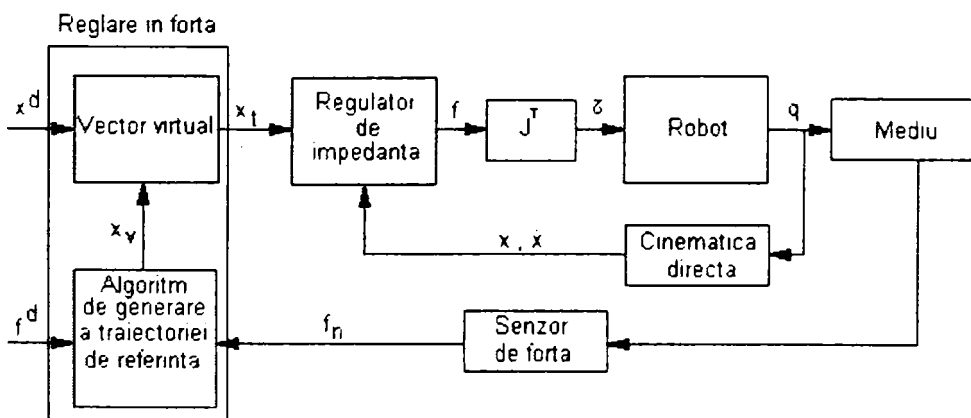


Fig.3.4. Schema conducerii în impedanță.

Din aceste considerente cercetările privind dezvoltarea strategiilor de

conducere bazate pe logica fuzzy și pe rețele neuronale artificiale sunt de o importanță deosebită.

3.2.2. Noțiuni privind logica fuzzy

Noțiunea de *mulțime vagă* a fost introdusă în matematică și teoria sistemelor de Lofti A. Zadeh în 1965 sub denumirea de *mulțime "fuzzy"*, (în traducere mulțime vagă). Teoria mulțimilor fuzzy și în general conceptele fuzzy au apărut din necesitatea de a exprima cantitativ mărimi imprecise. Se descriu în continuare noțiunile de bază privind teoria mulțimilor fuzzy ce vor fi utilizate în următoarele capitole ale tezei de doctorat.

Dacă X este o mulțime oarecare, se numește *mulțime fuzzy* (în X) rezultatul unei aplicații

$$F : X \rightarrow [0,1]. \quad (3.37)$$

Mulțimea fuzzy F este caracterizată de *funcția de apartenență*:

$$m_F : X \rightarrow P[0,1] \quad (3.38)$$

Valorile 0 și 1 reprezintă cel mai mic și respectiv cel mai mare grad de apartenență la F al unui element $x \in X$.

Pentru descrierea fuzzy a unor fenomene și procese, aplicațiile $m_F(x)$ pot admite diferite exprimări analitice. De exemplu funcția de apartenență triunghiulară este definită pe intervalul $[a, b]$, mijlocul acestui interval fiind $c = (a + b)/2$ iar expresia analitică a acestei funcții de apartenență, este dată de :

$$m(x) = 1 - 2 \frac{|x - c|}{b - a} \quad (3.39)$$

Operațiile uzuale din teoria clasică a mulțimilor se pot redefini în cazul mulțimilor fuzzy în termenii funcției de apartenență:

- *mulțimea vidă* $\emptyset \subseteq X$ este caracterizată de:

$$m_{\emptyset}(x) = 0; \quad x \in X \quad (3.40)$$

- *mulțimea totală* X de:

$$m_X(x) = 1, \quad x \in X \quad (3.41)$$

- două mulțimi fuzzy sunt *egale*, dacă funcțiile lor de apartenență sunt identice :

$$M = N \Leftrightarrow m_M = m_N; \quad (3.42)$$

- mulțimea fuzzy M este *conținută* în mulțimea fuzzy N , adică:

$$M \subseteq N \Leftrightarrow m_M \leq m_N \quad (3.43)$$

- între mulțimile fuzzy M și N se pot defini operațiile:

- *reuniune*:

$$M \cup N, \text{ cu } m_{M \cup N}(x) = m_M(x) \vee m_N(x), \quad x \in X \quad (3.44)$$

- *intersecție*:

$$M \cap N, \text{ cu } m_{M \cap N}(x) = m_M(x) \wedge m_N(x), \quad x \in X \quad (3.45)$$

- *complementara*

$$C_M, \text{ cu } m_{CM}(x) = m(x), x \in X. \quad (3.46)$$

- *produsul algebric* al mulțimilor fuzzy M și N , notat $M \bullet N$, este caracterizat de funcția de apartenență

$$m_{M \bullet N} = m_M \bullet m_N \quad (3.47)$$

- *suma algebrică* a mulțimilor fuzzy M și N , notată $M + N$, este caracterizată de funcția de apartenență

$$m_{M+N} = m_M + m_N - m_M \bullet m_N \quad (3.48)$$

Operațiile $\bullet, +$ sunt asociative, comutative, dar nu sunt distributive.

Presupunând că V_1, V_2, \dots, V_n sunt variabile logice, în logica fuzzy acestea iau valori în intervalul $[0,1]$.

Orice variabilă V_i este o *formulă fuzzy*. Dacă P, Q, \dots sunt formule în logica fuzzy, valorile logice (de adevăr) ale compuselor $P \vee Q, P \wedge Q, \bar{P}$ se stabilesc astfel:

$$A(P \vee Q) = \max(A(P), A(Q)) \quad (3.49)$$

$$A(P \wedge Q) = \min(A(P), A(Q)) \quad (3.50)$$

$$A(\bar{P}) = 1 - A(P) \quad (3.51)$$

$$A(P \rightarrow Q) = \min(1 - A(P) + A(Q), 1) \quad (3.52)$$

Logica fuzzy admite existența unor elemente particulare cu privire la *variabilele lingvistice*: relația de *implicație fuzzy* și noțiunea de *inferență fuzzy*.

Echivalentul valorii scalare în sens determinist este pentru o variabilă fuzzy, *termenul lingvistic* [ȘOFRON, PREITL] asociat acesteia. Domeniul de valori ale mărimii deterministe corespunzătoare se numește *univers de discurs*.

Fiecărui atribut al unei variabile lingvistice îi este asociată o funcție de apartenență, a cărei valoare (în sens determinist) indică nivelul de încredere cu care unei valori deterministe i se poate asocia acel atribut al variabilei lingvistice.

În logica fuzzy, implicația este o operație de compunere a formulărilor (variabilelor) fuzzy, în sensul corelării a două categorii de evenimente, denumite *premise*, respectiv *consecințe*. Rezultatul unei implicații fuzzy este, de asemenea, o submulțime fuzzy notată cu:

$$Q' \equiv P \rightarrow Q \quad (3.53)$$

care are aceeași termeni lingvistici ca și Q , dar funcțiile ei de apartenență, ce exprimă gradul de adevăr :

$$A(Q') = A(P \rightarrow Q) \quad (3.54)$$

rezultă în urma unor calcule algebrice efectuate asupra valorilor funcțiilor de apartenență corespunzătoare termenilor lingvistici ce compun implicația fuzzy.

Considerând formulele fuzzy:

$$\begin{aligned} P: x \text{ este MARE,} \\ Q: y \text{ este MIC,} \end{aligned} \quad (3.55)$$

unde x , respectiv y , reprezintă variabila deterministă aparținând universului de discurs al submulțimii P , respectiv Q , se exprimă implicația fuzzy:

$$Q \equiv P \rightarrow Q \Leftrightarrow \text{DACĂ } x \text{ este MARE, ATUNCI } y \text{ este MIC} \quad (3.56)$$

Combinarea mai multor variabile fuzzy, în conformitate cu o anumită logică, conduce la apariția unor expresii fuzzy cu mai mulți termeni, legați prin operații logice elementare (*ȘI*, *SAU*, *NU*). Gradele de apartenență fuzzy pot fi combinate cu ajutorul *operatorilor de compunere fuzzy* :

$$m_A \text{ ȘI } m_B = \min(m_A, m_B) \quad (3.57)$$

$$m_A \text{ SAU } m_B = \max(m_A, m_B) \quad (3.58)$$

$$NU \ m_A = 1 - m_A. \quad (3.59)$$

O altă serie de metode sunt folosite pentru obținerea unor valori ferme din valori fuzzy, Acestea poartă numele de metode de defuzzificare. Cele mai folosite metode de defuzzificare sunt [PREITL1]:

- *metoda eșantionului maxim* - în care din toate regulile de activare se selectează regula cu gradul de realizare maxim, ce va determina prin valoarea funcției de activare de ieșire valoarea fermă;

- *metoda centrului de greutate* – prin calculul centrului de greutate al ariei funcției de apartenență.

În cadrul tezei de doctorat logica fuzzy s-a utilizat în realizarea unui regulator utilizat la sistemul de comandă al șlefuirii eșantioanelor metalografice, la un algoritm de calcul al priorității eșantioanelor pentru procesarea în cadrul celulei flexibile și în cadrul algoritmilor de recunoaștere a imaginilor.

3.2.3. Stadiul actual al cercetărilor în domeniul rețele neuronale artificiale.

O rețea neuronală artificială poate fi definită ca un sistem informațional compus din „neuroni” de același tip organizați într-o topologie structurată, dotată cu algoritm de învățare și care este capabil să redea informațiile învățate. Noțiuni de bază ale rețelelor neuronale sunt date în [DUMIT1] și [TODER1] iar aplicațiile acestora în inginerie și conducerea proceselor sunt date în [LEVINE1], [REICH1] și [REICH2].

Funcționarea unei rețele neuronale poate fi defalcată în două faze separate: faza de antrenare sau învățare și faza de procesare a informațiilor pe baza celor învățate

Din definiția rețelelor neuronale rezultă că acestea pot fi privite ca memorii speciale în care se înscriu date prin învățare și se recitesc apoi în timpul funcționării. În ceea ce privește modul de implementare al rețelei neuronale, aceasta poate funcționa în impulsuri (în timp discret) sau continuu, în funcție de sistemul hard digital, respectiv analogic pe care este implementată rețeaua neuronală. Tot de sistemul hard depinde și modul de funcționare paralel (în cazul sistemelor multi-procesor) sau succesiv (în cazul sistemelor cu un singur procesor). În al doilea caz se poate vorbi doar de simularea funcționării rețelei neuronale, pentru că modul de funcționare natural (intrinsec) este cel paralel.

Structura generală a unui element de bază al unei rețele neuronale (denumit neuron artificial), este arătată în schema din figura 3.5.

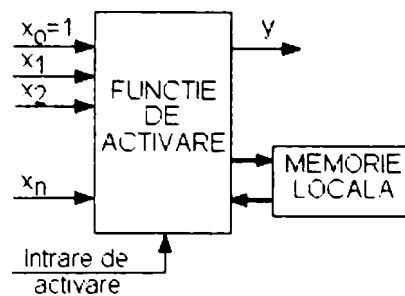


Fig.3.5. Schema de principiu a unui neuron artificial.

În figura 3.5. *intrarea de activare*, există numai în cazul rețelelor cu funcționare discretă, această intrare are scopul de a da tactul rețelei.

Expresia analitică ce leagă ieșirea de intrările neuronului este de forma :

$$y(k) = f(x(k), x(k-1), \dots, x(k-Mx), y(k-1), y(k-2), \dots, y(k-My)) \quad (3.60)$$

în care : $x(k) = [x_0(k), x_1(k), \dots, x_N(k)]^T$

În decursul dezvoltării rețelelor neuronale s-au definit mai multe tipuri de structuri, dintre care se prezintă cele mai semnificative.

Schema unui neuron simplu (perceptron), fără memorie locală, este prezentată în figura 3.6. Aceasta este baza din care s-au dezvoltat toate celelalte tipuri de rețele neuronale.

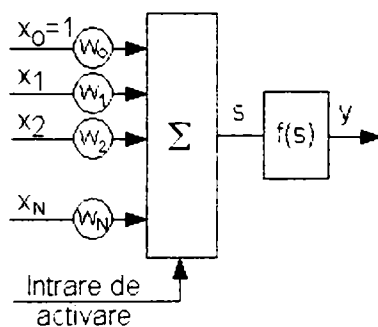


Fig.3.6. Schema de principiu a unui perceptron.

Expresia ieșirii perceptronului se calculează prin suma ponderată $s = \sum_{i=0}^N w_i x_i$ și apoi rezultatul acesteia este $y = f(s)$, în care $f(s)$ este o funcție neliniară. Suma ponderată s poartă denumirea de excitație iar ieșirea y este răspunsul la această excitație. Sunt cazuri în care : $f(s) = s$, atunci: $y = \sum_{i=0}^N w_i x_i = w^T x$ și $y = s = w^T x$. Acest tip de neuron, intră de obicei în componența rețelelor mai mari.

Perceptronul prezentat în figura 3.6. a fost definit de Rosenblat (1958). Când ieșirea de activare $f(s)$ este o funcție neliniară, neuronul prezentat în figura 3.6. se numește ADALINE (ADAptive LInear NEuron - Widrow și Hoff, 1960). Funcția neliniară de activare este de obicei o funcție de tipul : $y = \text{sign}(s); s \in R \rightarrow [-d, e]$. În afară de aceasta se mai folosește tipul de funcții de activare $y = \frac{1 - e^{-ks}}{1 + e^{-ks}}; k > 0$.

De multe ori intrările se clasifică după faptul că întăresc (măresc) sau slăbesc (micșorează) ieșirea, deci intrările pot fi: excitatoare sau inhibatoare.

Schema neuronului dotat și cu intrări excitatoare sau inhibatoare este prezentată

în figura 3.7.a. (Fukushima 1988) și se folosește cu succes în prelucrarea imaginilor.

În cazul acestui tip de neuron, funcția f_1 este definită prin relația:

$$f_1(x_1, \dots, x_M) = \frac{1 + \sum_{i=1}^N w_i x_i}{1 + w_M x_M}, \text{ iar funcția } f_2 \text{ are forma prezentată în figura 3.7.b.}$$

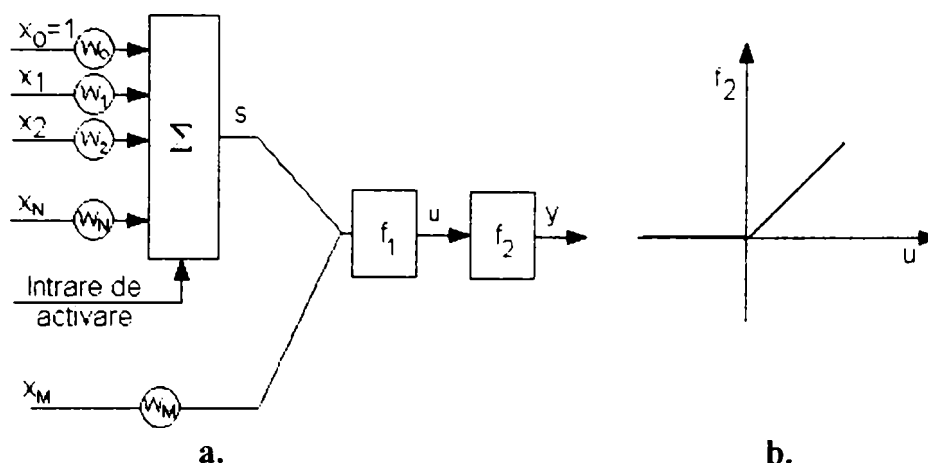


Fig.3.7. Schema neuronului dotat cu intrări de excitație și inhibitoare (a), reprezentarea grafică a funcției $f_2(u)$ (b).

O altă variantă a neuronului este cea reprezentată în figura 3.8. în care elementul sumator lipsește și cele N intrări ale neuronului reprezintă de fapt variabile ale unei funcții de tip distanță sau normă $(x_1, x_2, \dots, x_N) \rightarrow u; (x_1, x_2, \dots, x_N) \in R^N; u \in R$, având expresia : $u = \|x, -c\|$, care în particular poate fi scrisă ca $u = \sqrt{(x_1 - c_1)^2 + \dots + (x_N - c_N)^2}$. Ieșirea este o funcție a distanței dintre vectorul intrărilor $x = [x_1, x_2, \dots, x_N]^T$ și vectorul „c” specific neuronului $c = [c_1, c_2, \dots, c_N]^T$.

În cazul folosirii unei funcții de distribuție normală cu rol de funcție de activare, $f(u)$ are o simetrie radială în jurul mediei c , iar σ este dispersia acestei funcții.

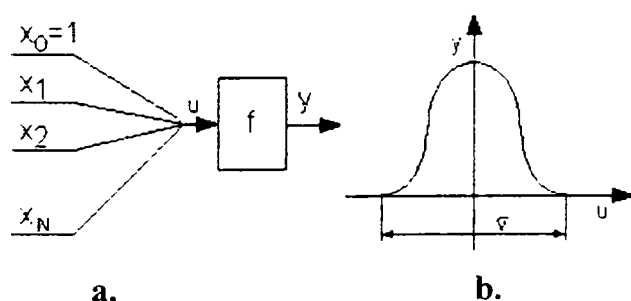


Fig.3.8. Schema unui neuron RBF (a) și reprezentarea grafică a funcției de activare (funcție de distribuție normală) (b).

Rețelele neuronale compuse cu astfel de neuroni se numesc de tip *RBF* (Radial Basis Function).

Pentru a întări rolul valorilor intrărilor învățate neuronii se dotează cu memorii locale. În acest scop, de obicei, se introduce în schema generală un element de tip filtru linear *FIR* (Finite Impulse Response) sau *IIR* (Infinite Impulse Response) la ieșirea sau la intrarea neuronului.

Funcția de memorie a neuronului se realizează în componenta de memorie a filtrului. În figura 3.9. se prezintă schema unui neuron cu filtru pe ieșire, iar în figura

3.10. o rețea cu M neuroni cu filtre pe intrări și un filtru pe ieșire.

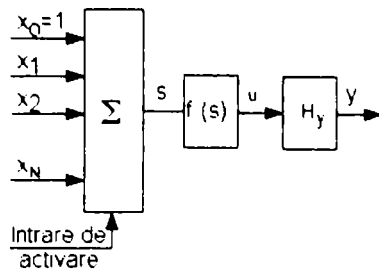


Fig.3.9. Schema unui neuron cu filtru pe ieșire.

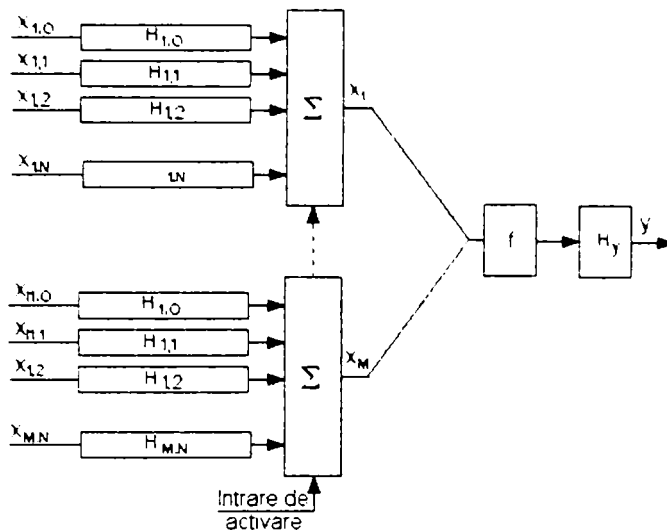


Fig.3.10. Schema de principiu a unei rețele cu memorii (filtre) pe intrări și ieșire.

Ponderile rețelei neuronale se ajustează prin învățare în așa fel ca la o intrare dată, ieșirea generată de rețea să fie cât mai apropiată de ieșirea dorită. În unele cazuri în timpul procesului de învățare se poate modifica și structura rețelei, nu numai parametrii acesteia.

După modul în care se realizează învățarea, pot exista următoarele situații:

- învățare supravegheată (sau dirijată), în care ieșirile dorite sunt impuse de proiectantul rețelei, (exemplu : aproximarea funcțiilor, realizarea unui model de proces) (supervised learning);
- învățarea nesupravegheată, în care nu sunt specificate explicit ieșirile dorite, ci ieșirile sunt clasificate de către rețea pe baza unor norme sau distanțe definite intrinsec în structura rețelei (clasificări ale unor seturi de date - clustering) (unsupervised learning).

Modul în care se realizează învățarea este hotărâtor pentru succesul aplicării acesteia în practică. Stabilirea structurii rețelei se poate realiza cu puține cunoștințe despre procesul practic, dar contrar cu aceasta, învățarea este mult mai dependentă de natura aplicației.

3.3. Procesarea computerizată a imaginilor.

3.3.1. Considerații generale.

Într-un articol publicat încă în 1996 [FOSTER1], s-a subliniat impactul pe care îl are dezvoltarea tehnicii digitale (computere, camere video) în metalogarfie. De

atunci această tehnică este într-o continuă ascensiune, oferind cercetătorilor unele din ce în ce mai performante.

În continuare sunt prezentate pe de o parte noțiunile de bază în ceea ce privește structurarea informațiilor ce alcătuiesc o imagine în cadrul unui sistem digital iar pe de altă parte sunt descrise echipamentele necesare captării și prelucrării imaginilor.

Imagini digitale

Orice imagine se poate exprima printr-o structură bidimensională (tablou, matrice imagine) de date. Un element al imaginii se numește *pixel* (cuvânt preluat din engleză : “picture” - “element”). Aceste date pot fi numere naturale, reale sau complexe, reprezentate în baza 2 pe un număr finit de biți [VERTAN]. Referința la un element al matricii imagine se poate face prin indicarea liniei (*i*) respectiv a coloanei (*j*) în care este amplasat elementul respectiv. După tipul datelor din această structură bidimensională, imaginile prelucrate pot fi împărțite în mai multe categorii:

- imagini scalare, în care fiecare element este un scalar : punctele au doar două valori posibile (alb-negru, imagini binare) și imaginile cu nivele de intensitate (sau de gri) în care elementele pot avea mai multe valori;
- imagini vectoriale, în care fiecare componentă este un vector de numere: imagini color, în care vectorul are trei elemente ce corespund celor trei componente de bază ale oricărei culori, Roșu-Verde-Albastru (Red-Green-Blue – RGB);

După modul reprezentării lor, imaginile pot fi: de intensitate și imagini indexate.

Într-o imagine de intensitate valoarea fiecărui pixel este o măsură directă a intensității luminoase sau a mărimii fizice preluate de senzori. O imagine indexată este acea imagine în care valoarea fiecărui pixel este un indice prin care se regăsește, într-o tabelă (LUT), informația de culoare asociată pixelului respectiv.

Structura sistemului de achiziție și prelucrare a imaginilor

În domeniul achiziției imaginilor, începând cu anii 1990, s-a trecut de la dispozitivele analogice (iconoscop), la cele digitale (camere echipate cu senzori CCD, NMOS, CID), acestea fiind mai robuste, mai compacte și mai ieftine.

Un avantaj major al sistemelor digitale îl constituie flexibilitatea acestora în ceea ce privește programarea. În cadrul tezei de doctorat se vor trata numai sistemele digitale, fiind cele mai răspândite și totodată având în vedere că în aplicațiile practice vor fi folosite astfel de sisteme.

Structura tipică a unui sistem de captare și procesare a imaginilor este arătată în figura 3.11 [CORKE].

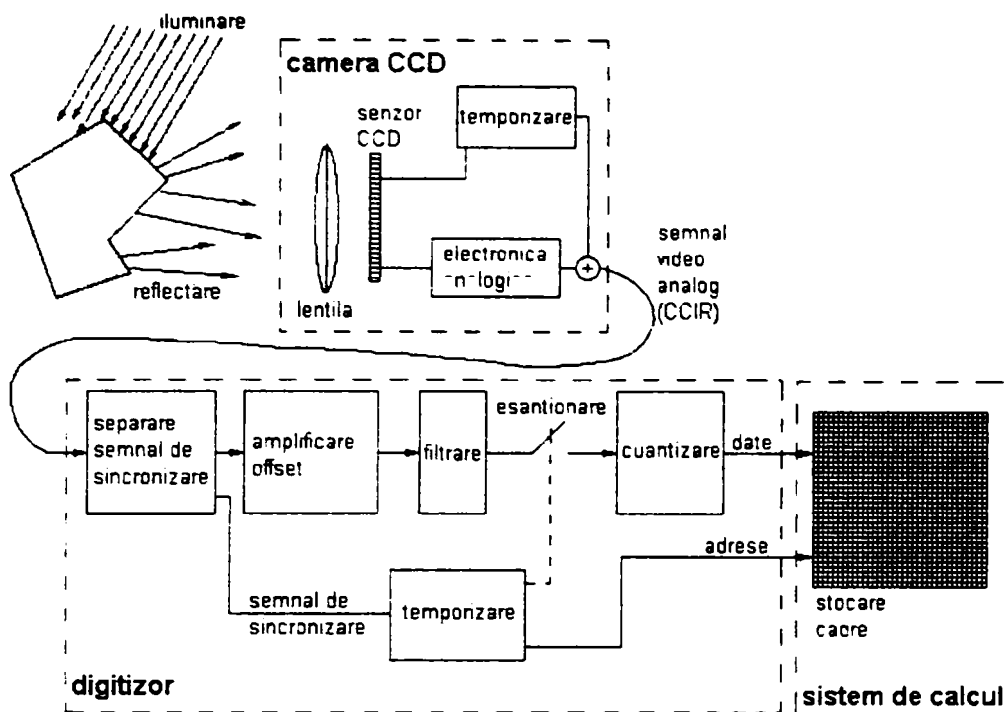


Fig. 3.11 Schema generală a unui sistem de captare și procesare a imaginii.

Acesta este realizat din trei subsisteme :

- a. *subsistemul de captare a imaginii* (camera CCD), ce se compune din:
 - *dispozitivul optic* ("lentila") care are rolul de a focaliza imaginea pe senzor;
 - *senzorul vizual*, („senzor CCD”) transformă energia radiațiilor luminoase în semnale electrice;
 - *blocul de electronică analogică* („electronica analogică”) realizează amplificarea semnalului și pregătirea acestuia pentru transmitere spre digitizor;
 - *blocul de temporizare*, („temporizare”) generează un semnal de sincronizare corespunzător unei linii a imaginii citite. Semnalul de sincronizare este compus cu semnalul video (în schemă compunerea semnalelor este reprezentată de către sumator).
- b. *subsistemul de digitizare* („digitizor”) :
 - *blocul de separare* („separare semnal de sincronizare”), realizează separarea semnalului de sincronizare de semnalul video;
 - *blocul de amplificare* („amplificare, offset”), realizează amplificarea și corectarea offset-ului (deplasării semnalului față de nul);
 - *blocul de filtrare* („filtrare”) realizează filtrarea zgomotelor ce intervin în procesul de transmitere a semnalului;
 - *elementul de eșantionare* („eșantionare”) realizează discretizarea semnalului video în așa fel încât fiecare valoare discretă să corespundă unui pixel din cadrul unei linii;
 - *blocul de temporizare* („temporizare”), pe lângă comanda eșantionării, generează adresa de linie a datelor ce sunt înscrise în memoria sistemului de calcul;
 - *blocul de conversie A/D* („cuantizare”) trunchiază valoarea semnalelor

discrete pentru a putea fi reprezentate în format binar.

c. *subsistemul de stocare și prelucrare* („sistem de calcul”) : realizează stocarea datelor transmise de către digitizor în memoria volatilă în vederea prelucrării, vizualizării pe monitor sau a transcrierii lor în fișier.

Descrierea funcțională a senzorului CCD

Senzorii vizuali digitali cei mai des utilizați sunt senzorii CCD (Charge Coupled Device), ce constau dintr-o matrice de “zone fotosensibile”. Fiecare dintre “zone” acumulează o sarcină electrică proporțională cu iluminarea incidentă pe durata expunerii. Fotonii ce bombardează un pixel generează perechi de electroni-goluri în materialul semiconductor.

Electronii astfel generați sunt captați în “acumulatori de sarcină” de către un câmp electric generat de electrozi de metal. Pachetele de electroni astfel formate sunt apoi transportate de către câmpuri electrice generate de tensiuni multi-tact în electrozii de metal, iar apoi se formează semnale ce se trimit spre digitizor și se încarcă în următoarea linie. Unii fotoni ce au o lungime de undă mare, pot pătrunde mai adânc în stratul semiconductor provocând încărcarea cu sarcină, nu numai a pixelului incident ci și a vecinilor acestuia (“cross talk”).

Acest fenomen duce la creerea unor efecte difuze în imaginea rezultată. Rezultate negative ca de exemplu prezența în imagine a unor zgomote (“dark current”) pot fi produse și de către efectul termoelectric și “efectul de tunel”. Unii senzori de mare precizie sunt echipați cu dispozitive de răcire pentru a preveni aceste efecte.

Module de achiziție a imaginilor

Subsistemul de digitizare („digitizor”) prezentat în figura 3.11, se realizează practic sub forma unor circuite electronice care se cuplează pe magistrala sistemului de calcul, numite “module de achiziție a imaginilor” (frame grabber, video blaster). În acest mod camera CCD se poate cupla direct la modulul de achiziție de imagini ce face parte integrantă din sistemul de calcul.

Sisteme de calcul

În [VERTAN] se arată că unitatea de prelucrare hardware folosită la aplicațiile de prelucrarea imaginilor este în majoritatea cazurilor un PC obișnuit. Sistemul software specializat care realizează efectiv o sarcină concretă poate fi descompus în mai multe module : îmbunătățirea, restaurarea, compresia, segmentarea, ș.a.m.d.

Alegerea sistemului de achiziție și prelucrare de imagini pentru o aplicație concretă.

În domeniul roboticii achiziția și prelucrarea de imagini poate fi împărțită în două domenii distincte și anume aplicații *interactive* și *intercorelate*.

Aplicațiile *interactive* sunt acelea în care sistemul vizual oferă în mod direct informații sistemului de conducere al robotului: identificarea obiectelor dintr-o scenă 3D în care este necesară o examinare a scenei din mai multe situații ale camerei; urmărirea unei ținte; situarea efectorului final într-o poziție adecvată prehensării unui obiect; detectarea unor obstacole ce intersectează traiectoria robotului.

Caracteristicile esențiale ale acestor aplicații sunt:

- condițiile de iluminare variază în domenii largi;
- timpurile de realizare a mișcărilor robotului impun condiții restrictive asupra timpului de achiziție și analiză a imaginii;
- mișcărilor robotului se efectuează în spațiu, deci se impune tratarea 3D a imaginilor.

Aplicațiile *intercorelate* vizează observarea unor caracteristici vizuale a unor obiecte în scopul clasificării și/sau identificării acestora : analiza texturilor; analiza unor scene fără modificarea poziției camerei; analiza micrografiilor; controlul formei și/sau dimensiunilor unor obiecte;

Caracteristicile esențiale ale acestor aplicații sunt:

- condițiile de iluminare variază în domenii înguste (sunt quasi-constante);
- timpurile de realizare a mișcărilor robotului nu impun restricții asupra timpului de achiziție și analiză a imaginii;
- se tratează cu precădere imagini 2D.

Configurarea structurii unui sistem de achiziție și prelucrare a imaginilor este întotdeauna dependentă de aplicația în care este utilizat sistemul. Diversitatea mare de aplicații a impus dezvoltarea unei game largi de componente (camere CCD, module de achiziție imagini, programe pentru prelucrarea imaginilor).

Din cele prezentate anterior se poate afirma că există două criterii majore ce se impun a fi luate în considerare la configurarea unui sistem de achiziție și prelucrare a imaginilor.

a. Criteriul calității imaginii.

Acest criteriu include mai mulți parametri de care depinde realizarea lui, și anume :

1. *Rezoluția imaginii* – este parametrul ce definește numărul de pixeli pe unitatea de suprafață a dispozitivului de afișare, depinzând în primul rând de calitatea senzorului CCD;
2. *Adâncimea de culoare* – depinde de sensibilitatea senzorului CCD dar și de capacitatea de cuantizare a digitizorului;
3. *Eliminarea zgomotelor* – depinde de construcția senzorului, filtrarea efectuată de către digitizor și filtrarea prin soft;

b. Criteriul vitezei de achiziție și prelucrare.

Parametrii de care depinde viteza de achiziție sunt :

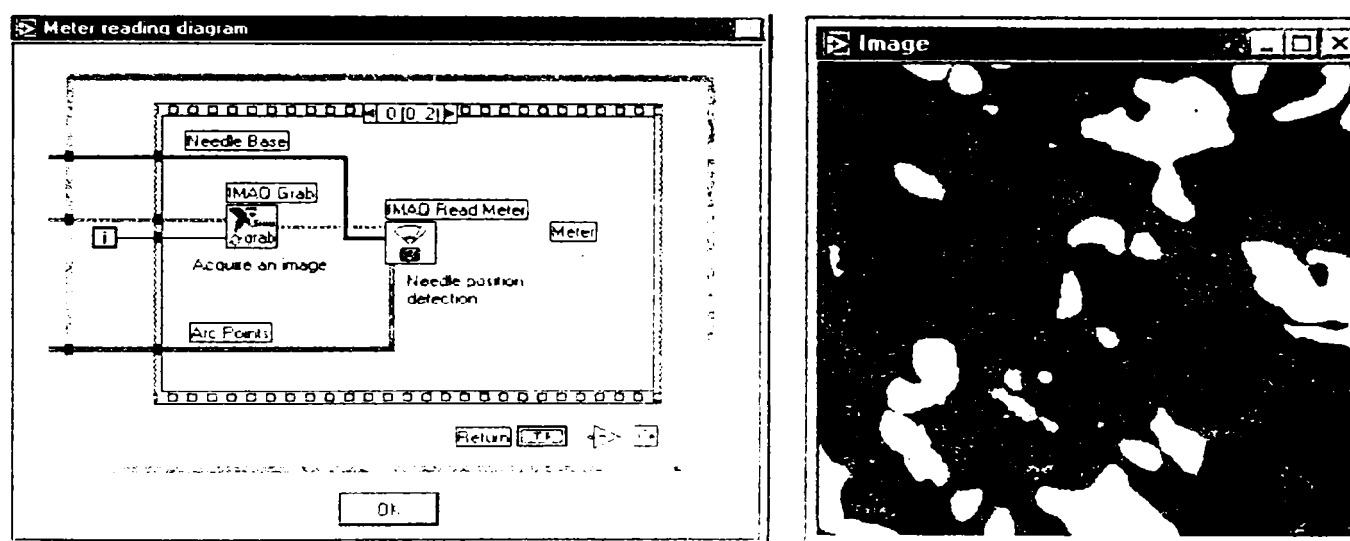
1. *Rata de achiziție a senzorului CCD* – depinde de timpul de expunere și de viteza cu care se realizează transferul sarcinilor în regiștrii senzorului CCD;

2. *Viteza de eșantionare a digitizorului* – depinde de întârzierile introduse de componentele electronice ale digitizorului;
3. *Viteza de acces la memoria volatilă a sistemului de calcul* – depinde de tehnologia de realizare a chip-urilor de memorie (EDORAM, SDRAM, DDRAM).
4. *Timpul total de transfer al datelor* – depinde de suma întârzierilor tuturor componentelor mai puțin a celor care se suprapun.

Aplicațiile din cadrul tezei de doctorat, ce vizează vederea artificială, au ca obiectiv analiza structurilor metalografice, încadrându-se în domeniul aplicațiilor intercorelate.

În domeniul sistemelor de prelucrare de imagini ce au posibilitatea să proceseze imagini ale microstructurilor, se prezintă câteva exemple.

Firma “National Instruments” produce module grafice ce sunt integrabile pe calculatoare personale și livrează împreună cu acestea softuri pentru dezvoltarea aplicațiilor în domeniu (IMAQ). Dezvoltarea aplicațiilor se realizează prin editarea programelor în mod grafic cu ajutorul editorului prezentat în figura 3.12.a,b.



a.

b.

Fig. 3.12. Tehnica de programare pentru mediul de dezvoltare IMAQ (a). imaginea unei microstructuri, realizată cu programul IMAQ (b).

Firma canadiană Clemex livrează sisteme de analiză a imaginilor metalografice și înregistrare a datelor analizei în diverse variante, figura 3.13.a,b.

Peformanțe similare cu cele ale sistemului Clemex [CLEMEX1] sunt realizate și de către sistemul „MetaMorph” al firmei Universal Imaging Corporation [UIC1] și sistemul „LUCIA” al firmei Laboratory Universal Computer Image Analysis [LUCIA1], în domeniul metalografiei și a științei materialelor, de sistemul firmei „LIFESPAN – Biosciences” [LIFESPAN1] și de sistemul „ImageXpress” al firmei CMIS Research [CMIS1], în domeniul bio-medical.

Fiind dată marea diversitate a acestor metode și relativa noutate a conceptelor aplicate, s-au definit mai multe clasificări a metodelor de prelucrare a imaginilor.



Fig.3.13. Sistemul Clemex CMT (a). interfața grafică a programului Cemex (b).

O clasificare ar putea fi definită după rolul pe care îl are metoda respectivă (figura 3.14.) și anume de analiză a imaginii sau de înțelegere (interpretare) a imaginii [IACV].

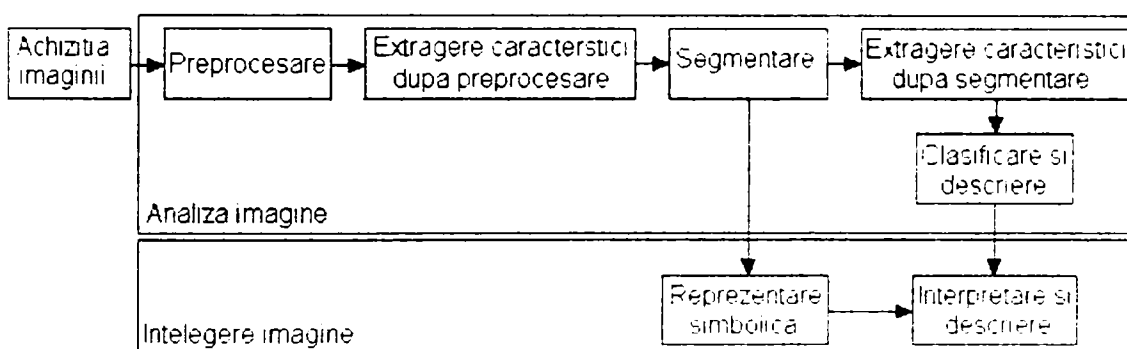


Fig.3.14. Schema generală a unui sistem de prelucrare a imaginilor [IACV].

3.3.2. Îmbunătățirea imaginii

Operațiile de modificare a contrastului sunt asocieri ce leagă nivelul de intensitate (sau gri) original, de noua sa valoare [VERTAN]. O asemenea asociere este reprezentată prin funcția:

$$y = T(u), u \in [0; L - 1] \quad (3.61)$$

în care : - y reprezintă valoarea nivelului de gri a imaginii după transformare;

- u reprezintă valoarea nivelului de gri a imaginii înainte de transformare;

- L este numărul nivelelor de cuantizare a nivelelor de gri;

- T este funcția de transformare cu condiția: $0 \leq T(u) \leq L - 1, \forall u \in [0; L - 1]$.

Pentru o imagine u de $M \times N$ pixeli și L nivele de gri, histograma este definită [VERTAN], ca probabilitatea de apariție în imagine a diferitelor nivele de gri posibile.

$$h(i) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \delta(i - u(m, n)) \quad i = 0, 1, \dots, L - 1 \quad (3.62)$$

unde : - $M \times N$ este dimensiunea imaginii definite ca o matrice al nivelelor de gri;

- i este nivelul de gri curent;
- $u(m,n)$ nivelul de gri al punctului de coordonate m,n ;
- L valoarea maximă a nivelelor de gri;

Operațiile de îmbunătățire a imaginilor care realizează redistribuirea nivelelor de gri, astfel ca acestea să ocupe întreaga gamă de variație disponibilă, în mod uniform se numește *egalizare de histogramă*.

3.3.3. Filtrare.

Operația de filtrare liniară calculează noua valoare a unui pixel al imaginii $y(m,n)$ ca o combinație liniară a unui număr de valori din imaginea originală $u(m,n)$:

$$y(m,n) = \sum_{(k,l) \in W} w_{kl} u(m-k, n-l) \quad (3.63)$$

în care w este vecinătatea punctului curent numită fereastră de filtrare și este o formă plană, descrisă ca o mulțime de puncte din sistemul de coordonate (m,n) , în coordonate relative (k,l) . Scalarii w_{kl} sunt atașați pozițiilor (k, l) din fereastra de filtrare și poartă numele de coeficienți ai filtrului.

Limitările operațiilor liniare de filtrare apar în momentul în care se consideră zgomote ce nu au distribuție normală, și nu sunt aditive. O clasă de metode de filtrare, ce pot fi folosite cu succes pentru eliminarea acestor zgomote sunt cele bazate pe ordonare "*rank order filtering*". Operația de ordonare plasează valorile aberante (extremale) la capetele șirului de valori (deci într-o zonă bine definită), permițând astfel identificarea și eliminarea acestora.

3.3.4. Transformări integrale.

Cea mai utilizată metodă de descompunere a unui semnal complex în semnale mai simple este transformarea Fourier. Elementul esențial al transformării Fourier este faptul că semnalul este descompus într-o serie de sinusoidale. În practică folosind prelucrarea computerizată a datelor, singurul tip de transformare utilizat este Transformarea Fourier discretă, întâlnit în majoritatea documentațiilor sub denumirea de DFT („Discret Fourier Transform”). Prin intermediul DFT semnalul inițial se va descompune într-un set de $N/2+1$ cosinusoide și un set de $N/2+1$ sinusoidale. Fiecare dintre acestea are o frecvență predefinită notată cu k , semnificând numărul de perioade ce se regăsesc în cele N valori ale semnalului inițial.

Amplitudinile funcțiilor cosinus sunt notate $\text{Re } \bar{X}[k]$ iar amplitudinile funcțiilor sinus sunt notate $\text{Im } \bar{X}[k]$. Ecuația sintezei Fourier (IDFT) este dată de relația :

$$x[i] = \sum_{k=0}^{N/2} \text{Re } \bar{X}[k] \cos(2\pi ki / N) + \sum_{k=0}^{N/2} \text{Im } \bar{X}[k] \sin(2\pi ki / N) \quad (3.64)$$

în care :

- $x[i]$ reprezintă valoarea i a setului de date de intrare (date achiziționate);

- i este indicele valorilor de intrare ($i = 0 \dots N-1$);
- k reprezintă frecvența cosinusoidei sau sinusodei curente;
- N este numărul eșantioanelor setului de intrare;
- $\text{Re } \bar{X}[k]$ este amplitudinea cosinusoidei de frecvență k ;
- $\text{Im } \bar{X}[k]$ este amplitudinea sinusodei de frecvență k .

Pentru a aplica aceste transformări unor imagini, se definește transformarea Fourier discretă bidimensională (DFT2), calculată prin transformata Fourier rapidă bidimensională (FFT2). Dacă se consideră că valorile de intensitate ale imaginii sunt stocate într-o matrice bidimensională, atunci FFT2 constă în :

- aplicarea FFT asupra fiecărei linii a matricii imagine în parte;
- rezultatele transformării se înscriu în două matrici intermediare, respectiv matricea valorilor reale și cea a valorilor imaginare;
- se aplică apoi FFT asupra coloanelor matricilor intermediare rezultând de această dată matricile real și imaginar definitive ale transformării.

3.3.5. Morfologie.

În cadrul prelucrării morfologice imaginea este considerată ca un ansamblu asupra căruia se aplică transformări ce au ca scop comparația cu mulțimi mai simple, numite elemente de structură. Scopul acestor transformări este extragerea de forme mai simple din formele complexe ale imaginii inițiale [VERTAN].

Transformarea “hit or miss” se definește pe baza unei părți (B_1, B_2) a elementului structurant $B: B=B_1 \cup B_2$ și $B_1 \cap B_2 = \emptyset$ ca:

$$A \otimes B = \{x / (B_1)_x \subseteq A\} \cap \{x / (B_2)_x \subseteq A^c\} \quad (3.65)$$

În relația (3.65) A este mulțimea căreia i se aplică transformarea, A^c este complementarea mulțimii A , B este elementul structurant și $(B_i)_x$ este mulțimea B_i translata cu vectorul x .

Erodarea morfologică a mulțimii A prin elementul structurat B se definește ca mulțimea punctelor (elementelor) cu care se poate translata elementul structurant astfel încât aceasta să fie inclusă în mulțimea de prelucrat A [VERTAN]:

$$A \ominus B = \{x / B_x \subseteq A\} \quad (3.66)$$

Dilatarea morfologică a mulțimii A cu elementul structurant B se definește ca mulțimea punctelor (elementelor) cu care se poate translata elementul structurant astfel încât aceasta să aibă puncte comune cu mulțimea de prelucrat A :

$$A \oplus B = \{x / B_x \cap A \neq \emptyset\} \quad (3.67)$$

3.3.6 Segmentare [VERTAN].

Segmentarea imaginilor se referă la descompunerea unei scene (imagini) în componentele sale. În urma procesului de segmentare vor fi extrase din imagine obiecte distincte, regiuni ce satisfac anumite criterii de uniformitate, sau alte elemente. Din punct de vedere matematic [VERTAN], segmentarea unei imagini f este definită ca partiționarea lui f (3.68) într-un ansamblu de mulțimi disjuncte nevide și conexe (3.69), ce satisfac fiecare un anumit criteriu C (3.70).

$$f = \bigcup_{i=1}^c f_i, f_i \text{ conexe} \quad (3.68)$$

$$f_i \cap f_j = \emptyset, \forall i \neq j, f_i \text{ este conexa}, \forall i \quad (3.69)$$

$$C(f_i) = TRUE, \forall i \text{ si } C(f_i \cup f_j) = FALSE, \forall i \neq j \quad (3.70)$$

După primitivele de extras, tehnicile de segmentare se împart în două categorii fundamentale : tehnicile de segmentare orientate pe regiuni și tehnicile de segmentare orientate pe contur.

Segmentarea orientată pe regiuni

În general, operația de segmentare orientată pe regiuni urmărește extragerea din imagine a zonelor ocupate de diferite obiecte prezente în scenă. Un obiect se definește ca o entitate caracterizată de un set de parametri ale căror valori nu se modifică în diferitele puncte ce aparțin entității considerate, deci obiectul are proprietatea de uniformitate a parametrilor de definiție.

Metoda Bhattacharyya se bazează pe descompunerea histogramei în moduri individuale Gaussiene, adică se încearcă exprimarea histogramei imaginii ca o sumă ponderată de funcții de densitate de probabilitate de tip normal (Gaussian).

Metoda de segmentare cu prag optim face apel la teoria deciziilor (criteriul de decizie Bayes) pentru stabilirea valorii pragurilor de segmentare ce optimizează un anumit criteriu de eroare.

Aplicarea tehnicilor de segmentare pe histogramă este condiționată în primul rând de reprezentarea diferitelor clase de obiecte din imagine pe intervale de nivele de gri diferite care nu se suprapun (sau se suprapun parțial pe porțiuni foarte mici); apoi este necesară cunoașterea numărului de tipuri de obiecte diferite. În fine, se presupune că valorile prag corespunzătoare se pot determina cu o precizie corespunzătoare.

Segmentarea orientată pe contururi

Principiul acestor metode constă în definirea punctelor de contur ca fiind acei pixeli ai imaginii în care apar schimbări importante (abrupte) ale nivelului de gri. Măsurarea acestei variații se face prin operatori derivativi de tip gradient. Pentru o

imagine cu suport spațial continuu, pe direcția unei muchii, derivata va fi maximă. Derivata imaginii pe direcția r , ce face unghiul θ cu orizontala, este dată de combinația liniară a derivatelor parțiale pe direcțiile orizontală și verticală :

$$\frac{\partial f}{\partial r} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial r} = \frac{\partial f}{\partial x} \cos \theta + \frac{\partial f}{\partial y} \sin \theta \quad (3.71)$$

Din punct de vedere practic, pentru implementarea acestei metode este nevoie de calculul, pentru fiecare punct la imaginii, a derivatelor parțiale f_x și f_y . Valoarea gradientului maxim din fiecare punct al imaginii este apoi comparată cu un prag fixat. Dacă pragul este depășit (deci gradientul maxim în pixelul respectiv este suficient de important) atunci pixelul testat este pixel de contur.

Unul dintre principalele dezavantaje ale metodelor de gradient este precizia slabă de localizare a conturului în condițiile unei pante mai puțin abrupte. Derivata a doua poate fi însă folosită pentru a determina capetele tranziției, sau pentru a marca centrul tranziției. Operatorul bazat pe trecerea prin zero a derivatei secunde este operatorul “zero-crossing”. Precizia sporită a operatorilor de acest tip (*laplacieni*) conduce însă la o sensibilitate crescută la zgomote (mai mare decât a operatorilor de gradient) iar un alt dezavantaj constă în faptul că laplacianul nu mai conține informație relativă la direcția tranziției.

3.3.7. Recunoaștere [VERTAN].

Recunoașterea automată a formei obiectelor se realizează cu ajutorul unor caracteristici “extrase” din imaginea analizată care se numesc “parametrii de formă”.

Prin parametrii de formă se înțelege în general orice scalar sau funcție (cu suport unidimensional sau bidimensional) asociată unei forme plane pe care o caracterizează. Forme asemănătoare sunt caracterizate de parametrii de formă ce le sunt asociați, formele diferite prezintă diferențe mari între parametrii de formă ce le sunt asociați. Deoarece formele ce urmează a fi recunoscute pot să apară în scenă în orice poziție, acești parametrii trebuie să fie invarianți la translație, rotație și scalare. Tehnicile de recunoaștere a formelor sau de clasificare sunt precedate întotdeauna de o etapă de extragere a caracteristicilor formei.

Parametrii geometrici

Această categorie de parametrii se bazează pe măsurarea unor trăsături geometrice simple arie (S), perimetru (P), numărul lui Euler (numărul de regiuni conexe). Cum nu toți acești parametrii sunt invarianți, în practică se folosesc rapoarte ale acestora.

Raportul de compactitate este raportul dintre pătratul perimetrului și suprafața

formeii:

$$k = \frac{P^2}{4\pi S} \quad (3.72)$$

Circularitatea formeii (măsura în care forma se deosebește de un cerc) poate fi definită ca un raport al razelor cercurilor circumscrise (R) și înscrise (r) formeii:

$$c = \frac{R}{r} \quad (3.73)$$

Momente statistice.

Interpretând funcția caracteristică a formeii ca pe o funcție de densitate de probabilitate bidimensională, se pot defini momentele statistice asociate celor două variabile aleatoare.

În cazul imaginilor de dimensiuni $M \times N$, coordonatele sunt discrete și funcția este exprimată prin relația :

$$m_{pq} = \sum_{x=1}^M \sum_{y=1}^N x^p y^q \quad (3.74)$$

Folosirea momentelor statistice pentru caracterizarea unei forme nu asigură îndeplinirea a nici unuia dintre principiile de invarianță căutate. De aceea au fost introduse momente statistice invariante.

Momentele statistice invariante la translație sunt momentele statistice centrate:

$$\mu_{pq} = \sum_{x=1}^M \sum_{y=1}^N (x - \bar{x})^p (y - \bar{y})^q \quad (3.75)$$

în care \bar{x} și \bar{y} reprezintă valori medii.

Momentele statistice invariante la translație și scalare sunt definite de:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}, \gamma = 1 + \frac{p+q}{2} \quad (3.76)$$

Folosind momentele invariante, se mai pot deduce alte atribute cum ar fi de exemplu excentritatea suprafeței, care măsoară gradul de uniformitate al distribuției punctelor formeii în jurul centrului și orientarea suprafeței.

3.3.7. Metode de caracterizare și recunoaștere a texturilor.

Metodele descrise mai sus, se utilizează în mod frecvent la caracterizarea și recunoașterea „obiectelor”. Microstructurile metalografice pot fi privite însă și ca structuri abstracte ce nu pot fi corelate cu o formă bine definită (în termeni geometrici) ci ca „texturi”. În [IACV] termenul de „textură” se referă în general la repetarea în imagine a unei structuri imagistice de bază numită „texel” (prin similitudine cu „pixel”). Un texel conține un grup de pixeli ce pot avea o distribuție spațială periodică, pseudo-periodică sau aleatoare. Clasificarea metodelor de

caracterizare a texturilor după [IACV] este dată în figura 3.15.

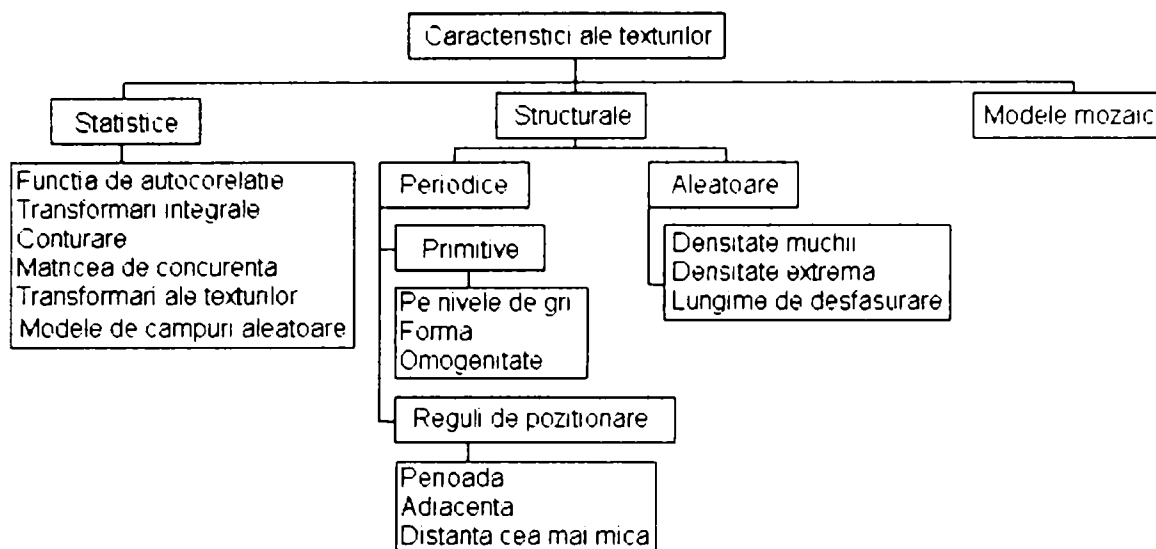


Fig.3.15. Caracteristici ale texturilor [IACV].

În [TUCERYAN] se arată că „Structura texturilor este atribuită unor modele repetitive în care elementele sau primitivele sunt aranjate conform unor reguli de amplasare”.

Deși nu există o definiție riguroasă a texturii, s-au formulat o serie de proprietăți intuitive ce sunt general acceptate [TUCERYAN]:

- „Textura este o proprietate legată de arie, textura unui punct nu are sens. Deci textura este o proprietate contextuală iar definiția ei trebuie să implice valorile de intensitate a vecinătății spațiale. Mărimea acestei vecinătăți depinde de tipul texturii și de mărimea primitivelor ce formează textura”.
- „Textura implică distribuția spațială a nivelelor de intensitate. Histogramele bidimensionale și matricile de co-ocurență sunt unelte de analiză rezonabilă a texturilor”.
- „Textura dintr-o imagine poate fi percepută la diverse scări sau rezoluții. Se consideră de exemplu imaginea unui perete alcătuit din cărămizi. La o rezoluție slabă textura este percepută ca fiind formată din cărămizile individuale, detaliile interioare ale acestora fiind pierdute. La o rezoluție mai mare, când sunt numai câteva cărămizi în câmpul vizual, textura percepută este cea din interiorul cărămizii”.
- „O regiune este percepută ca având o textură când numărul de primitive din imagine este mare. Dacă în imagine sunt prezente doar câteva primitive acestea sunt percepute ca obiecte și nu ca textură”.

Faptul că percepția texturii este atât de diferită a dus la elaborarea unui număr mare de metode de caracterizare. O analiză comparativă a metodelor de caracterizare a texturilor este dată în [MATERKA]. Din studiul mai multor lucrări în domeniu se poate concluziona că nu există o metodă universală, iar alegerea metodei se realizează în funcție de aplicație. În sprijinul testării metodelor de caracterizare a texturilor s-au

creat baze de date cu imagini texturate, cum ar fi „albumul Brodatz” sau „Columbia-Utrecht texture database” ce pot fi accesate pe internet.

Metodele de analiză a texturilor pot fi clasificate și după scopul în care sunt utilizate [TUCERYAN] :

- segmentarea texturilor –delimitarea zonelor din imagine cu texturi diferite;
- clasificarea texturilor;
- sinteza texturilor – realizarea unor algoritmi de modelare a texturilor, texturile sintetizate fiind apoi utilizate în scopuri de clasificare sau segmentare;
- formă din textură – utilizarea caracterizării texturii pentru recunoașterea obiectelor 3D.

Un domeniu de aplicații recent îl constituie caracterizarea texturilor dinamice și temporale [SAISAN, ZHONG], ce apar în secvențe de imagini.

Se prezintă în continuare câteva tipuri de metode reprezentative pentru caracterizarea texturilor.

Metode statistice

Printre primele metode utilizate în caracterizarea texturilor sunt statisticile spațiale, dintre care cele mai utilizate sunt :

- de ordinul întâi : depind numai de valorile de intensitate ale pixelilor individuali și nu depind de valorile pixelilor vecini - pot fi calculate din histogramele imaginii;
- de ordinul doi : sunt proprietăți a perechilor de pixeli. Julesz a emis ipoteza că imagini ale căror statistici de ordin doi sunt similare nu pot fi discriminate pre-atenționat definind caracteristici ca: terminații, închideri, colinearități ce duc la conceptul de „texton”.

Haralick și Picard au studiat texturile cu ajutorul matricilor de co-ocurență ce estimează proprietăți ale imaginilor ce sunt în legătură cu statisticile de ordinul 2. Atât parametrii calculați din matricea de co-ocurență cât și cei calculați prin auto-corelație au la bază estimarea relației dintre valorile de intensitate a unor pixeli, aflați la o anumită distanță d (exprimată în pixeli), în imaginea analizată.

Dezavantajul acestor abordări constă în faptul că nu există o metodă bine stabilită de calcul a distanței d , iar iterarea exhaustivă a valorilor lui d nu este fezabilă. Deasemenea numărul mare de trăsături ce se pot calcula cu ajutorul acestor metode ar necesita o metodă de selecție a trăsăturilor reprezentative, care din nou lipsește.

Este evident că metodele ce utilizează matrici de co-ocurență și funcțiile de auto-corelație sunt similare cu metodele frecvențiale, relațiile dintre acestea fiind arătate în figura 3.16.

Metodele statistice sunt utilizate într-o serie de lucrări : attribute texturale bazate pe parametrii statistici sunt utilizate în mai multe aplicații [ROSENBERG],

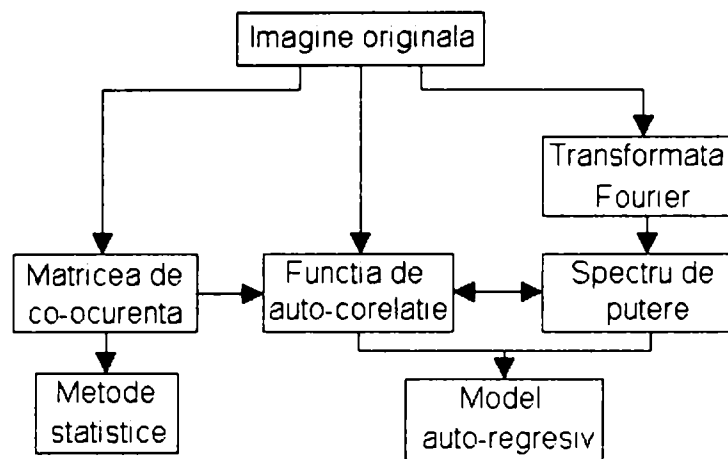


Fig.3.16. Relația dintre metodele statistice și modelele autoregresive.

[MACGILLAVRY] metode statistice de ordinul 1 și 2, [CULA1] statistici de ordinul 1 în recunoașterea 3D a texturilor, [CULA2] realizarea unei biblioteci de textoni pentru texturi 3D utilizând tehnici de învățare și histogramme, [SHARMA] comparații între diferite metode statistice (autocorelație, matrici de co-ocurență), [MESSER] utilizează modele statistice pentru recunoașterea țintelor prin texturi, [ELSHOLZ] caracterizarea suprafețelor rugoase, metode statistice și corelație, [KARKANIS] descriptori statistici în imagini medicale, [JASVINDER] sinteza texturilor bazată pe descompunere piramidală, în care se pornește de la zgomotul alb și se redefinesc pixeli până când parametrii statistici ai texturii sintetice corespund cu cei ai texturii reale.

Metode geometrice

Metodele geometrice sunt utilizate în primul rând în căutarea primitivelor texturii și a regulilor de amplasare a primitivelor în imagine, urmând ca pe baza acestora să se aplice metode statistice [TUCERYAN]. Cu ajutorul poligoanelor Voronoi se pot defini vecinătăți spațiale locale iar distribuțiile acestora sunt reflectate în forma poligoanelor Voronoi. Metodele structurale presupun că texturile sunt definite prin plasarea primitivelor conform unor reguli bine stabilite (inferența regulilor de amplasare a primitivelor). Aceste metode sunt aplicabile numai texturilor foarte regulate.

Metode bazate pe model

Metoda câmpurilor aleatoare – se modelează texturile pe baza câmpurilor Markov aleatoare, presupunând că intensitatea unui pixel depinde numai de vecinătatea imediată a acestuia. În [WANG] se utilizează modele de câmpuri aleatoare Markov în clasificarea texturilor.

În afară de metodele amintite se mai pot aminti:

- analiza fractală [FARDIN] în care se studiază dependența de scară a rugozității suprafețelor rocilor, bazată pe calculul dimensiunii fractale și a parametrului de amplitudine;
- în [CARVALHO] se utilizează metoda numărării casetelor la imagini infraroșu a formațiunilor meteorologice;
- metode specifice procesării semnalelor [FUJITA] bazată pe modele a secvențelor de imagini ale unor formațiuni naturale (valuri curenți, fum);
- în [SZIRANYI] este descrisă segmentarea texturilor cu ajutorul rețelelor neuronale CNN (Cellular Neural Networks);
- în [CLARK] se utilizează analiza Fourier a texturilor pe imagini neregulare utilizând algoritmi de deconvoluție.

3.3.9. Sinteza cercetărilor recente în domeniul analizei de imagini.

În paragrafele precedente s-au prezentat metode de prelucrare a imaginilor ce reprezintă obiecte și texturi. Deși aceste metode au o aplicație largă în domeniul analizei de imagini, ele nu acoperă aplicațiile din ce în ce mai complexe ce apar în domeniul roboticii, științei materialelor, medicinei sau al biologiei. Pe de altă parte dezvoltarea tehnicii de calcul, a conceptelor teoretice moderne și a mediilor de programare oferă posibilitatea dezvoltării de noi metode. Din studiul lucrărilor de specialitate apărute la nivelul anilor 2000, se disting câteva concepte a căror aplicare dă rezultate ce nu pot fi obținute cu ajutorul metodelor „clasice”. Aceste concepte sunt: analiza fractală, analiza "wavelet", analiza cu ajutorul algoritmilor genetici, analiza cu ajutorul automatelor celulare, analiza cu ajutorul rețelelor neuronale.

Domeniile în care se utilizează prelucrarea de imagini sunt foarte diverse, reconstrucția 3D a corpurilor, pe baza imaginilor (secțiuni sau vederi) [DONG1, PARDO1, WOO1], segmentarea texturilor [BHALEARO1, CHIZHIK1, DEBACKER1, GOTTSTEIN1], simularea creșterii cristalelor [VARMA1, BASANTA1], fiind doar câteva din acestea.

Luând în considerare specificul tezei de doctorat, metodele moderne de analiză de imagini au fost sintetizate pe două categorii în funcție de domeniul de aplicare al acestora. În prima categorie au fost selectate metode ce fac posibilă analiza automată a rugozității suprafețelor ce este utilă în cadrul operațiilor de pregătire a eșantioanelor metalografice, iar în a doua categorie au fost selectate metode de analiză a microstructurilor.

Metode de analiză a rugozității.

Din această categorie sunt prezentate câteva metode noi de măsurare și estimare a rugozității suprafețelor obținute prin prelucrări mecanice.

Ca principiu de măsurare majoritatea studiilor realizate în domeniul măsurării rugozității suprafețelor utilizează fie măsurarea cu ajutorul profilometrelor fie cu ajutorul unui sistem de achiziție și prelucrare a imaginilor.

În cazul măsurării cu profilometrul se încearcă reducerea timpilor de măsurare, ca de exemplu măsurarea în spirală în loc de traiectorie dreaptă [WIECZROWSKI1] sau îmbunătățirea preciziei prin realizarea unor profilometre ultraprecise [LEACH1]. În [BRINKMANN1] se realizează prelucrarea datelor obținute prin măsurarea rugozității cu ajutorul profilometrului, metoda prezentată asigurând o precizie ridicată și evaluarea undulațiilor suprafeței. În [LU1] se prezintă măsurarea rugozității cu ajutorul unui sistem dublu compus dintr-un profilometru și un dispozitiv laser bazat pe reglarea distanței focale.

Ca alternativă la măsurarea cu profilometrul, evaluarea rugozității pe baza imaginii suprafeței este din ce în ce mai utilizată, metoda prezentând avantajul unei măsurări fără contact.

În [BALEIX1] se analizează suprafața unor scule utilizate la deformarea superplastică supuse unor variații intense de temperatură. Pe suprafața acestor scule se formează straturi de oxizi (în funcție de rugozitate inițială a suprafeței și de regimul termic) ce sunt fotografiate cu ajutorul unui microscop electronic și sunt analizate cu ajutorul unui program de prelucrare a imaginilor. De obicei rugozitatea suprafeței este măsurată cu ajutorul unui profilometru ce realizează o profilogramă citind variațiile înălțimii neregularităților de-a lungul unei linii drepte. Neregularitățile suprafeței pot fi urmărite și pe o imagine considerându-se că regiunile mai luminoase corespund unor vârfuri iar regiunile întunecoase corespund unor depresiuni. Astfel analiza unei imagini de 24 mm² corespunde la 100 de profilograme. Aplicând un algoritm de interpolare se poate obține o diagramă 3D a suprafeței studiate. În lucrare, analiza imaginilor a permis determinarea relației dintre mărimea petelor de oxid și grosimea acestora (ce va crește rugozitatea suprafeței), funcție de timpul de expunere la factorii oxidanți.

În lucrarea [YANAGI1] se definește o metodă de identificare a vârfurilor și direcționalității suprafețelor texturate prin analiza imaginii suprafeței. În primă instanță se utilizează o aproximare sinusoidală a asperităților suprafeței pentru a determina un număr rezonabil de regiuni de analiză ce se vor constitui din maximele sinusoidale de aproximare. Pe aceste regiuni se caută maxime locale și pornind de la acestea se determină direcționalitatea asperităților, legând între ele un număr de maxime găsite. În lucrare se demonstrează că această metodă este superioară metodelor de extragere a muchiiilor utilizată în alte lucrări.

În [JIANG1] se arată că suprafețele prelucrate prin strunjire, frezare, rectificare prezintă o complexitate ridicată atunci când sunt analizate cu ajutorul microscopelor

optice, electronice sau microscopice utilizând forța atomică (AFM – Atomic Force Microscope). Suprafețele izotropice (simplu-fractale) prelucrate mecanic, se pot caracteriza prin funcția M-W (Mandelbrot-Weirstrass). Pentru caracterizarea suprafețelor anisotropice lucrarea propune o metodă bazată pe utilizarea algoritmului FFT bidimensional în calculul dimensiunii fractale. Se prezintă de asemenea și un studiu de caz pentru suprafețe rectificite.

Ideea de bază prezentată în lucrarea [ZAHOUANI1] este descompunerea rugozității suprafeței privită ca o topologie 3D, definită prin amplitudine, lungime de undă și direcție, în elemente de formă convexă („summit”) și reprezentarea grafică a acesteia cu ajutorul unor arbori morfologici. Potrivit autorilor se pot defini corelații clare între caracteristicile arborilor morfologici obținuți și proprietățile tribologice ale suprafețelor tehnice analizate.

În [LEE1] se arată că interdependența dintre rugozitatea suprafeței, prelucrate printr-o anumită metodă tehnologică, și imaginea acesteia a fost studiată în mai multe lucrări și se bazează, în principiu, pe dependența funcțională dintre înălțimea rugozității și nivelul de gri a pixelilor din imaginea suprafeței. Autorii au studiat stabilirea acestei corelații cu ajutorul unei rețele polinomiale, antrenate cu rezultatele unui set de experimente, ce au ca intrare viteza de așchiere, avansul, adâncimea de așchiere și media aritmetică a nivelului de gri al imaginii. Rețeaua polinomială este capabilă să estimeze rugozitatea suprafeței cu o precizie acceptabilă, ceea ce a fost demonstrat și prin măsurările comparative realizate cu ajutorul unui rugozimetru. Avantajul metodei descrise constă în posibilitatea aplicării on-line a acesteia în cazul prelucrărilor prin strunjire.

Lucrarea [JOSSO1] propune o nouă metodă de analiză și caracterizare a rugozității suprafețelor. Cu ajutorul unui algoritm compus din trei faze se realizează clasificarea imaginilor texturilor unor suprafețe obținute prin mai multe tipuri de prelucrări tehnologice (turnare, frezare, rectificare). În primă fază imaginea este supusă unei transformări „wavelet” normalizată în frecvență obținându-se un set de imagini pentru scări și unghiuri diferite. În a doua fază se extrag valorile unor parametrii caracteristici pe baza cărora în a treia fază se realizează clasificarea imaginilor cu ajutorul unor clasificatori și discriminanți.

Din sinteza cercetărilor prezentate se poate reține concluzia că metoda de analiză a imaginii suprafeței este de preferat măsurării cu ajutorul profilometrului, deoarece este mai rapidă și nu necesită contact cu piesa analizată. S-a arătat de asemenea că există o dependență clară între înălțimea asperităților și nivelul de gri, (părțile mai înalte ale asperităților corespunzând pixelilor de intensitate mai mare) și că, în principiu, estimarea rugozității se poate realiza în mod similar cu analiza texturilor.

Metode ce pot fi utilizate în domeniul analizei microstructurilor.

Imaginile microstructurilor sunt, în principiu, mult mai complexe decât cele ale rugozității suprafețelor, ceea ce impune utilizarea unui arsenal de metode avansate.

Analiza fractală.

De la definirea lor de către Mandelbrot în 1970, fractalii au făcut obiectul multor studii și cercetări în domeniul matematicii și al modelării semnalelor. În domeniul analizei imaginilor fractalii oferă o alternativă la descrierea euclidiană a obiectelor cuprinse în imagine. O tratare exhaustivă a noțiunilor de bază despre fractali este dată în [OLIVER1]. Datorită proprietății de auto-similaritate s-au dezvoltat două domenii mari de utilizare a fractalilor, compresia de imagini și caracterizarea imaginilor prin dimensiunea fractală.

Compresia de imagini cu ajutorul algoritmilor fractali [SAUPE1, COHEN1] s-a dezvoltat odată cu nevoia de a transmite și stoca imagini în format electronic, ce să ocupe cât mai puțin spațiu de memorie. Primele încercări ale lui M. Barnsley (1981, și apoi Barnsley și Sloan în 1987), aplicarea sistemului funcției iterate (IFS – iterated function systems - șablonul fractal din care prin iterații succesive se dezvoltă imaginea fractală), nu a adus rezultatele scontate. Problema de bază a fost faptul că prin IFS imaginea se reprezenta ca un colaj realizat din transformate ale imaginii însăși. Rezolvarea acestei probleme a fost dată de către A. Jaquin în 1989, prin introducerea conceptului de redundanță a părților imaginii. Astfel, imaginea este reprezentată ca un colaj al unor părți ale acesteia iar metoda a fost denumită sistemul funcției iterate partiționate sau locale (PIFS – partitioned iterated function system). În prima fază se împarte imaginea în „blocuri de selecție”, de dimensiuni mai mici (RB range blocks), care vor forma o „colecție de coduri” (CB - code book). Se încearcă apoi găsirea unor „blocuri domenii”, mai mari, din imagine, (DB domain blocks) care să poată fi approximate cu RB, prin transformate afine și de intensitate. Schema de codificare fractală se prezintă în figura 3.17.

În figura 3.17., V_R reprezintă o transformare afină de simetrie (în jurul axei de simetrie verticală a DB, în cazul prezentat), W_R este o transformare de intensitate de tipul $y_{i,j} = \alpha x_{i,j} + \beta$, în care $y_{i,j}$ este intensitatea pixelului (i,j) după transformare iar $x_{i,j}$ înainte de transformare, α este parametrul de contrast iar β parametrul de luminozitate. Ultima transformare aplicată este o scalare R_R , care de obicei are valoare de 0,75, rezultând un bloc de dimensiune egală cu cea a RB. Se compară apoi DB transformat cu RB și se decide, pe baza unui criteriu de apropiere (distanță) dacă aproximarea este acceptabilă sau nu. Se repetă algoritmul pentru toate DB din imagine. Pentru decompresie trebuiesc reținute toate RB utilizate și V_R , W_R și R_R aferente. Cu cât numărul de RB ce descriu imaginea este mai mic, cu atât compresia este mai bună. Reducerea timpului de căutare face obiectul unor cercetări intense în

care se studiază soluții prin reducerea la problema celui mai apropiat vecin [CARDINAL1], prin utilizarea similarității dintre codificarea fractală și wavelet [DAVIS2], prin îmbinarea celor două metode (wavelet și fractal) [LI1] sau prin utilizarea căutării adaptive [TONG1].

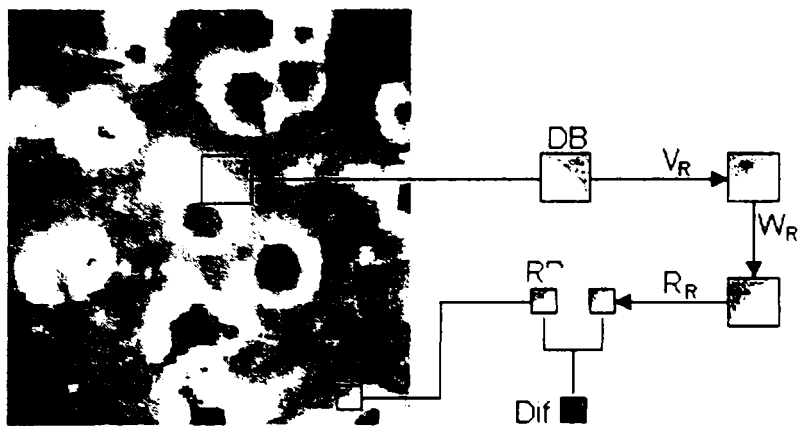


Fig.3.17. Schema codificării fractale.

Analizând din punct de vedere statistic, parametrii utilizați la codificarea fractală (de exemplu cu cât se aplică mai puține transformări de simetrie cu atât parametrul „simetria texturii” are o valoare mai scăzută), se obțin caracteristici (simetria texturii, finețea texturii, uniformitatea spațială a texturii) cu valori specifice fiecărei imagini, ce pot fi utilizați pentru extragerea muchiilor sau regiunilor imaginii și implicit la recunoașterea unor obiecte sau texturi [SCHOUTEN1]. În lucrare se urmărește aplicarea acestor concepte la căutarea după conținut a imaginilor într-o bază de date.

A doua categorie de aplicații consacrate a fractalilor în prelucrarea de imagini este caracterizarea imaginii sau a unor părți ale acesteia în funcție de dimensiunea fractală.

Un obiect fractal, matematic, este generat în cadrul unui algoritm iterativ, prin copierea de un număr N de ori a unui obiect redus, la fiecare iterație, cu o anumită scară r . Obiectul fractal astfel obținut poate fi caracterizat prin dimensiunea fractală (dimensiunea Hausdorff-Besicovitch, exponenții Renyi [GORSKY1]), și în forma cea mai simplă este definită ca raportul dintre logaritmul lui N și logaritmul lui r . Cu ajutorul definiției se pot calcula dimensiunile fractale ale unor obiecte generate matematic precis. Pentru imagini reale însă, în cadrul procesării computerizate, se aplică algoritmi de calcul cum ar fi : metoda numărării casetelor (box counting method) [GORSKY1] și metoda masei cumulative (CM - cumulative mass method) [HENRY1, MORENCY1]. La ora actuală există o serie de programe comerciale ce utilizează calculul dimensiunii fractale pentru caracterizarea unor obiecte din imagini [IDOLON1], ce pot fi folosite în diverse domenii de la biologie [HENRY1] la imaginile unor formațiuni geografice naturale sau artificiale [MORENCY1, AGU1]. La începutul utilizării fractalilor a existat un entuziasm exagerat ce a generat ideea generalității absolute a aplicabilității acestora și a metodelor de calcul a dimensiunii fractale. După un studiu aprofundat însă, tot mai mulți cercetători [GORSKY1,

HORNBOGEN1] au arătat că metodele fractale trebuie utilizate cu precauție și că mai rămân de clarificat o serie întreagă de probleme.

În [GORSKY1], de exemplu se arată că metoda numărării casetelor și derivatele acesteia dau rezultate false în unele cazuri dând dimensiuni fracționare pentru seturi a căror dimensiune calculată analitic este 0, astfel de obiecte fiind denumite pseudo-fractali.

Tot aici se prezintă și o clasificare a obiectelor în următoarele categorii :

- fractali matematici;
- fractali fizici;
- pseudo-fractali;
- non-fractali.

În [AGU1] se consideră că imaginile unor formațiuni geografice pot conține mai multe structuri fractale, fiecare dintre ele având propria dimensiune fractală. Utilizând conceptul dimensiunii generalizate, se dezvoltă o metodă de decelare a dimensiunilor fractale aferente fiecărei structuri.

În metalografie, metodele fractale au fost studiate încă de la apariția noțiunii de fractal. În 1984, Mandelbrot și coautorii, publicau un articol în revista „Nature” despre caracterul fractal al suprafețelor de rupere (fracturi) ale metalelor. De atunci, au existat diferite încercări pentru a demonstra caracterul fractal al microstructurilor metalice, cum ar fi de exemplu [HINOJOSA1], în care se utilizează numărarea casetelor pentru caracterizarea structurilor dendritice ale unui aliaj de aluminiu-siliciu.

O analiză mai amplă și mai obiectivă este realizată în [HORNBOGEN1] în care se arată că aplicarea geometriei fractale în analiza microstructurilor, comportă trei aspecte importante :

- *analiza fractală* – măsurarea experimentală a caracteristicii fractale (dimensiune Hausdorff-Besicovitch, exponenți Renyi) și a intervalului de auto-similaritate prin metode microscopice sau numerice (numărarea casetelor, CM).
- *sinteza fractală* – definirea unor noi forme de geometrice cu ajutorul funcțiilor iterative, servind ca model pentru generarea unor structuri reale;
- *relații microstructură fractală – proprietăți*, în care se încearcă căutarea unor corelații între caracteristicile fractale ale microstructurii și proprietățile mecanice ale acestora.

În aceeași lucrare, fără a subestima aplicabilitatea metodelor fractale, se arată că multe dintre supozițiile a diverși autori în legătură cu caracteristicile fractale ale unor microstructuri nu au fost confirmate nici teoretic și nici experimental.

Analiza "wavelet".

Metoda wavelet este o metodă dezvoltată din analiza Fourier, ce se aplică la un domeniu larg de aplicații în prelucrarea semnalelor și imaginilor. Principalele aplicații

se referă la compresie (compresia imaginilor amprentelor digitale din baza de date FBI) și filtrare (filtrarea semnalelor EKG, EEG, Spectre de Rezonanță Magnetică). O descriere completă a utilizării wavelet în compresia imaginilor este dată în [DAVIS1].

Analiza Fourier este utilă pentru prelucrarea semnalelor și imaginilor în diferite aplicații (paragraful 3.4), dar această metodă frecvențială prezintă dezavantajul major al pierderii informațiilor locale (informațiile de timp în cazul semnalelor temporale și de spațiu în cazul imaginilor).

În 1946, Dennis Gabor a adaptat transformarea Fourier, pentru analiza unei ferestre din semnal care se deplasează de-a lungul coordonatei timp, obținând astfel o serie de spectre ce oferă și o aproximare a caracteristicilor locale. Această tehnică a fost denumită „transformată Fourier pe timp scurt” (STFT – Short Time Fourier Transform), sau „reprezentare timp-frecvență”. Această metodă stă la baza unor studii de cercetare în scopul segmentării texturilor prin construirea unor filtre Gabor optimale [TSAI1]. Problema de bază a acestui tip de transformare este dimensiunea fixă a ferestrei de scanare.

Analiza wavelet reprezintă următorul pas în dezvoltarea metodelor de analiză prin utilizarea unor tehnici ce sunt similare utilizării unei ferestre de dimensiuni variabile. Sinteza dezvoltării analizei wavelet este dată în figura 3.18.

Se poate observa că analiza wavelet utilizează o reprezentare timp-scară în loc de reprezentarea timp-frecvență.

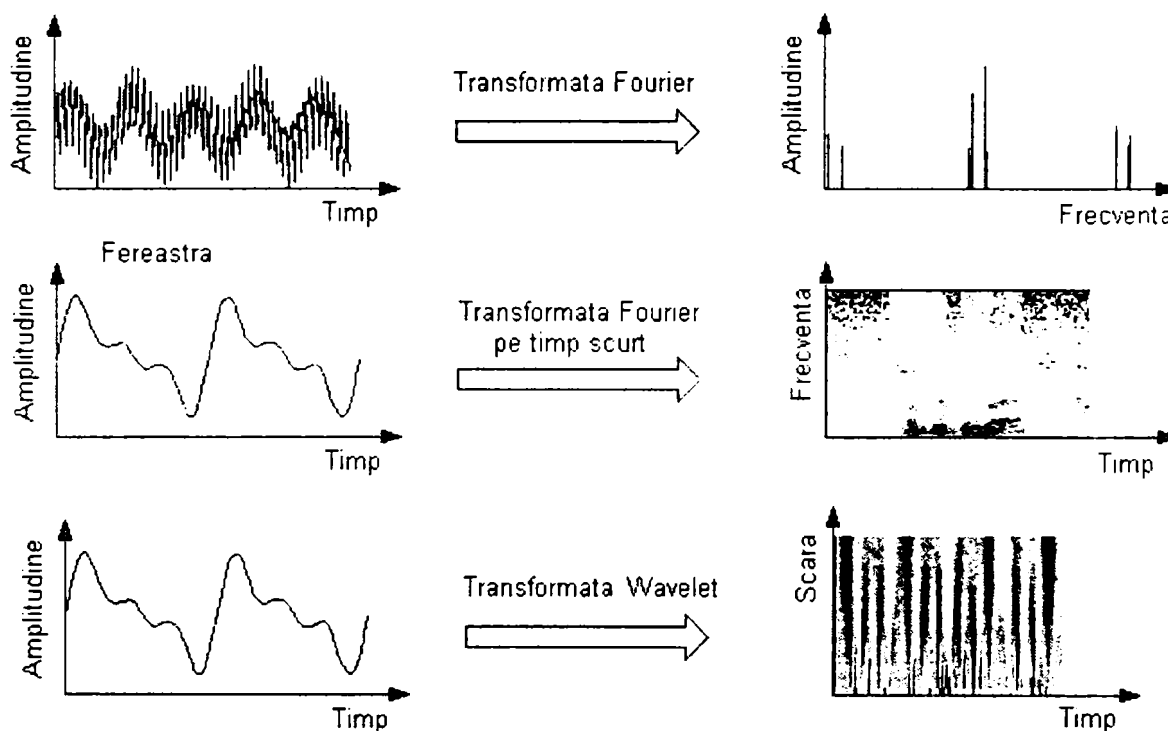


Fig.3.18. Sinteza dezvoltării analizei wavelet.

După cum se cunoaște, transformata Fourier descompune un semnal în sinusoides de diferite frecvențe. În mod similar transformata wavelet descompune un semnal în așa numita „wavelet de bază” (mother wavelet) și în variante scalate

(comprimate sau extinse în domeniul timp) și deplasate (tot în domeniul timp) ale acesteia.

Matematic transformata wavelet se poate defini ca :

$$C(a, k) = \int_{-\infty}^{\infty} f(t)\psi(a, k)dt \quad (3.77)$$

în care : $C(a, k)$ – sunt coeficienții wavelet; $f(t)$ – semnalul; ψ – funcția wavelet; a – scara; k – deplasarea în domeniul timp.

Alegerea funcției wavelet $\psi(a, k)$ are o importanță deosebită și depinde de aplicația concretă la care se utilizează. Un exemplu de funcție wavelet (Daubechies) este arătat în figura 3.19.

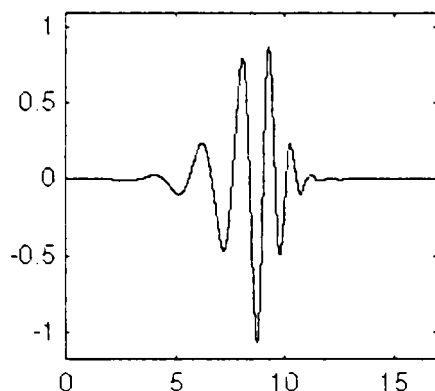


Fig.3.19. Reprezentarea grafică a funcției wavelet Daubechies de ordinul 7.

Există o serie de tipuri consacrate de funcții wavelet utilizate în special în aplicații de compresie și filtrare a semnalelor cum ar fi Haar, Daubechies, Meyer. Desigur pentru prelucrarea imaginilor există corespondente bidimensionale ale funcțiilor wavelet amintite. Proiectarea unor funcții wavelet specializate face în continuare obiectul unor studii ca de exemplu extragerea muchiiilor grăunților utilizând funcții wavelet direcționale [JOURNAUX1], estimarea și caracterizarea rugozității suprafețelor utilizând funcții wavelet spline biortogonale [XIAO1] sau transformări wavelet normalizate în frecvență [JOSSO1] și analiza suprafețelor tridimensionale [JIANG1].

O altă clasă de aplicații se concentrează asupra filtrării imaginilor și a eliminării zgomotelor [ZERVAKIS1].

Transformarea wavelet continuă (CWT), realizată pentru o variație continuă a lui a și k este dificil de utilizat în practică, din care cauză se utilizează transformata wavelet discretă (DWT) ce se realizează pentru valori ale lui a și k , puteri ale lui 2.

DWT se implementează printr-un algoritm foarte eficient (dat de Mallat în 1988) numit „codificator sub-bandă pe două canale” (two-channel subband coder) sau „transformarea wavelet rapidă” (fast wavelet transform).

Algoritmul DWT de bază constă în filtrarea semnalului (figura 3.20.a), ce are ca rezultat obținerea coeficienților wavelet a „aproximatei” semnalului (cA), corespunzând componentelor de frecvență joasă și a coeficienților wavelet a „detaliilor” acestuia (cD), corespunzând componentelor de frecvență înaltă.

Algoritmul de bază poate fi aplicat succesiv (figura 3.20.b), obținându-se mai multe trepte de aproximare. Se observă că după operația de filtrare se realizează o subeșantionare (se elimină fiecare al doilea coeficient obținut) pentru ca în final numărul de coeficienți după transformare să rămână același cu numărul de eșantioane ale semnalului inițial.

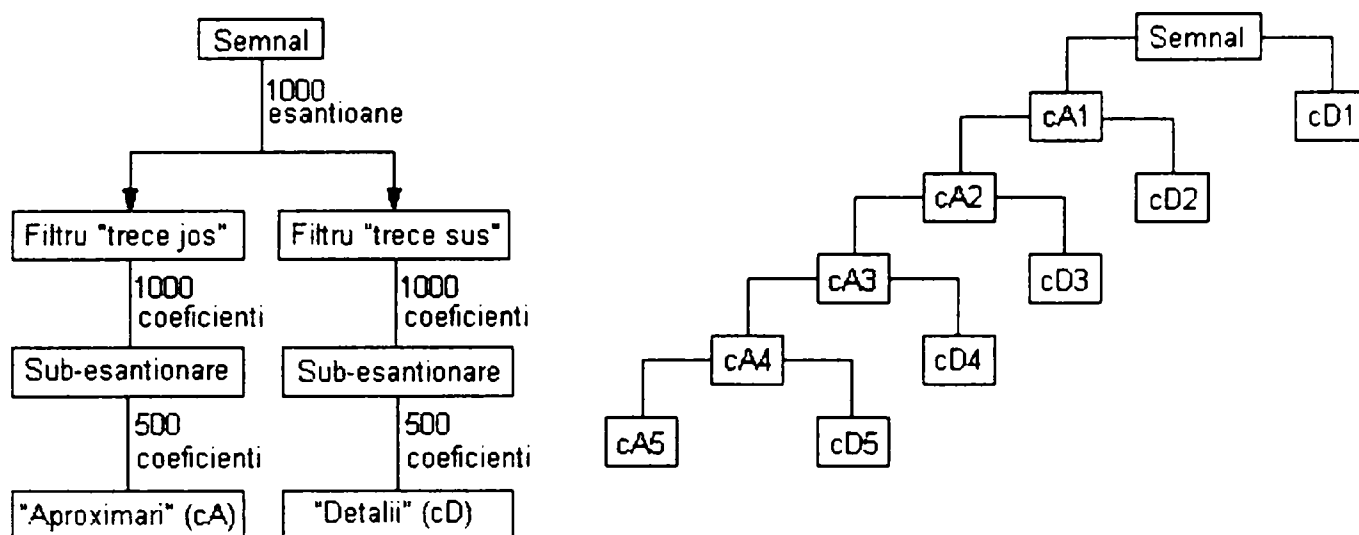


Fig.3.20. Schemele de implementare a transformatei wavelet :
a. Transformata wavelet pe un nivel; b. Transformata wavelet multi-nivel.

Extinderea algoritmului pentru cazul procesării imaginilor este arătat schematic în figura 3.21.

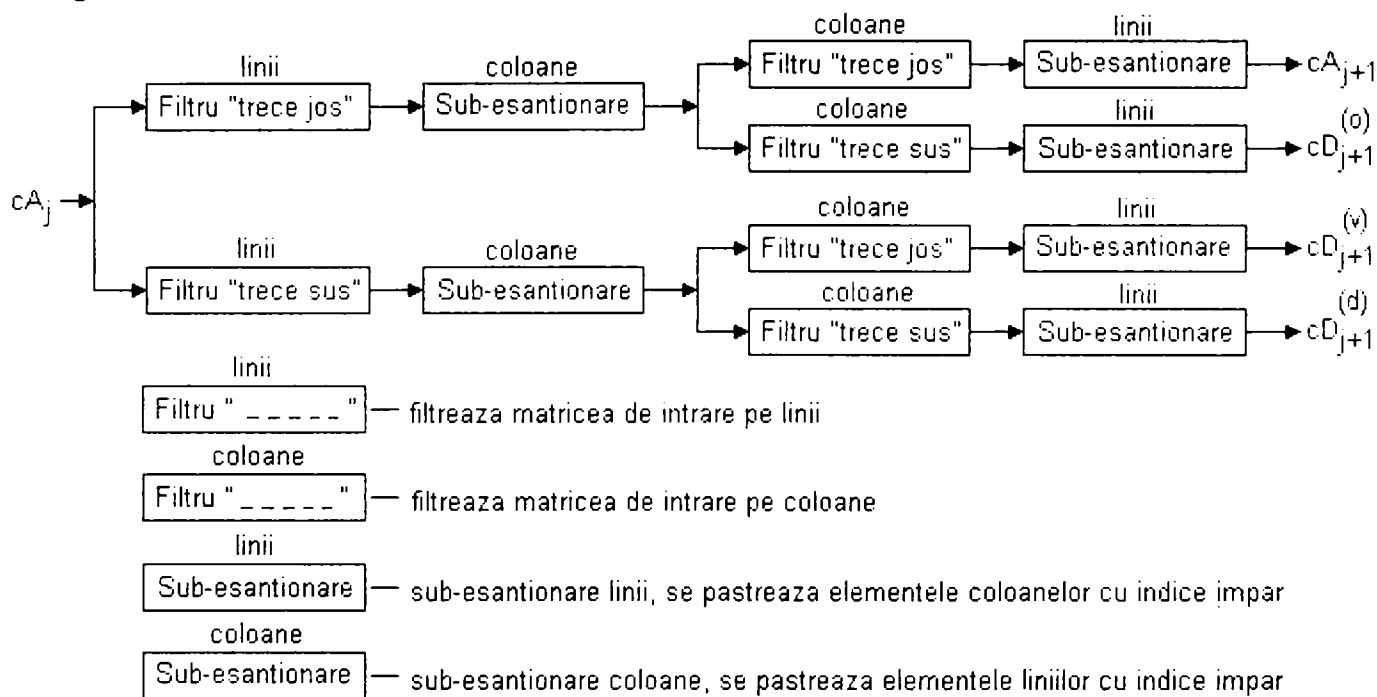


Fig.3.21. Reprezentarea schematică a unui nivel de transformare wavelet bi-dimensională.

Se consideră că imaginea inițială (I) este o matrice ce are ca valori nivele de gri. S-a notat cu j nivelul transformării, deci $cA_{(0)} = I$ iar cA_j este matricea coeficienților transformării pentru nivelul j . Rezultatele transformării wavelet pe

nivelul j vor fi : matricea coeficienților wavelet cA_{j+1} a aproximărilor; matricea coeficienților wavelet $cD_{j+1}^{(o)}$ a detaliilor după direcție orizontală; matricea coeficienților wavelet $cD_{j+1}^{(v)}$ a detaliilor după direcție verticală; matricea coeficienților wavelet $cD_{j+1}^{(d)}$ a detaliilor după direcție diagonală.

Datorită caracterului multi-scară (reprezentare scară–timp, pentru semnale temporale și respectiv scară–poziție pe linie–poziție pe coloană, pentru imagini), algoritmul wavelet este înrudit (sau mai bine zis poate fi aplicat aceluiași tipuri de aplicații) cu analiza fractală.

Analiza cu ajutorul algoritmilor genetici.

Algoritmii genetici (GA – „Genetic algorithms” în limba engleză) sunt metode stohastice de căutare similare conceptelor evolutive din biologia naturală [POHLHEIM1]. GA au ca subiecte, populații de indivizi (potențiale soluții ale unei probleme) pe care se aplică principiul „supraviețuirii celui mai apt” pentru a produce o din ce în ce mai bună aproximare a soluției căutate. La fiecare nouă generație se crează un nou set de aproximări (indivizi sau cromozomi) prin procesul de selectare și înmulțire (recombinare) similar cu cel întâlnit în genetica naturală. În cadrul procesului de căutare a soluției se utilizează concepte și metode ca : selecția, recombinarea, mutația, migrația, localizarea, vecinătatea. GA are un caracter de calcul paralel datorită faptului că lucrează cu populații de indivizi în loc de soluții singulare.

În figura 3.22. se prezintă structura unui algoritm genetic mono-populație.

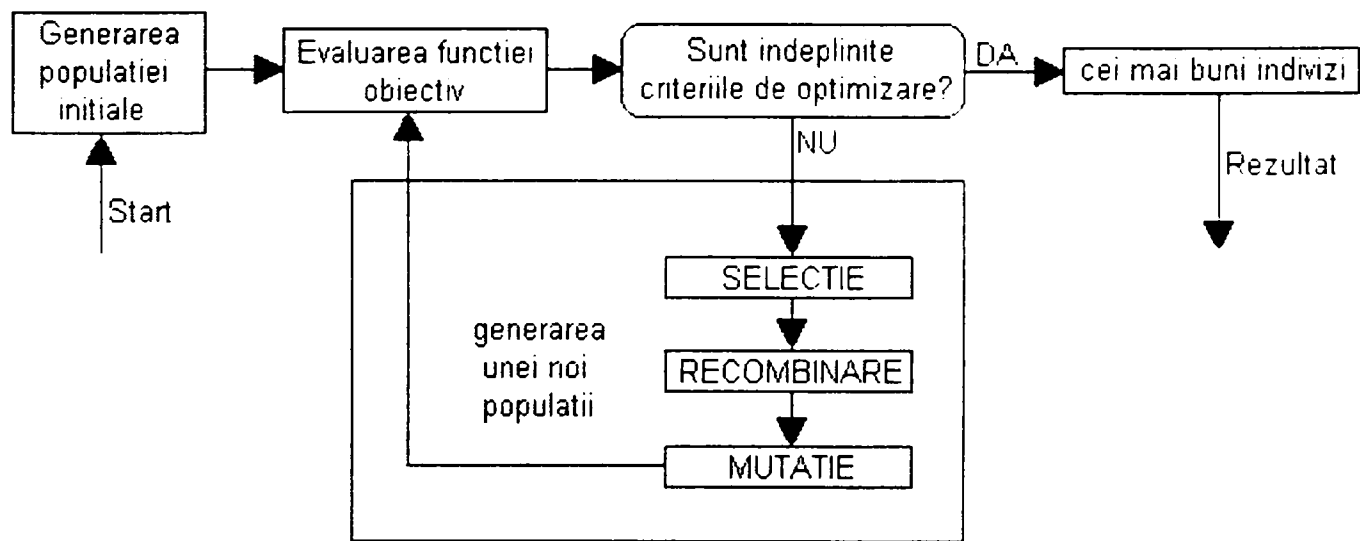


Fig.3.22. Structura GA mono-populație.

Au existat trei școli importante în privința elaborării acestor tipuri de algoritmi în ultimii 30 de ani :

- J.H.Holland (1975, USA), sub denumirea de algoritmi genetici;
- J. Rechenberg (1973, Germania), sub denumirea de strategii evolutive;

- H.P. Schwefel (1981, Germania), sub denumirea de programare evolutivă;

La începutul calculului se inițializează în mod aleator o anumită populație, după care se evaluează funcția obiectiv pentru indivizii din populația respectivă. Dacă criteriul de optimizare nu este satisfăcut, se generează o nouă populație, în mai multe etape :

- indivizii (cărora li s-a alocat un grad de aptitudine) sunt selectați pentru a produce urmași, de regulă fiind selectați indivizii cei mai apti;
- toți urmașii vor fi supuși unei mutații cu un anumit grad de probabilitate;
- se calculează gradul de aptitudine al urmașilor;
- urmași sunt reinserați în populație înlocuind părinții sau indivizii cu gradul de aptitudine cel mai mic, astfel va lua naștere o nouă generație;
- ciclul se repetă până la îndeplinirea criteriului de optimizare.

Deși GA mono-populație pot rezolva o serie de probleme complexe, există cazuri când soluțiile găsite au un caracter local. Rezultate mai bune s-au obținut prin utilizarea mai multor populații simultan, numite sub-populații, care sunt lăsate să se dezvolte izolat unele de altele, un anumit număr de generații, după care se fac schimburi de indivizi între populații (migrație).

Există diferențe esențiale între GA și alte metode de căutare și optimizare, dintre care se pot aminti :

- căutarea se realizează într-o manieră paralelă;
- GA pot furniza un număr de soluții potențiale alternative;
- GA utilizează reguli de tranziție probabilistice.

În [BRUMBY1] problema abordată este de a găsi o combinație optimă de operații elementare de prelucrare a imaginilor (dilatate, erodare, filtrare), în scopul analizei unei imagini specifice. În acest sens, șirul de operații este tratat ca un „cromozom” iar operațiile în sine sunt „genele” acestuia. În cursul evoluției mai multor generații vor lua naștere cromozomi din ce în ce mai adaptați unor prelucrări complexe cum ar fi extragerea caracteristicilor unei imagini (extragerea muchiilor sau a regiunilor). Criteriul de optimizare adoptat este distanța Hamming dintre caracteristicile extrase cu ajutorul cromozomilor și caracteristicile țintă.

În cadrul CIMI Laboratory (Computer Integrated Medical Intervention) [CIMI1] s-a elaborat un program ce utilizează tehnici GA pentru extragerea caracteristicilor (contururilor unor organe) din imaginile ultrasonice bio-medicale.

În lucrarea [VENCES1], în afara aplicațiilor de extragerea caracteristicilor algoritmi genetici sunt utilizați și în aplicații de compresie a imaginilor în combinație cu metode fractale.

Analiza cu ajutorul automatelor celulare.

Automatele celulare (CA – Cellular Automata) sunt reprezentări matematice simple, abstractizate ale unor sisteme naturale [WOLFRAM1]. Ele se compun dintr-o latice de elemente (celule) discrete identice ce pot avea un număr finit de stări (valori

numerice întregi). Stările celulelor la un moment dat, evoluează în pași de timp discretizat, potrivit unor reguli (deterministe sau stohastice) ținând cont de starea celulelor vecine, a stării anterioare și a unor factori externi (ce pot simula un anumit mediu). Din punct de vedere matematic automatele celulare pot fi privite ca abstractizări discrete a ecuațiilor diferențiale cu derivate parțiale. Din cauza naturii lor discrete, automatelor celulare, pot fi privite ca și procesoare digitale de construcție simplă interconectate, capabile de procesare paralelă.

Dezvoltarea automatelor celulare a fost oarecum limitată în ani 1960-1980 de capacitatea calculatoarelor din aceea vreme. Actualmente însă s-au reformulat conceptele automatelor celulare, acestea fiind aplicate în cele mai diverse domenii ale modelării.

Ușurința cu care se pot aplica conceptele CA pentru prelucrarea imaginilor constă în faptul că imaginile digitale (formate din pixeli) se pot asocia cu laticia celulelor, corelând fiecare celulă cu un pixel (sau un grup de pixeli), și starea celulei cu nivelul de gri al pixelului respectiv. Evoluția dinamică a CA asociat imaginii reprezintă astfel o prelucrare a acesteia.

În sprijinul demonstrării aplicabilității conceptului CA în prelucrarea imaginilor se prezintă în continuare câteva lucrări din acest domeniu.

În [POPOVICI1] se tratează problemele binecunoscute ale filtrării imaginilor și a detectării conturului dar de această dată utilizând automate celulare (CA). Similar cu metodele bazate pe rețele neuronale, automatele celulare sunt realizate prin algoritmi paraleli, ceea ce le conferă o viteză de execuție superioară. Algoritmi prezentați utilizează vecinătăți de tip von Neumann ce s-au dovedit propice pentru scopul propus. Se prezintă deasemenea o analiză comparativă a filtrării CA cu un filtru gaussian și detectarea de contur CA cu un algoritm consacrat denumit SUSAN.

În [VARMA1] autorii au realizat un model de difuzie bidimensional care are capacitatea de a descrie transformarea austenitei în ferită și perlită în condițiile unei răcirii continue. Se presupune că germinarea feritei are loc la limita grăunților iar cea a perlitei în tot grăuntele și la interfața ferito-austenitică, atunci când condițiile de temperatură și concentrație sunt favorabile. Pentru descrierea creșterii feritei și perlitei se utilizează un algoritm bazat pe automate celulare. Rezultatele simulării au fost comparate cu fotografiile ale microstructurilor obținute experimental.

În [BASANTA1] se arată că, în principu, automatele celulare (CA) sunt caracterizate printr-o stare inițială și un set de reguli. În concepția clasică setul de reguli este fixat la pornirea simulării și nu este modificat pe parcursul acesteia. Autorii propun aplicarea unor algoritmi genetici care să realizeze generarea automată a setului de reguli, ceea ce conferă o șansă mai mare unei simulări a creșterii microstructurilor. Acest concept a fost aplicat pentru realizarea unor simulării utilizând automate efectoare (EfA), o subclasă de automate celulare a căror stare este reprezentată de poziția automatului în latică.

În [NAKANO1] se presupun cunoscute și bine definite muchiile regiunilor ce trebuie extrase dintr-o imagine. Fiecărui pixel din imagine i se atribuie o celulă.

Stările ce pot fi atribuite celulelor sunt : „ne-procesat”, „în curs de procesare” și „deja-procesat”. Utilizând proprietățile de paralelism a automatelor celulare se extrag regiuni din imagine, pixelii fiecărei regiuni trecând din starea „ne-procesat” în starea „deja-procesat”, până când se extrag toate regiunile din imagine. Avantajul metodei față de metoda clasică a etichetării regiunilor constă în posibilitatea aplicării ei pe un circuit digital cu procesare paralelă, măbind astfel viteza de prelucrare a imaginii.

În [SPITTLE1] autorii au realizat un model cu automate celulare bi-dimensional pentru studiul formării structurilor dendritice, ce pornește de la o stare inițială și aplică fiecărei celule un set de reguli stabilite empiric. Aceste reguli stabilesc comportamentul celulei pornind de la starea lichidă și condițiile solidificării (temperatură, saturație). Solidificarea unei celule implică și efectele acestui fenomen asupra celulelor deja solidificate cât și asupra celor în stare lichidă. Cu ajutorul modelului s-a obținut simularea creșterii dendritelor într-o manieră foarte apropiată de realitate.

Programul „Morphogenesis” al firmei ITK (membru al grupului de proiectare soft Kitware) [MORPHO1] realizează simulări complexe bazate pe conceptul CA. Programul realizează simularea comportamentului organismelor unicelulare (bacterii) prin utilizarea principiilor de bază a automatelor celulare cu seturi de reguli modificabile. Aplicațiile acestui gen de modelare în analiza imaginilor consideră imaginea ca un mediu de creștere al „bacteriilor” ce interacționează atât cu mediul cât și unele cu altele. Astfel diferitele culori ale imaginii sunt asociate cu medii ce facilitează sau stopează creșterea coloniilor. Se obțin astfel procesări similare cu morfologia și segmentarea, având avantajul unei procesări paralele.

Alte lucrări ce utilizează conceptul CA tratează subiecte ca simularea 3D a recristalizării în metale prelucrate la rece [MARX1], simularea dinamicii moleculare în cupru nanocristalin [BELAK1], separarea fazelor în prezența vacanțelor [MUKHERJEE1].

Analiza cu ajutorul rețelelor neuronale.

Datorită avantajelor pe care le prezintă rețelele neuronale artificiale (ANN Artificial Neural Networks), în variate domenii ale științei, există o serie de cercetări a aplicării acestora și în domeniul prelucrării imaginilor. ANN au fost testate în rezolvarea problemelor de recunoașterea formelor, cu rezultate promițătoare mai ales datorită abilității de clasificare, optimizare și selectare a caracteristicilor. Principiul de funcționare a ANN fiind descris în capitolul precedent, se prezintă aici câteva aplicații reprezentative în acest domeniu. Trebuie remarcat că în legătură cu aplicarea ANN în probleme specifice de metalografie, încă nu au fost publicate lucrări importante.

În [YUAN1] se arată că în principiu pentru recunoașterea obiectelor și estimarea situării acestora, în afară de o imagine 2D, trebuie utilizate informații suplimentare de adâncime a căror achiziție necesită resurse hard și soft considerabile. Caracteristicile imaginii obiectului analizat se compară de obicei cu caracteristici

dintr-o bază de date de mari dimensiuni ce stochează o mulțime de vederi model, pentru fiecare obiect în parte. În cazul utilizării ANN dimensiunile bazei de date sunt reduse considerabil, deoarece în timpul antrenării se rețin numai caracteristicile cele mai relevante, și pe lângă aceasta ANN fiind deosebit de robuste nu este necesară memorarea unui număr mare de vederi model.

În domeniul recunoașterii și clasificării formelor cu ajutorul ANN au fost realizate studii pentru utilizarea unor seturi de rețele (fiecare dintre acestea urmărind să obțină a altă caracteristică) ce reușesc performanțe mai bune decât o singură rețea [GIACINTO1]. Rezultatele clasificării obținute de fiecare ANN din cadrul ansamblului sunt apoi analizate în vederea elaborării deciziei finale. În aceste studii se arată că performanțele ansamblului de rețele sunt acceptabile numai atunci când ANN componente dau erori independente unele de altele (dacă mai multe ANN componente dau aceeași eroare, decizia finală poate fi eronată). Pentru a elabora componente ANN ce dau erori independente se propune generarea unor seturi mai largi dintre care apoi să se selecteze un ansamblu ANN mai restrâns dar care posedă proprietățile dorite.

În scopul realizării unui asistent automat pentru conducătorii auto s-a realizat un sistem ANN [WOHLER1] de analiză în timp real a imaginilor ce poate detecta trei cazuri distincte : urmărirea unui autovehicul țintă, recunoașterea laturii șoselei și recunoașterea pietonilor ce traversează șoseaua. În acest sens intrarea ANN cuprinde imagini în diferite secvențe de timp și realizează analiza acestora fără a efectua o extragere de caracteristici (de exemplu pietonii sunt recunoscuți pe baza mișcării specifice a picioarelor în timpul mersului). Prezentarea la intrarea rețelei a secvențelor de imagini simultan duce la realizarea unei caracteristici specifice spațio-temporale.

Prelucrarea imaginilor medicale este o aplicație importantă utilizată la stabilirea diagnozelor clinice. În diferite studii s-au încercat diverse metode de segmentare cu rezultate mai mult sau mai puțin satisfăcătoare. Printre acestea utilizarea ANN de tip Hopfield datorită limitărilor arhitecturale, poate produce doar performanțe similare cu binarizarea după histogramă. Pentru mărirea performanței segmentării cu ANN Hopfield, s-a conceput o nouă arhitectură spațială de forma unui cub [CHANG1] în care stratul de bază este stratul de intrare, ce primește ca valori imaginea de prelucrat (fiecare pixel al imaginii pe o intrare), iar straturile următoare (pe verticală) vor conține la sfârșitul procesării, câte un obiect extras din imaginea inițială. Studiul comparativ al acestei metode față de altele indică performanțe demne de luat în considerare.

Rețele neuronale celulare (CNN- Cellular Neural Network) CNN reprezintă un tip aparte de ANN, în care rețeaua neuronală celulară de bază este o structură bidimensională rectangulară fiind formată din circuite analogice neliniare, dispuse regulat, numite celule [GACSADI1]. Structurii bidimensionale a CNN de bază i se poate asocia o imagine, fiecare celulă corespunzând unui pixel. Starea fiecărei celule depinde de starea inițială, o lege de tranziție și starea pixelilor vecini. Mai multe rețele de bază suprapuse pot forma o structură multistrat. Datorită posibilității de

implementare ușoară pe circuite analogice, CNN oferă o viteză de prelucrare foarte rapidă a imaginii. Cu CNN s-au experimentat toate operațiile de bază a prelucrărilor de imagini (îmbunătățire, filtrare, morfologie), cu rezultate bune.

Sisteme expert fuzzy.

După cum se arată în sinteza metodelor fuzzy, prezentată în capitolul precedent, acest algoritm este deosebit de utilizat în luarea deciziilor atunci când este posibilă definirea unui set de reguli. Algoritmul fuzzy nu se aplică în mod direct la procesarea imaginilor dar este utilizat cu succes în procesul de recunoaștere și interpretare [HILTNER1]. În această lucrare se prezintă utilizarea sistemelor expert fuzzy și a metodelor lingvistice pentru analiza imaginilor de rezonanță magnetică (MRT - Magnetic Resonance Tomograph) în domeniul medical. Pentru aceasta se utilizează atât descrierea bazată pe model cât și cea lingvistică, baza de cunoștințe fuzzy fiind consultată în fiecare fază a analizei imaginii. Schema de principiu a metodei este dată în figura 3.23.

Pentru descrierea lingvistică a obiectelor din imagine se definesc diverse caracteristici : „circularitate”, „elongație”, „suplețe”, ce se utilizează apoi la generarea gradelor lingvistice fuzzy și a setului de reguli.

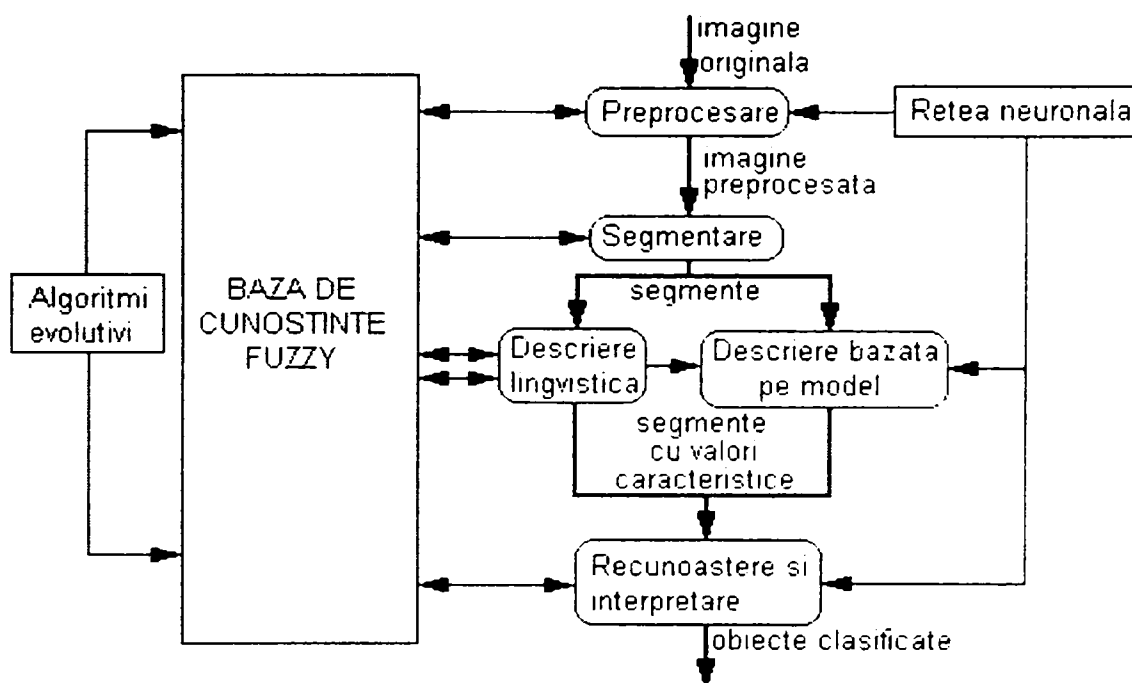


Fig.3.23. Sistem expert fuzzy, pentru recunoașterea imaginilor, aplicat în domeniul medical.

4. Sisteme automatizate pentru laboratoare.

4.1. Dezvoltarea sistemelor automatizate pentru laboratoare.

Celulele flexibile au avut o dezvoltare intensă în ultimele decenii, mai ales în domeniul industriei constructoare de mașini. Dezvoltarea automatizării controlului calității (în care se încadrează analiza metalografică) nu a urmat în totalitate aceeași tendință, datorită complexității operațiilor ce trebuie realizate.

Mai recent însă, în acest domeniu, s-au obținut rezultate remarcabile, mai ales în aplicații privind automatizarea laboratoarelor chimice, biologice și medicale și în mai mică măsură cele de știința materialelor și metalografie.

Conform rapoartelor ALA (Association for Laboratory Automation) apărute recent, domeniul automatizării laboratoarelor a generat, în anul 2000 în întreaga lume, peste 1,2 miliarde de dolari profit și s-a estimat o creștere cu 15% a cifrei de afaceri, în următorii ani.

Principalele motive pentru automatizarea proceselor din laboratoare sunt :

- creșterea nevoilor de testare și descoperire a materialelor noi, respectiv a produselor noi în domeniul farmaceutic și biologic;
- eliminarea erorilor umane;
- urmărirea unui număr mare de eșantioane;
- eliminarea contaminării eșantioanelor (mai ales în laboratoarele biologice și cele pentru nanomateriale).

Odată cu apariția cerințelor de dezvoltarea a laboratoarelor automate, în domeniul medical, s-a inițiat, de către o serie de instituții din domeniu ALA, LRIG (Laboratory Robotics Interest Group), și în domeniul laboratoarelor pentru medicină LAPCT (Laboratory, Automated, and Point of Care Testing) un standard de automatizare specific. Acest standard este prevăzut să se includă în specificațiile pentru comunicarea informațiilor și declanșarea evenimentelor din domeniul medical HL7 (Health Level Seven) versiunea 3x., ca o secțiune separată de Automatizare a Laboratoarelor (Lab Automation).

Având în vedere cifra de afaceri remarcabilă realizată de acest domeniu al industriei, pe lângă firmele mici specializate numai pe instalații și echipamente de laborator, o serie de firme de renume mondial în domeniul roboticii (KUKA, FANUC, SIEMENS) s-au implicat în producția de roboți și echipamente specializate pe operații de laborator, mai ales în domeniul biologic, medical și nanomateriale. O importanță deosebită a avut automatizarea laboratoarelor în domeniul geneticii la stabilirea genoamelor ce necesită analiza unui număr deosebit de mare de eșantioane.

În domeniul științei materialelor și a metalografiei nu s-au înregistrat dezvoltări atât de spectaculoase, laboratoarele automatizate în acest domeniu reprezentând doar 10% din totalul pe ramură. În ceea ce privește laboratoarele metalografice situația actuală este cea prezentată în capitolul 2 și în paragraful următor, neexistând la ora

actuală sisteme care să integreze atât operațiile de pregătire a eșantioanelor cât și cele de analiză.

4.2. Realizări în domeniul laboratoarelor automatizate.

După cum se poate constata din documentația de specialitate, laboratoarele automatizate se pot împărți în două categorii :

- laboratoare proiectate pentru o sarcină specifică, utilizate într-un domeniu îngust;
- laboratoare proiectate modular, cu o flexibilitate ridicată.

Ca exemple pentru prima categorie, din domeniul biologic, se poate aminti laboratorul automat pentru analiza cărnii dezvoltat de către firma 'DANSK TEKNOLOGI' (figura 4.1) iar în domeniul științei materialelor, laboratorul pentru pregătirea eșantioanelor și analiza cu raze X realizat de către firma 'Labman' (figura 4.2). În domeniul testării substanțelor chimice organice se poate da ca exemplu sistemul realizat de către firma „BECKMAN-COULTER” (figura 4.3). Aceste laboratoare au o structură ce permite doar un flux rigid, executând operații prestabilite ce pot varia într-un domeniu foarte restrâns. Acest tip de laboratoare se pretează pentru domenii în care trebuie analizat un volum mare de eșantioane într-un timp relativ scurt.

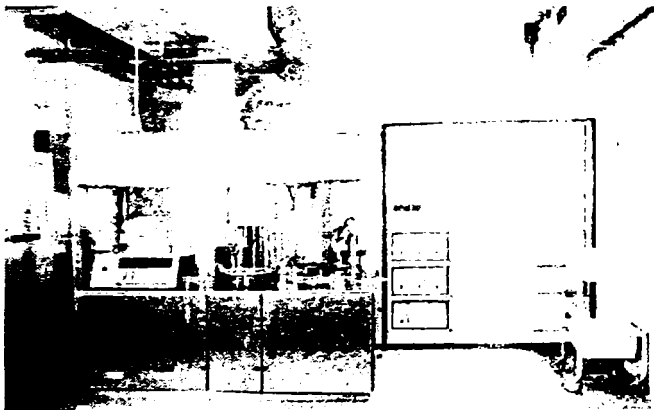


Fig.4.1. Sistem automat pentru analiza cărnii.



Fig.4.2. Sistem automat pentru pregătirea și cântărirea eșantioanelor.

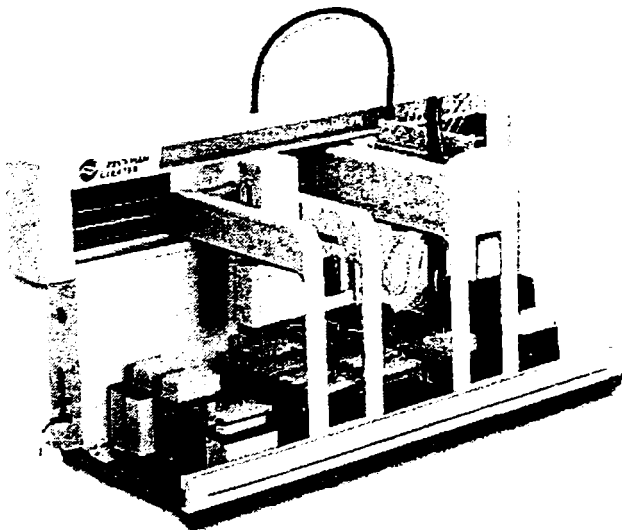


Fig.4.3. Sistem automat pentru analiză chimică.

Pentru a doua categorie furnizorii de echipamente oferă module de manipulare, transport, stocare și prelucrare într-o mare varietate. Ca exemplu sunt prezentate în figurile 4.4.,4.5. și 4.6., produsele firmei „CRS”. Concepția modulară permite utilizarea acestor produse într-o gamă largă de aplicații asigurând în plus o flexibilitate structurală ce ușurează trecerea de la un flux de prelucrare la altul.



Fig.4.4. Module de manipulare (roboți).



Fig.4.5.. Module de manipulare integrate cu diverse dispozitive.

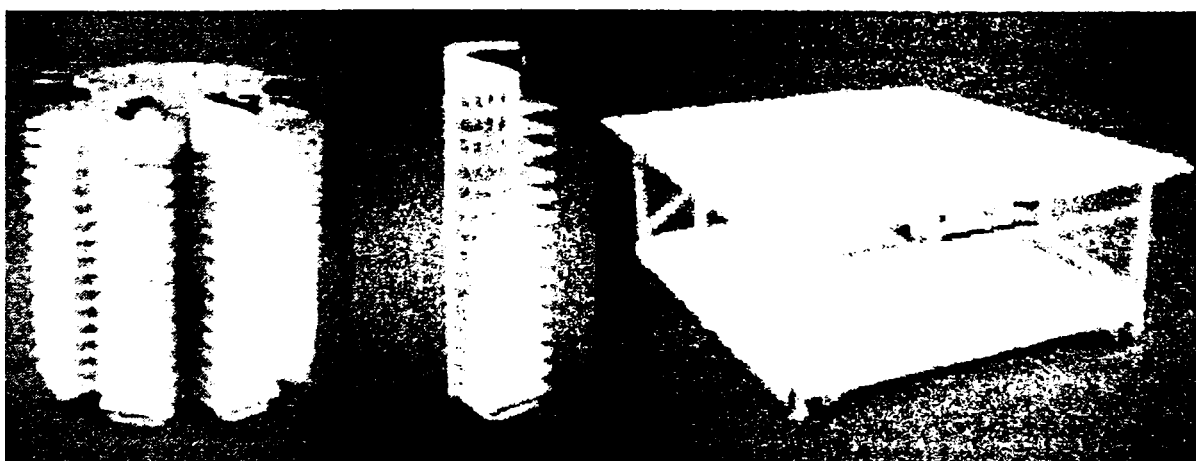


Fig.4.6. Module de stocare a eșantioanelor.

În domeniul metalografiei robotizarea nu a avut o dezvoltare la fel de rapidă ca în celelalte domenii, deși există tendințe clare în acest sens. Un exemplu concludent îl constituie produsele de vârf ale firmei „Clemex” (Canada), care la sfârșitul anilor 1990 oferea sisteme ce constau dintr-un microscop echipat cu cameră CCD, masă poziționabilă prin teleoperare și soft aferent (figura 4.7.) iar în anul 2004 produce sisteme ce integrează un robot pentru manipularea eșantioanelor și un post de stocare a acestora (figura 4.8.).

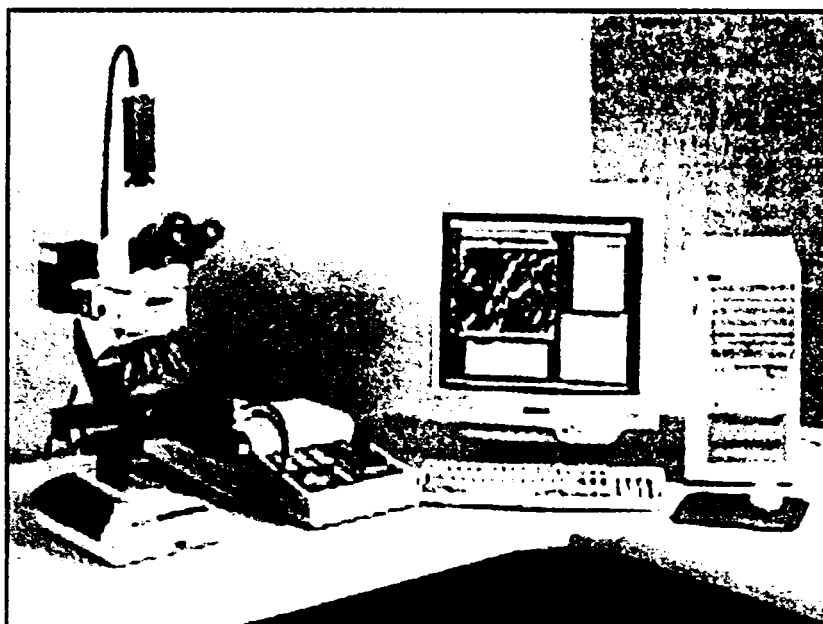


Fig.4.7. Sistemul Clemex IMPAK, 1997.

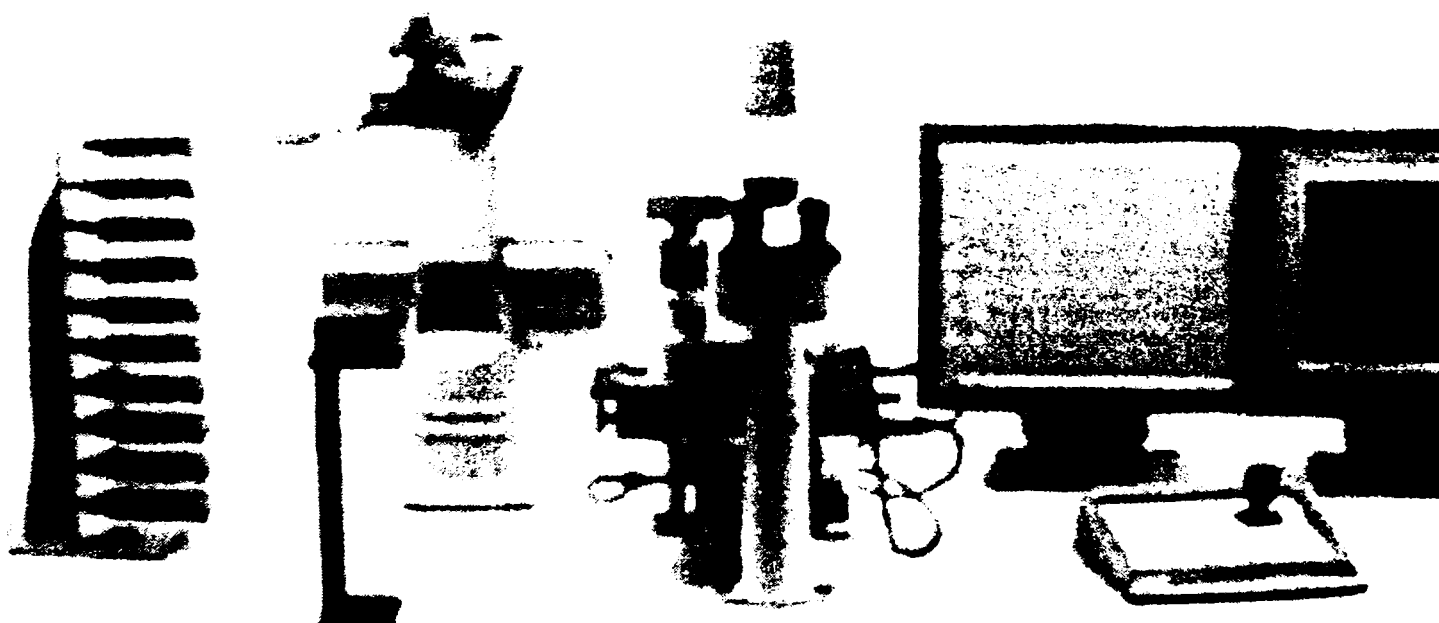


Fig.4.8. Sistemul Clemex robotizat , 2004.

4.3. Concluzii

Față de laboratoarele automatizate din domeniul biologic, medical, chimic, laboratoarele metalografice prezintă câteva deosebiri esențiale prezentate în tabelul 4.1.

Tabel.4.1.

Caracteristica	Domeniu	
	Biologic, chimic, medical	Metalografie
Număr de eșantioane	mare	medie, mică
Diversitatea operațiilor	mică, medie	mare
Necesitatea de eliminare a inter-contaminărilor	ridicată	medie

Din analiza caracteristicilor rezultă că pentru laboratoarele metalografice se pretează în primul rând conceptele modulare ce permit o flexibilitate ridicată. Pe de altă parte o serie de proceduri și instalații utilizate în domeniul bio-medical pot fi luate ca model pentru rezolvarea problemelor în domeniul metalografic. Un exemplu în acest sens sunt sistemele de manipulare și procesare chimică. Experiența acumulată în acest domeniu poate fi utilizată la rezolvarea operațiilor de atac cu reactiv și spălare. De asemenea procedurile de evitare a contaminării eșantioanelor pot fi preluate și în cadrul laboratoarelor metalografice, cu restricții mai puțin severe, pentru operațiile de șlefuire în care există pericolul contaminării cu praf abraziv și în cazul operațiilor de atac chimic în care se utilizează soluții acide.

5. Celula flexibilă pentru laborator metalografic (CFLM1)

5.1. Considerații generale

Din studiul posibilităților de automatizare a operațiilor de pregătire și analiză metalografică rezultă că există o mare diversitate de metode prin care se obțin informațiile asupra microstructurii eșantioanelor. Cele mai avansate sisteme în domeniu [CLEMEX1], la ora actuală, rezolvă doar parțial sarcinile de analiză metalografică, manipularea și stocarea eșantioanelor, fără a aborda sarcinile de pregătire a acestora. În cadrul tezei de doctorat s-a studiat posibilitatea realizării sarcinilor de pregătire cu pondere ridicată și a sarcinilor de analiză într-un sistem flexibil automatizat. Desigur, sistemul propus nu rezolvă integral problematica pregătirii eșantioanelor și analizei metalografice dar prin aplicarea conceptului de flexibilitate (structurală și informațională), pe care îl integrează, constituie un model ce poate fi dezvoltat și extins.

Din multitudinea de sarcini de procesare ce pot să apară în cadrul laboratorului metalografic au fost alese trei, ce pot fi privite ca „sarcini generice” : șlefuirea, atacul cu reactivi și analiza metalografică. Studiul rezolvării sarcinilor generice și a cuplărilor dintre acestea, generează o bază de cunoștințe deosebit de utilă pentru aplicațiile viitoare.

5.2. Schema de amplasament a celulei flexibile.

Celula flexibilă de pregătire și analiză a eșantioanelor metalografice a fost concepută luând în considerare documentațiile prezentate în capitolele 2 și 3, cât și resursele tehnico-materiale existente în laboratorul de Mecatronică al Universității din Oradea. Schema de amplasament a celulei concepute este prezentată în figura 5.1.

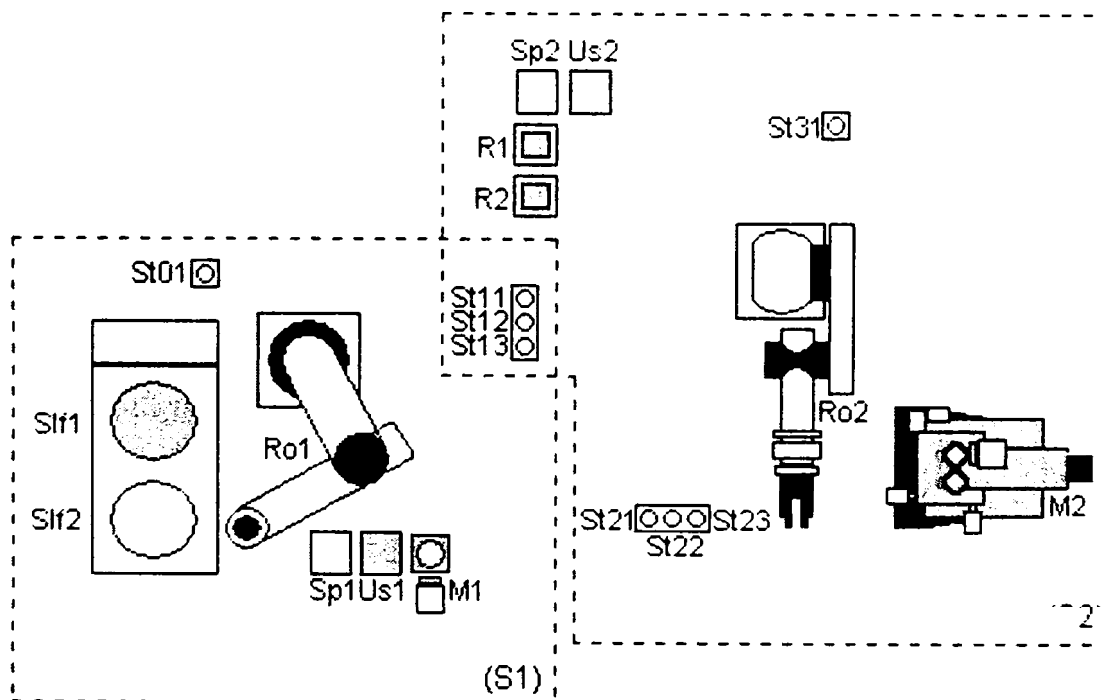


Fig.5.1. Schema de amplasament a celulei flexibile.

Notațiile utilizate în figura 5.1 sunt :

- Slf1,Slf2 – posturi de șlefuire a eșantioanelor;
- St01,St11,St12, St13,St21,St22, St23,St31 – stocatoare;
- Ro1,Ro2 – roboți industriali;
- Sp1,Sp2 – posturi de spălare a eșantioanelor;
- Us1,Us2 – posturi de uscare a eșantioanelor;
- R1,R2 – posturi de atac cu reactivi;
- M1,M2 – posturi de determinare a stării suprafeței și respectiv a structurii metalografice;

Din punct de vedere al deservirii de către roboți, celula se compune din două subsisteme :

- subsistemul (S1) ce cuprinde posturile : Slf1, Slf2, Sp1, Us1, M1, St01 deservite de robotul Ro1;
- subsistemul (S2) ce cuprinde posturile : R1, R2, Sp2, Us2, M2, St21, St22, St23, St31 deservite de către robotul Ro2.

Eșantionul este introdus în sistem prin intermediul stocatorului St01.

Etapele de pregătire a eșantioanelor constă din operațiile de șlefuire, determinarea stării suprafeței și atac chimic cu reactiv. Această etapă este divizată între cele două subsisteme șlefuirea și determinarea stării suprafeței realizându-se în subsistemul (S1) deservit de Ro1 iar atacul cu reactivi în subsistemul (S2) deservit de Ro2. În subsistemul (S2) deservit de Ro2 se realizează în totalitate etapa de analiză metalografică în vederea determinării microstructurii eșantionului.

Șlefuirea unui eșantion se realizează în mai mulți pași succesivi. Prelucrarea efectivă este urmată întotdeauna de operația de spălare și uscare a eșantionului realizată la posturile Sp1, respectiv Us1. Eșantionul prelucrat este apoi transferat la postul de determinare a stării suprafeței (M2). Dacă suprafața eșantionului nu corespunde parametrilor ceruți atunci prelucrarea acesteia va fi reluată. Ciclul prelucrare - verificare este reluat până când suprafața eșantionului este prelucrată în mod corespunzător.

Stocatoarele St11, St12, St13 au rolul de interfață între cele două subsisteme.

Odată ce eșantionul este prelucrat corespunzător el este depus de către robotul Ro1 pe unul din aceste stocatoare de unde este preluată de către robotul Ro2 și introdusă în al doilea subsistem.

După transferul eșantionului în al doilea subsistem (S2), se realizează atacul cu reactiv la unul din posturile R1 sau R2 după care întotdeauna se realizează operația de spălare și uscare la posturile Sp2 și respectiv Us2.

Etapele finale a evoluției eșantionului în celula flexibilă o reprezintă analiza metalografică realizată la postul M2. Stocatoarele St21, St22, St23 au rolul de stocare a eșantioanelor în cazul în care postul M2 este ocupat. Dacă postul M2 este liber, eșantionul este transferat direct la acest post și se realizează analiza metalografică.

După analiza metalografică eșantionul părăsește celula prin intermediul postului de stocare St31.

„Layout”-ul celulei flexibile a fost conceput în așa fel încât să permită accesul roboților Ro1 și Ro2 la posturile pe care le deservește, aferente

subsistemului (S1) respectiv (S2).

5.3. Descrierea funcțională a celulei flexibile.

5.3.1. Schema de ansamblu a celulei.

Pentru elaborarea schemei de ansamblu s-au avut în vedere ca principii :

- respectarea layout-ului stabilit anterior, ținând cont de spațiul desemnat pentru celula flexibilă;
- utilizarea sistemelor și echipamentelor existente în laboratorul de Mecatronică al Universității din Oradea;
- posibilitățile de adaptare a echipamentelor existente în scopul realizării funcțiilor celulei flexibile.

Schema de ansamblu a celulei este prezentată în figura 5.2.

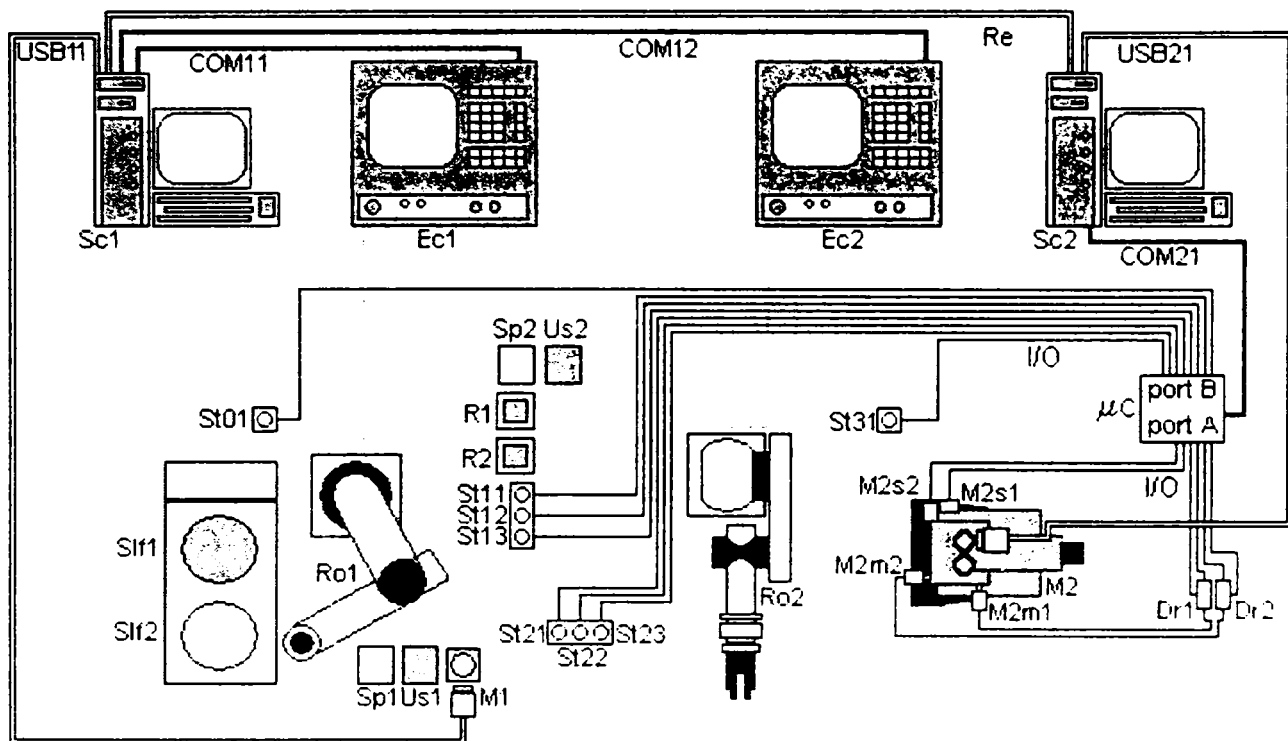


Fig.5.2. Schema de ansamblu a CFLM1.

Pe lângă posturile de lucru și roboți, în schema de ansamblu (figura 5.2) au fost reprezentate și sistemele de calcul și echipamentele de comandă cât și elementele de acționare ale postului M2:

- Sc1, Sc2 – sisteme de calcul;
- Ec1, Ec2 – echipamentele de comandă ale roboților Ro1 și Ro2;
- μ C – micro calculator;
- Dr1, Dr2 – circuite de comandă motoare pas cu pas (MPP);
- M2m1, M2m2 – MPP pentru poziționarea mesei microscopului;
- M2s1, M2s2 – microlimitatoare poziție schimbare eșantion (PSE).

Fluxul de informații între diferitele elemente ale celulei este realizat prin următoarele canale de comunicare:

- Re – rețea locală LAN Windows2000;

- USB11,USB21 – canale de comunicare USB;
- COM11,COM12,COM21 – canale de comunicare serială RS232;
- I/O – intrări și ieșiri digitale μ C.

În figura 5.2 se observă că posturile Slf1,Slf2,Sp1,Us1,R1,R2,Sp2,Us2 nu sunt controlate de către sistemele de calcul.

Posturile de șlefuire Slf1, Slf2 sunt concepute să funcționeze neîntrerupt pe timpul funcționării celulei deoarece frecvența de utilizare a acestor posturi este mare și nu se justifică pornirile și opririle repetate ale discului de șlefuire.

Posturile de spălare și uscare execută în principiu procesări simple : robotul aduce și menține eșantionul în poziția de procesare ce constă în spălarea eșantionului cu un jet de apă sau uscarea acestuia cu jet de aer cald.

În ambele cazuri oprirea jetului se poate realiza printr-o temporizare la nivelul controlerului postului sau oprirea automată în momentul în care robotul îndepărtează eșantion din poziția de procesare.

În ultimul caz se poate utiliza ca element de comandă al pornirii/opririi procesării, un simplu micro-întrerupător mecanic, senzor optic sau un senzor de proximitate, temporizarea fiind specificată la nivelul echipamentului de comandă al robotului.

Pentru posturile de atac cu reactiv etapele procesării sunt aceleași ca cele menționate anterior, dar procesarea constă în aplicarea unei pelicule de reactiv pe suprafața eșantionului sau imersarea parțială a acesteia în reactiv. Pe durata staționării postului, reactivul trebuie să fie izolat ermetic, deci va trebui să existe un dispozitiv specific, o trapă sau un capac etanș cu închidere și deschidere automată. Deschiderea și închiderea capacului se poate realiza tot cu ajutorul unui senzor de proximitate.

În consecință se consideră că declanșarea procesării și oprirea acesteia la posturile de spălare, uscare și atac cu reactiv este realizată de robot (la apropierea și respectiv depărtarea acestuia de poziția de procesare) și nu este necesară utilizarea unor echipamente de comandă cuplate la sistemele de calcul.

Totuși, în cadrul programului de comandă al CFLM1, posturile amintite sunt tratate în aceeași manieră ca și posturile ce sunt comandate de sistemele de calcul, fapt ce conferă programului un caracter de generalitate.

5.3.2. Schema fluxului eșantioanelor în cadrul celulei flexibile.

CFLM1 a fost concepută pentru studiul diverselor situații ce apar în cadrul procesării eșantioanelor. La fel ca în cazul definirii structurii celulei și în cazul stabilirii itinerariilor tehnologice s-a stabilit un „flux tehnologic generic” care să cuprindă toate elementele majore ce pot să apară în decursul procesării eșantioanelor (ramificații, cicluri de revenire) ce dau posibilitatea studierii unui număr cât mai mare de situații. „Fluxul tehnologic generic” ales pentru studiu este reprezentat schematic în figura 5.3.

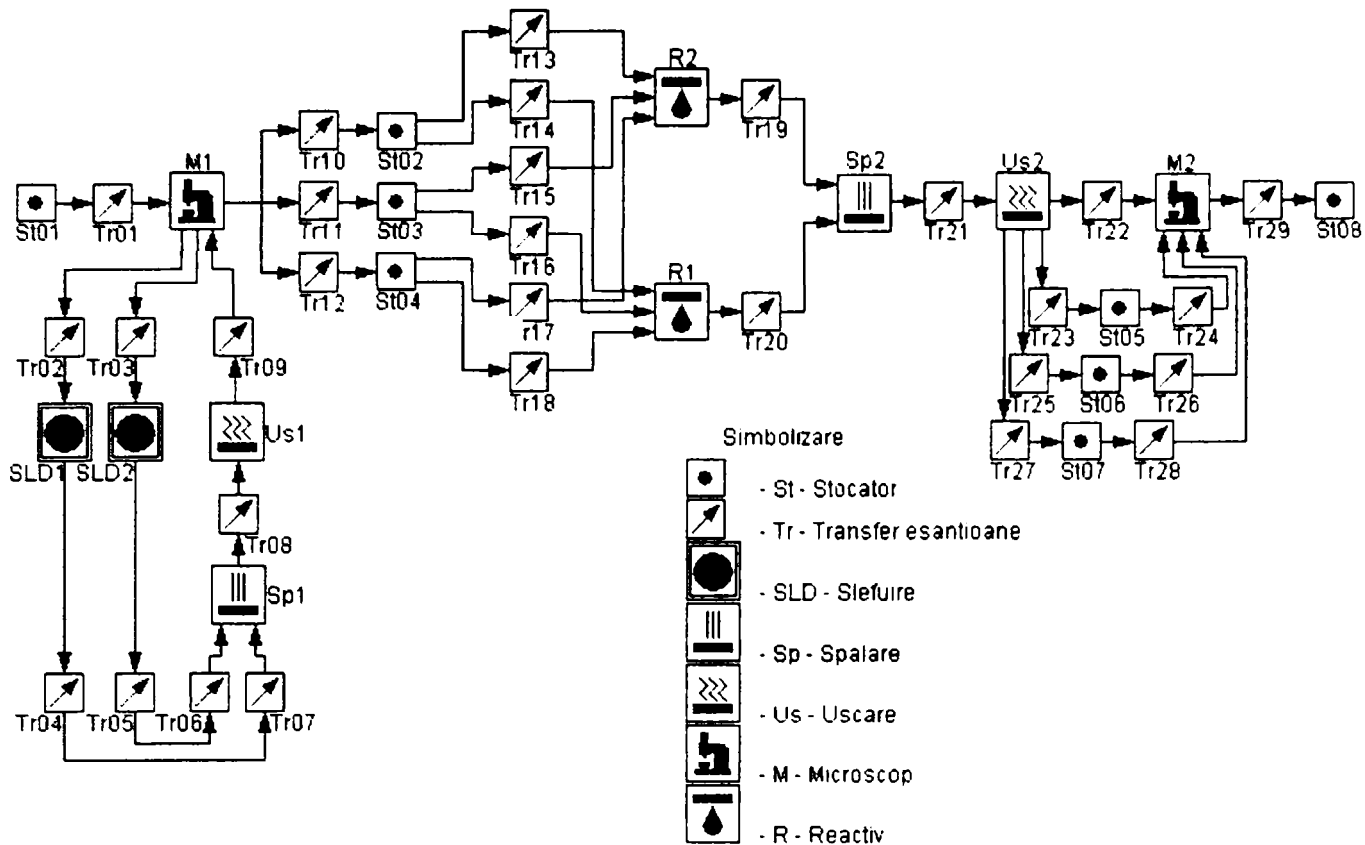


Fig.5.3. Schema „fluxului generic” al eşantioanelor în cadrul celulei.

5.4. Componente și subsisteme ale celulei flexibile.

5.4.1. Structura ierarhică a subsistemelor componente ale celulei flexibile.

Componentele CFLM1 au fost integrate într-o structură ierarhică (figura 5.4.) stabilită în urma estimării capacității sistemelor de calcul Sc1 și Sc2, de a comanda procesele aferente acestora.

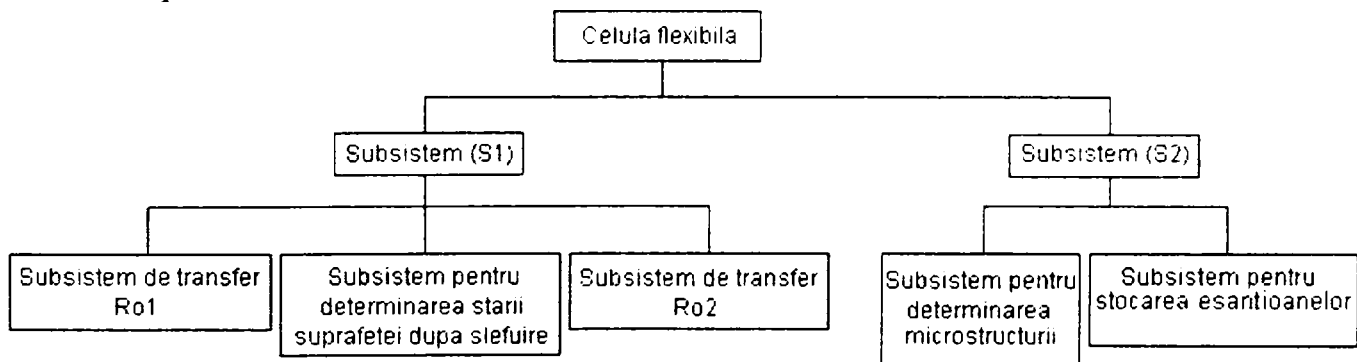


Fig.5.4. Schema ierarhică a componentelor celulei.

Pornind de la această structură s-a elaborat și structura informațională a celulei stabilindu-se distribuția programelor pe cele două sisteme de calcul. În continuare sunt descrise caracteristicile concrete a subsistemelor definite.

5.4.2. Subsistemul de transfer Ro1

Este concretizat prin robotul A.I.D-H5 (firma G.T.Robotique). Este un robot de tip „SCARA”, cu 4 axe (RRTR) (figura 5.5), acționat de motoare de curent continuu prin intermediul unor reductoare armonice frontale. Fiecare axă este echipată cu traductor incremental rotativ (sau liniar pe axa 3) și un tahogenerator. Robotul nu este dotat cu dispozitiv de prehensiune ceea ce a determinat necesitatea proiectării și realizării unui dispozitiv de prehensiune adecvat atât manipulării eșantioanelor cât și fixării acestora în cadrul operațiilor de șlefuire. Robotul este dotat cu un echipament de comandă numerică tip CNC EN5 (produs de Cynum-Industrie S.A.). Robotul poate lucra în regim autonom programele fiind lansate de operator sau în regim „RS-ON” când toate comenzile acceptate de către echipament pot fi transmise de la un sistem de calcul prin intermediul interfeței seriale. Această facilitate este utilizată și la funcționarea în cadrul CFLM1.

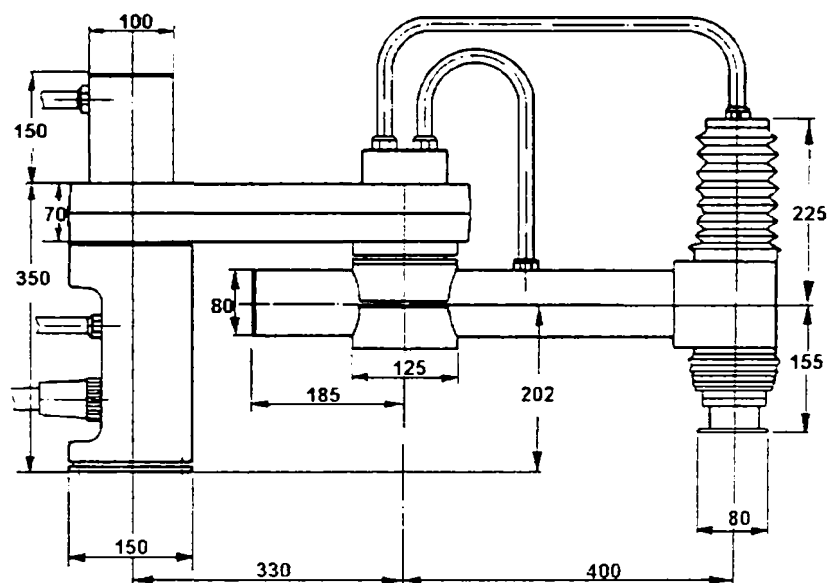


Fig.5.5. Schema structurii mecanice a robotului AID-H5

Echipamentul CNC lucrează cu format fix de fraze program, organizate în secvențe de câte 12 fraze. În fiecare frază există posibilitatea programării :

- coordonatelor exprimate în incrementi;
- procentului din viteza maximă;
- temporizărilor;
- contorizării ciclurilor;
- condițiilor interne și externe;
- numărului secvenței următoare.

Echipamentul posedă o capacitate de memorie de 99 de secvențe (127Ko) și o memorie tampon în care este transferată secvența ce urmează să fie executată.

5.4.3. Subsistem pentru determinarea stării suprafeței după șlefuire.

Este realizat dintr-un microscop „Citival” (mărire maximă x100) dotat cu o cameră CCD „Logitech”. Acest post realizează achiziția imaginii suprafeței și transmiterea acestei imagini la sistemul de calcul Sc1. În momentul poziționării

eșantionului în poziția de procesare, de către robot, se realizează achiziția imaginii și salvarea acesteia în memoria sistemului de calcul. După achiziția imaginii sistemul de calcul lansează programul de procesare imagine, care în final returnează rezultatul procesării ce va fi folosit pentru stabilirea acțiunilor ulterioare: transferul eșantionului la postul de șlefuire sau la postul de atac cu reactiv.

5.4.4. Subsistemul de transfer Ro2 (AID V6).

Subsistemul are în componența sa un robot tip A.I.D.V6 (firma G.T.Robotique) și echipamentul de comandă identic cu cel al robotului AID – H5. Structura robotului este de tip RRRRRR. Acționările robotului sunt similare cu cele ale robotului AID-H5. În figura 5.6. este reprezentată schematic structura mecanică a robotului AID-V6.

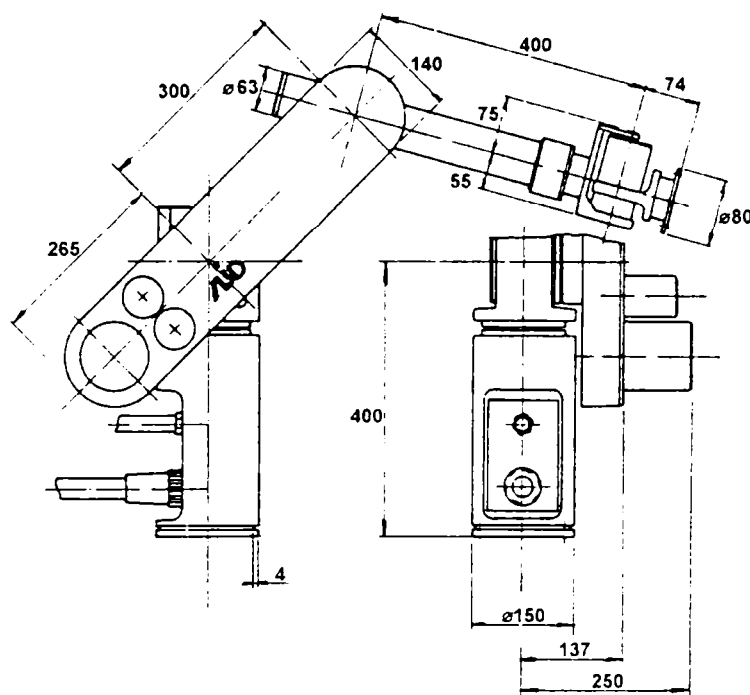


Fig.5.6. Schema structurii mecanice a robotului AID-V6

5.4.5. Subsistem pentru determinarea microstructurii.

Componenta de bază a subsistemului este microscopul metalografic MC2 (producție I.O.R.). Microscopul a fost modificat pentru a corespunde cerințelor de funcționare în cadrul CFLM1 (figura 5.7). Pe ocularul microscopului s-a montat o cameră CCD (firma Trust) cuplată la sistemul de calcul Sc2 prin portul USB. Pentru o gamă largă de tipuri de analiză metalografică este necesar studiul mai multor imagini prelevate din câmpuri vizuale diferite ceea ce implică necesitatea unor poziționării în regim automat a eșantionului. Pentru realizarea acestui deziderat masa microscopului a fost echipată cu două motoare pas cu pas (Mp1 și Mp2), comandate, prin interfețele Dr1 și Dr2 de un micro-calculator 485 (firma Blue-Earth, MICRO-485), ce comunică cu Sc2 prin intermediul unui port serial RS232. La cererea de inițiere a unui ciclu de achiziție, se selectează și se lansează

un program aplicație în funcție de tipul de analiză specific eșantionului. Ciclul de achiziție constă din diverse poziționări ale eșantionului corelate cu captarea imaginii în fiecare poziție. La terminarea ciclului de analiză trebuie poziționată masa microscopului în așa fel încât robotul să aibă acces la eșantion. Confirmarea atingerii poziției de schimbare eșantion (PSE) se realizează cu ajutorul a două micro-contactoare (M1 și M2). Odată ce s-a terminat ciclul de achiziție, este lansat programul de procesare a imaginilor (program MATLAB) ce rulează pe Sc2.

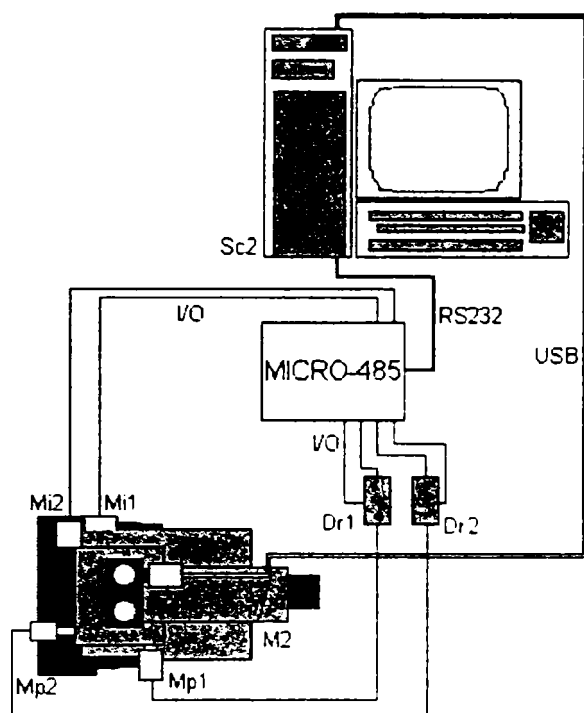


Fig.5.7. Schema de ansamblu a postului de analiză a microstructurii.

5.4.6. Subsistem pentru stocarea eșantioanelor.

Stocarea eșantioanelor se realizează pe posturi distincte echipate cu un micro-întrerupător ce semnalizează prezența eșantionului. În cadrul CFLM1 s-au utilizat 8 astfel de stocatoare, organizate în patru grupe, în funcție de rolul lor :

- grupa de intrare: stocator01;
- grupa de interfață dintre subsisteme: stocator11, 12, 13;
- grupa de depozit tampon: stocator21, 22, 23;
- grupa de ieșire: stocator31.

În cazul în care în decursul experimentărilor apar blocaje ale fluxului de eșantioane pot fi introduse stocatoare suplimentare.

Stocatoarele sunt conectate la un port de intrări numerice ale micro-calculatorului 485, prezența sau absența eșantioanelor fiind detectabilă în cadrul programului de comandă al celulei.

5.5. Descrierea celulei flexibile sub aspect informațional.

5.5.1. Structura informațională a celulei

Structura informațională s-a stabilit în concordanță cu structura ierarhică

prezentată în figura 5.4. Pentru realizarea acestei structuri (figura 5.8) este nevoie de trei medii de programare :

- la nivelul sistemelor de calcul;
- la nivelul echipamentelor de comandă al roboților;
- la nivelul micro-calculatorului.

Mediul de programare al micro-calculatorului (un limbaj BASIC extins) și al roboților (limbajul echipamentului CNC-EN5) sunt date de către echipamentele utilizate. Pentru sistemele de calcul există mai multe opțiuni, dintre care s-a optat pentru mediul de programare MATLAB® [MATLAB2], deoarece :

- este mediul care a fost utilizat și în celelalte capitole ale tezei de doctorat;
- este un mediu des utilizat în rezolvarea unor probleme de cercetare în automatizări;
- conține o bibliotecă bogată pentru probleme de prelucrare de imagini;
- are toate facilitățile de comunicare cu echipamentele externe utilizate;
- codul MATLAB se poate traduce ușor în cod C++.

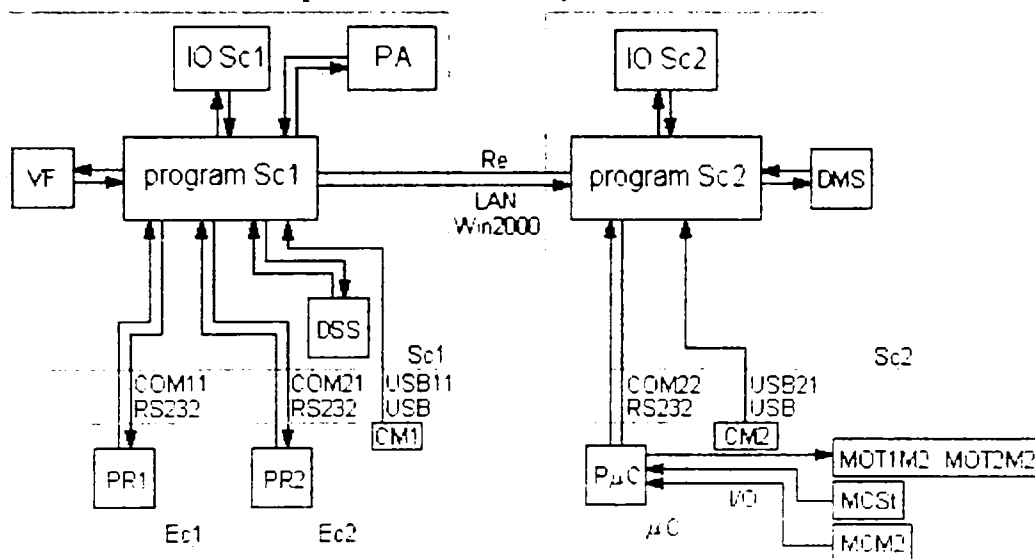


Fig.5.8. Schema structurii informaționale a CFLM1.

Notățiile utilizate în figura 5.8., sunt :

- IO Sc1 - interfața operator sistem de calcul 1;
- program Sc1 - program de comandă a celulei pe sistemul de calcul 1;
- PA - program de aplicație celulă;
- VF - program pentru vizualizarea fluxului de eșantioane;
- DSS - program pentru determinarea stării suprafeței;
- PR1 - program robot 1;
- PR2 - program robot 2;
- CM1 - camera CCD M1 (microscop pentru determinarea stării suprafeței);
- IO Sc2 - interfața operator sistem de calcul 2;
- program Sc2 - program de comandă a celulei pe sistemul de calcul 2;
- DMS - program pentru analiza microstructurii;
- PμC - program micro-calculator MICRO-485;
- CM2 - camera CCD M2 (microscop pentru analiza microstructurii);
- MOT1M2 – motor 1 (axa X) microscop M2;

- MOT2M2 - motor 2 (axa Y) microscop M2;
- MCM2 – micro-contactori poziție de schimbare eșantion axa X și Y (microscop M2);
- MCSt – micro-contactoare stocatoare.

CFLM1 a fost conceput în scopul experimentării robotizării laboratoarelor metalografice ce au un număr mic de angajați. Se consideră că operatorul de celulă este un angajat al laboratorului metalografic cu cunoștințe restrânse în domeniul programării. Din această cauză s-a conceput un limbaj specific pentru comanda celulei flexibile (limbajul programelor de aplicație), cu un număr mic de instrucțiuni, dar care să poată utiliza toate facilitățile existente. Programele scrise în acest limbaj sunt traduse în mod automat în cod MATLAB cu ajutorul unui program preprocesor.

5.5.2. Limbajul programelor de aplicație dezvoltat pentru CFLM1.

5.5.2.1. Introducere.

Limbajele utilizate pentru scrierea programelor de aplicație a celulelor flexibile pot fi concepute ca limbaje nestructurate, structurate sau structurate orientate pe obiecte. Alegerea tipului de limbaj utilizat în cadrul CFLM1 s-a realizat ținând cont de următoarele criterii :

- ușurința cu care pot fi scrise și corectate programele de aplicație :
 - lista instrucțiunilor să fie cât mai scurtă;
 - buclele de program să fie ușor de urmărit;
 - să existe un set cât mai mic de caractere speciale.
- reutilizarea liniilor de cod :
 - posibilitatea declarării subrutinelor.
- codul să fie ușor interpretabil în corespondență cu elementele fizice existente în CFLM1;
 - elementele fizice ale celulei să fie reprezentate în mod univoc în cadrul programului de aplicație;
 - logica programului de aplicație să fie în concordanță cu logica proceselor fizice ce se desfășoară în cadrul celulei.

Analizând criteriile prezentate s-a optat pentru un limbaj structurat, orientat pe obiecte.

5.5.2.2. Conceptul programării orientate pe obiecte utilizat în cadrul programelor de aplicație CFLM1.

5.5.2.2.1. Considerații generale.

Universul fizic al CFLM1 constă din obiecte fizice (reale) cu proprietăți specifice ce pot interacționa între ele. Scopul interacțiunii obiectelor este, în final, obținerea informațiilor asupra microstructurii eșantioanelor analizate. Interacțiunea obiectelor este comandată și controlată de către programul de aplicație al celulei. În

cadrul programului de aplicație sau în cadrul modulelor pe care acesta le apelează, trebuie să se regăsească într-o anumită formă informații asupra proprietăților obiectelor fizice și asupra posibilităților de interacțiune dintre acestea. Proprietățile obiectelor fizice se pot modifica în urma interacțiunii dintre ele, fapt care, de asemenea trebuie să se reflecte în modificări ale informațiilor la nivelul programului de celulă. Ca urmare, pentru reprezentarea informațiilor legate de obiectul fizic, în cadrul programului de aplicație, este util să se definească obiecte abstracte. Aceste obiecte, în unele surse documentare [MMS], mai sunt numite și *obiecte virtuale* (sau VMF - Virtual Manufacturing Device). Un obiect virtual este reprezentarea abstractă a obiectului fizic, în sensul că posedă proprietățile esențiale ale acestuia. Pentru fiecare obiect virtual, în afară de proprietăți, se stabilesc și operațiile ce pot fi efectuate cu acesta în cadrul programului de aplicație. Corespondența dintre obiectele virtuale și cele fizice trebuie să fie univocă și permanentă : dacă în decursul proceselor ce decurg în CFLM1 obiectele fizice își modifică unele proprietăți, se vor modifica și proprietățile obiectelor virtuale corespunzătoare și viceversa. Astfel, programul de aplicație inițiază interacțiuni între obiectele virtuale ce au ca rezultat interacțiuni între obiectele fizice din cadrul CFLM1, iar rezultatul interacțiunilor din universul real se reflectă în universul virtual. Universul virtual este inclus într-un univers mai larg definit ca univers informațional.

Este de asemenea util ca obiectele reale și în mod similar cele virtuale să fie grupate în clase sau categorii, deoarece în acest mod obiectele ce fac parte din aceeași categorie pot fi tratate în același mod, având aceleași proprietăți și fiind capabile de aceleași interacțiuni.

5.5.2.2.2. Clase de obiecte.

Clasele de obiecte au fost definite în urma analizei structurii CFLM1, și a stabilirii proprietăților și interacțiunilor esențiale ale obiectelor. Clasificarea obiectelor din cadrul CFLM1 este prezentată în figura 5.9.

Categoria „Transfer” include roboții (ROBOT) ce îndeplinesc funcția de manipulare a eșantioanelor. Posturile de lucru pot fi divizate în două subcategorii : „Stocare” (STO) și „Procesare”. Posturile de stocare sunt posturi pasive ce nu acționează asupra eșantionului iar cele de procesare sunt posturi ce modifică într-un anumit fel proprietățile eșantionului. Interacțiunea dintre robot și stocator se rezumă la deplasarea eșantionului la stocator și la încărcarea și descărcarea stocatorului.

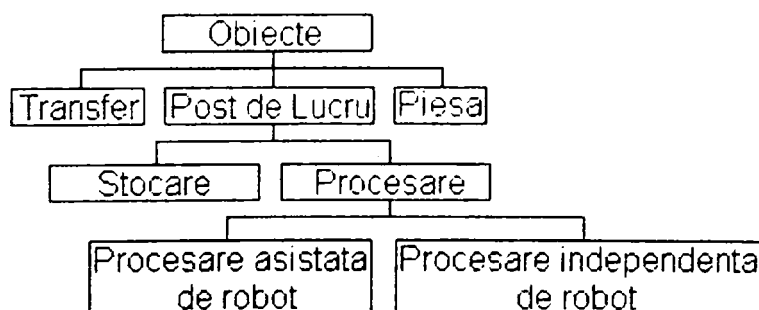


Fig.5.9. Diagrama clasificării obiectelor în cadrul CFLM1.

Din punctul de vedere al interacțiunii Robot – Post de Lucru este utilă divizarea posturilor de procesare în două subcategorii : „Procesare asistată de robot” (PAR) și „Procesare independentă de robot” (PIR).

În cazul categoriei PAR robotul execută următoarele mișcări :

- deplasare la postul de lucru;
- apropiere de poziția de procesare;
- mișcări prin care robotul ajută la procesare;
- mișcări de depărtare din poziția de procesare.

Simultan cu mișcările de asistență ale robotului, postul de lucru realizează sarcina de procesare.

Interacțiunile ROBOT - PIR sunt similare cu cele ROBOT – STO și anume deplasare la PIR și încărcare/descărcare PIR, în intervalul de timp dintre încărcare și descărcare, robotul fiind liber să execute alte sarcini. Postul de lucru PIR este diferit de postul STO, prin faptul că modifică proprietățile eșantionului iar descărcarea postului PIR nu se poate realiza până când nu s-a terminat sarcina de procesare, pe când în cazul stocatorului descărcarea se realizează fără nici o restricție.

Din structura de tip arbore a categoriilor arătată în schema din figura 5.9 se definesc ca și clase, categoriile terminale, obținând împărțirea obiectelor pe clase prezentată în figura 5.10.

În cadrul CFLM1 pot exista mai multe instanțieri ale unei clase, care diferă doar prin valoarea proprietăților lor, de exemplu clasa stocare (STO) conținând obiectele : „stocator01”, „stocator11”, ..., „stocator31”.

În figura 5.10 au fost reprezentate doar obiectele ce se supun principiului corespondenței dintre obiectele reale și virtuale. Fac excepție de la acest principiu obiectele clasei „Decizie” (DEC). Obiectele DEC nu sunt obiecte virtuale deoarece nu au corespondență în universul real. Aceste obiecte există doar în universul informațional (figura 5.11.) și au fost concepute ca obiecte (structuri ce au proprietăți și interacționează cu alte obiecte) pentru o tratare unitară a structurilor în cadrul programelor de aplicație.

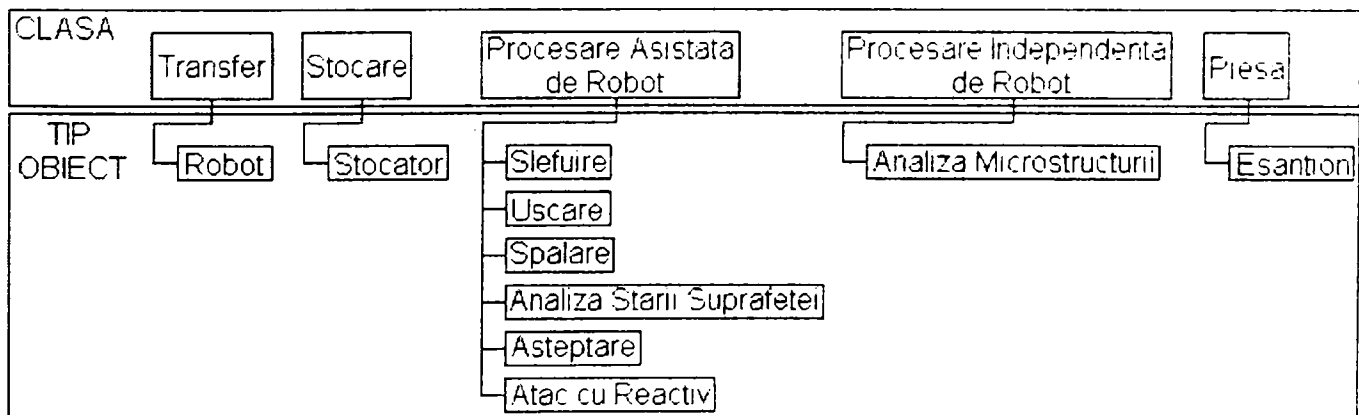


Fig.5.10. Diagrama claselor de obiecte virtuale.

Clasa DEC este implicată în aplicarea unor algoritmi de decizie „on-line” (modulele program „decizie”) ce utilizează un set de date inițiale și informații despre stările sistemului în momentul luării deciziei.

În cadrul CFLM1 s-au definit două instanțe ale clasei DEC :

- „optimizare” – optimizarea pe baza unui algoritm fuzzy a fluxului de eșantioane în funcție de prioritatea de procesare a eșantionului;
- „reactiv” – selectarea reactivului pe baza proprietăților eșantionului.

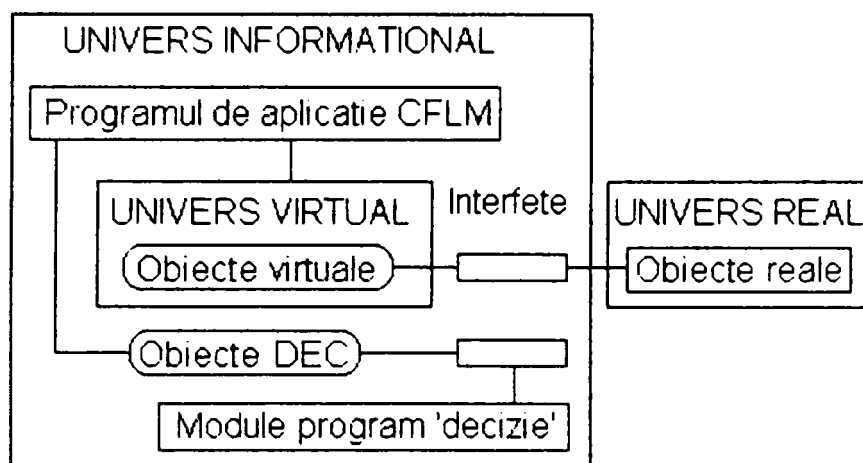


Fig.5.11. Definirea obiectelor din clasa DEC.

Indiferent de clasa din care fac parte, toate obiectele posedă două proprietăți cu aceeași semnificație : nume și identificator. Celelalte proprietăți sunt diferite în funcție de clasă. Numele obiectului este un șir de caractere specific ce este utilizat de către operator iar identificatorul este utilizat intern în timpul preprocesării și rulării programului de aplicație.

Pentru fiecare tip de interacțiune dintre obiecte se definește un „specificator” ce este utilizat în cadrul instrucțiunilor de execuție a unor sarcini în programul de aplicație.

În continuare sunt analizate clasele definite și obiectele ce aparțin acestora.

Clasa Stocare (STO).

Proprietățile obiectelor din clasa „Stocare” sunt :

- 'name' – nume obiect;
- 'id' – identificator obiect;
- 'status' – stare obiect.

Valoarea proprietății „status” a obiectelor din clasa „Stocare” arată dacă stocatorul este liber („empty”) sau ocupat („full”).

Pentru această clasă se definesc interacțiunile :

- deplasare la postul de stocare (figura 5.12.a.), deplasarea eșantionului din punctul curent A, în punctul B, mișcarea (1), specificator de execuție „go”;
- încărcare stocator (figura 5.12.b.), compusă din deplasarea A-(1)-B, deschidere griper (2), deplasarea B-(3)-A, specificator de execuție „load”;
- descărcare stocator (figura 5.12.c.), compusă din deschidere griper (1), deplasarea A-(2)-B, închidere griper (3), deplasare B-(4)-A, specificator de execuție „unload”.

Clasa Procesare Asistată de robot (PAR).

Pentru această clasă s-au definit următoarele proprietăți :

- 'name' – nume obiect;
- 'id' – identificator obiect;
- 'status' – stare obiect
- 'job_result' – rezultatul procesării

Proprietatea „status” poate lua valorile ocupat („busy”) sau liber („ready”) și arată dacă sarcina de procesare este în curs de execuție sau a fost terminată.

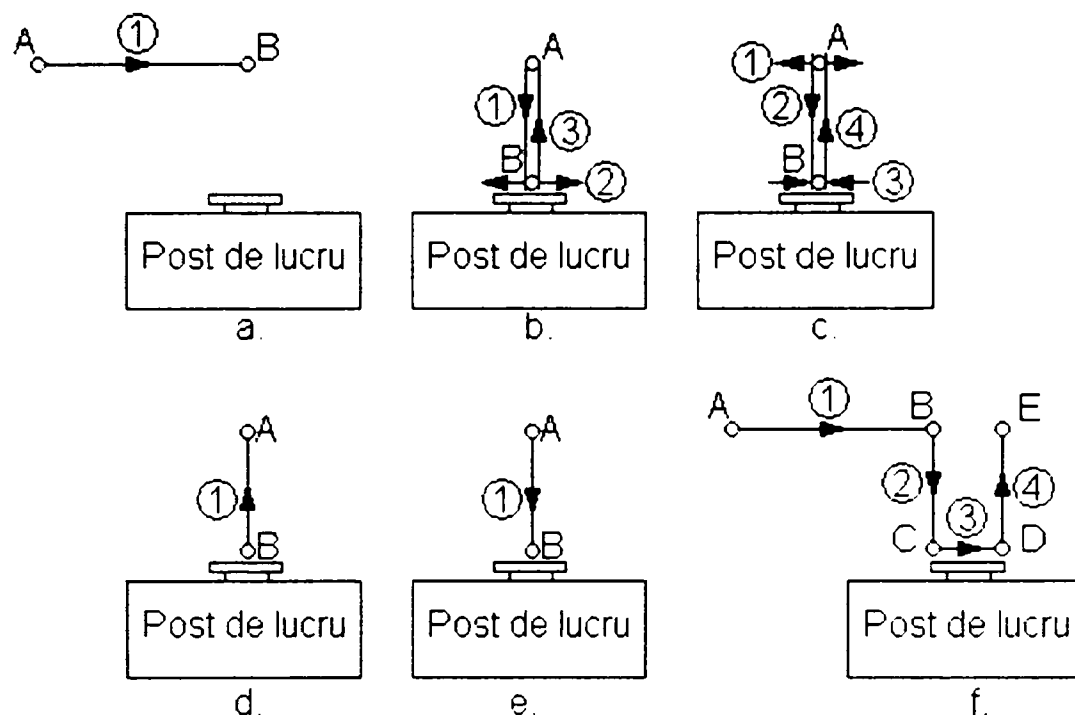


Fig.5.12. Schema explicativă a specificatorilor de execuție.

Proprietatea „job_result” este utilizată numai la posturile de procesare care generează un rezultat al procesării. În cadrul CFLM1 un astfel de post este cel de analiză a stării suprafeței la care rezultatul procesării („job_result”) poate lua una dintre valorile :

- job_result = „ok” - starea suprafeței este corespunzătoare și eșantionul poate trece la postul de atac cu reactiv și mai departe la analiza microstructurii;
- job_result = „șlefuire_eboș” – rugozitatea suprafeței eșantionului este mare și trebuie să se continue procesarea la postul de șlefuire grosieră;
- job_result = „șlefuire_fina” – rugozitatea suprafeței eșantionului este medie și trebuie să se continue procesarea la postul de șlefuire de fină;

În cazul celorlalte posturi PAR, proprietatea „job_result” nu a fost utilizată ea având valoarea „” (empty string - șir ce nu conține nici un caracter).

Interacțiunile pentru clasa PAR sunt :

- deplasare la postul de stocare (figura 5.12.a.), deplasarea eșantionului din punctul curent A, în punctul B, mișcarea (1), specificator de execuție „go”;
- apropiere de punctul de procesare (figura 5.12.e.), deplasarea A-(1)-B, specificator de execuție „down”;
- depărtare de punctul de procesare (figura 5.12.d.), deplasarea B-(1)-A,

specificator de execuție „up”.

- mișcări executate de robot în timpul procesării eșantionului (figura 5.12.f.), simbolizată aici prin deplasarea C-(3)-D, dar care poate să fie un set de mișcări de orice fel, specificator de execuție „assist”.

- procesarea eșantionului realizată de către PAR ce se realizează simultan cu mișcările de tip „assist” ale robotului, specificator de execuție „job”.

Ansamblul mișcărilor necesare unei operații de procesare PAR este alcătuit prin compunerea unui set de mișcări („go”, „down”, „assist”, „job”, „up”) ce pot fi specificate în subrutine. În figura 5.12.f. este prezentat ansamblul mișcărilor corespunzătoare unui proces PAR.

Schema lansării în execuție a instrucțiunilor ce conțin specificatorii „assist” și „job” este redată în figura 5.13.

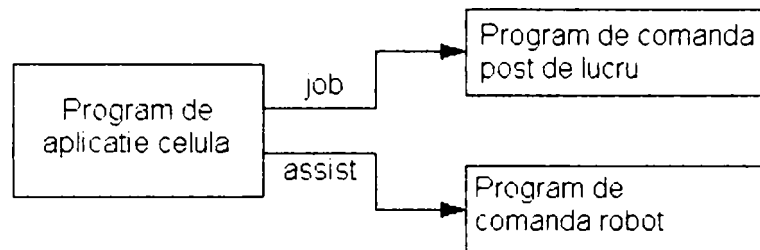


Fig.5.13. Schema de utilizare a specificatorilor de execuție „job” și „assist”.

Clasa Procesare Independentă de Robot (PIR).

Proprietățile clasei PIR sunt următoarele :

- 'name' – nume obiect
- 'id' – identificator obiect
- 'load_status' – stare de încărcare
- 'active_status' – stare de activitate
- 'job_result' – rezultatul procesării.

Proprietatea „load_status” exprimă starea de încărcare sau descărcare a postului putând lua valorile liber („empty”) respectiv ocupat („full”). Proprietatea „active_status” descrie starea în care se află procesul pe postul respectiv. Dacă procesul este în curs de execuție valoarea proprietății este „busy” (ocupat) iar dacă procesarea a fost terminată și eșantionul a fost adus în punctul de schimbare eșantion (PSE) atunci valoarea proprietății este setată „ready” (liber).

Interacțiunile robotului cu postul de lucru sunt inițiate în funcție de combinația valorilor proprietăților „load_status” și „active_status” conform tabelului 5.1.

Tabel 5.1.

Valoarea proprietății obiectului PIR		Situație	Acțiunea întreprinsă de obiectul ROBOT
load_status	active_status		
Empty	Busy	imposibilă	-
Empty	Ready	posibilă	încărcarea postului
Full	Busy	posibilă	-
Full	Ready	posibilă	descărcarea postului

Clasa Transfer (ROBOT).

Proprietățile obiectelor din clasa ROBOT sunt următoarele :

- 'name' – nume obiect
- 'id' – identificator obiect
- 'status' – stare robot
- 'griper' – stare griper

Proprietatea „status” descrie starea de ocupat („busy”) sau liber („ready”) a robotului. Inițierea unei acțiuni a robotului poate fi făcută doar când acesta se găsește în starea „ready”

Proprietatea „griper” poate lua valorile deschis („open”) sau închis („close”) și descrie starea în care se găsește dispozitivul de prehensiune al robotului. Această proprietate se testează pentru a evita eventuale coliziuni.

Clasa Decizie (DEC)

Proprietățile obiectelor din clasa „DEC” sunt :

- 'name' – nume obiect
- 'id' – identificator obiect
- 'status' – stare decizie
- 'result' – rezultat decizie
- 'data' – date

Proprietatea „status” se referă la starea de ocupare a obiectului : dacă modulul program „decizie” (figura 5.11), este în execuție valoarea proprietății este ocupat („busy”) iar dacă s-a terminat rularea este liber („ready”).

Rezultatul rulării programului este memorat în proprietatea „result” a obiectului. Ca și în cazul celorlalte clase și clasa DEC poate avea mai multe instanțe. În cadrul CFLM1 s-au definit, după cum s-a arătat anterior, două instanțe: „optimizare” și „reactiv” ce apelează fiecare câte un modul program decizie diferit.

În cazul obiectului „optimizare” proprietatea „result” poate lua ca valoare numele unui stocator de pe care se va descărca un eșantion („stocator11”, ..., „stocator23”).

În cazul obiectului „reactiv” proprietatea „result” poate lua ca valori numele unuia din posturile de atac cu reactiv, respectiv „nital” sau „persulfat”.

Procesul de decizie se inițializează prin instrucțiunea „execute” urmată de specificatorul de execuție „eval” și doi parametrii : numele obiectului DEC și numele unui fișier ce conține datele inițiale necesare luării deciziei.

Clasa Piesa (ESA)

Obiectele din clasa ESA pot fi definite în două moduri diferite. În primul caz (figura 5.14.a.) ESA sunt tratate ca obiecte virtuale de sine stătătoare. În realitate, în cadrul CFLM1, obiectul ESA este întotdeauna cuplat la un alt obiect virtual. Pentru realizarea urmăririi eșantioanelor va trebui atunci, ca fiecare obiect virtual din altă clasă decât ESA, să posede o proprietate ce să identifice eșantionul care este cuplat lui, la un moment dat. La inițierea unei interacțiuni, ce poate fi dependentă de proprietățile eșantionului sau la terminarea unei interacțiuni ce modifică proprietățile eșantionului, va trebui regăsit obiectul eșantion pe baza

identificatorului acestuia. Această regăsire implică o serie de operații la nivelul programului de aplicație. Din această cauză este util să se atașeze toate proprietățile eșantionului (nu numai identificatorul lui), tuturor celorlalte obiecte virtuale (figura 5.14.b). În acest caz nu va mai exista obiectul virtual ESA, în mod distinct, ci vor exista doar proprietățile acestuia atașate altui obiect virtual.

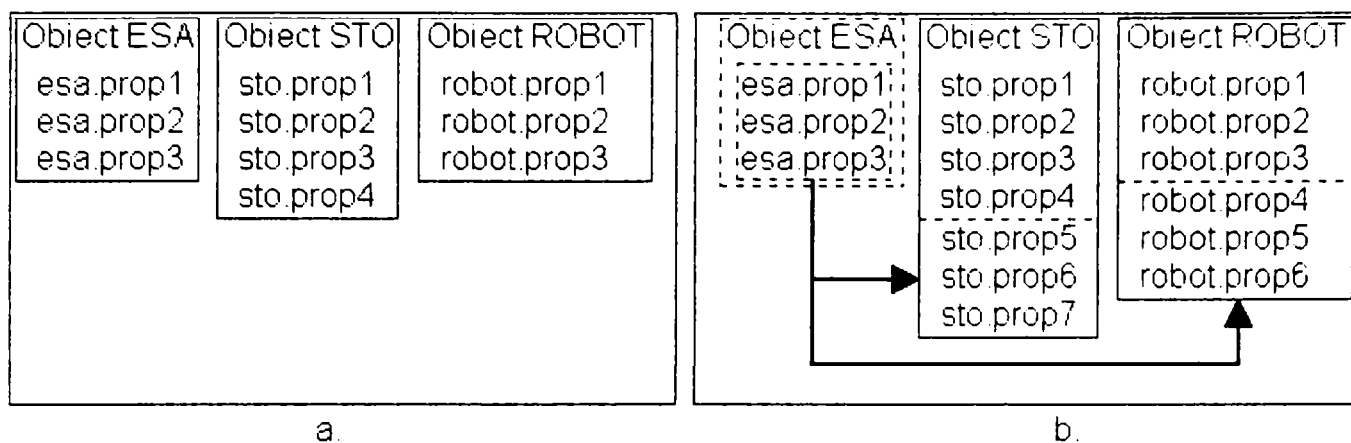


Fig.5.14. Schema tratării obiectului ESA : a.) obiect distinct: b.) obiect înglobat.

În figura 5.14.b proprietățile obiectului ESA sunt transferate obiectelor STO și ROBOT :

```

esa.prop1 => sto.prop5
           => robot.prop4
esa.prop2 => sto.prop6
           => robot.prop5
esa.prop3 => sto.prop7
           => robot.prop6
    
```

Această tratare simplifică programele de aplicație eliminând căutările ce sunt necesare în cazul eșantioanelor abstractizate ca obiecte virtuale distincte.

Proprietățile eșantionului ce vor fi atașate tuturor celorlalte obiecte virtuale, sunt :

- 'probe_id' – identificator eșantion
- 'probe_code' – codul extern al eșantionului
- 'probe_path' – itinerar eșantion
- 'probe_template' – șablon eșantion
- 'probe_priority' – prioritate inițială a eșantionului

Proprietatea „probe_id” conține identificatorul intern care se atribuie eșantionului la intrarea în CFLM1, similar unui marcaj. La ieșirea eșantionului, identificatorul este eliberat putând fi atribuit altui eșantion ce va intra în celulă.

Proprietatea „probe_code” are ca valoare codul extern al eșantionului, ce cuprinde identificatorul clientului și denumirea ce identifică în mod univoc eșantionul în afara CFLM1.

Proprietatea „probe_path” specifică itinerarul eșantionului în cadrul celulei stabilit de către operatorul de celulă. Există o serie de itinerarii predefinite, dar dacă este necesar, se pot defini itinerarii adiționale.

Pentru un anumit itinerar tehnologic pot exista mai multe seturi de parametri prin care se specifică datele concrete ale procesărilor (de exemplu timpul de prelucrare la șlefuire, sau timpul de atac cu reactiv). Aceste seturi de parametri sunt memorati într-o bază de date. Un set de parametri („șablon” sau „template”), din baza de date este atribuit fiecărui eșantion la intrarea acestuia în CFLM1, numărul șablonului fiind memorat în proprietatea „probe_template”.

În general fluxul pieselor într-o celulă flexibilă este stabilit pe baza unor criterii de optimizare. Dintre aceste criterii pot fi amintite : respectarea termenului de livrare către beneficiar, minimizarea duratei de procesare și maximizarea încărcării utilajelor. Primul criteriu a fost utilizat la stabilirea priorităților inițiale ale eșantioanelor. Pe parcursul procesării eșantioanelor, au fost aplicate și celelalte două criterii acestea împreună cu primul criteriu constituindu-se în intrările procedurilor fuzzy de stabilire a priorităților operative. Criteriul de respectare a termenului de livrare este specificat prin proprietatea „probe_priority”.

5.5.2.3. Specificarea sintaxei limbajului programelor de aplicație sursă.

5.5.2.3.1. Aspecte generale.

Limbajul programelor de aplicație nu este un limbaj complet în sensul clasic. Acest limbaj nu posedă elemente de declarație a identificatorilor și nici instrucțiuni de atribuire sau operații matematice. Toți identifiicatorii utilizați sunt predefiniți în module separate (module MATLAB). De remarcat însă că definirea unor noi identifiicatori nu este necesară în mod curent ci numai în cazul în care se modifică fizic structura CFLM1 (se introduc obiecte fizice în plus față de cele existente). Pe de altă parte limbajul posedă toate facilitățile unui limbaj structurat ce asigură realizarea buclelor, deciziilor și a apelului de subrutine.

Se prezintă în continuare elementele limbajului de aplicație.

5.5.2.3.2. Identificatori, simboluri speciale, separatori.

Atomii lexicali (cele mai mici entități) ai limbajului sunt clasificați în câteva grupe în funcție de rolul pe care îl îndeplinesc în cadrul programului.

Identificatori.

Identifiicatorii sunt șiruri de caractere compuse din literele alfabetului englez și cifrele de la 0 la 9. Identifiicatorii utilizați se împart în două categorii: instrucțiuni și constante.

Instrucțiunile sunt specificate prin cuvintele rezervate :

- loop
- break
- endloop
- if
- is
- endif

- execute
- endprogram

Constantele sunt specificate printr-un șir de caractere ce reprezintă :

- numele unui obiect : „stocator01”, „robot_h5”;
- numele unei proprietăți a unui obiect, ce se formează din numele unui obiect, caracterul „.” și numele proprietății, $\langle obiect \rangle . \langle proprietate \rangle$: „stocator01.status”, „robot_h5.griper”;
- valoarea unei proprietăți a unui obiect : „full”, „ready”;
- numele unui specificator de execuție : „go”, „load”;
- un șir de caractere ce definește rezultatul unei procesări : „ok”, „slefuire_ebos”, „stocator01”.

Faptul că se folosește același identificator atât pentru reprezentarea numelui unui obiect cât și pentru reprezentarea rezultatului unei procesări arată că limbajul de aplicație este un limbaj „dependent de context”.

Constantele enumerate nu pot fi modificate de operator decât dacă se modifică și modulele de interpretare aferente (respectiv dacă se modifică structura fizică a CFLM1).

Constantele utilizate la definirea subrutinelor și a parametrilor formali se pot stabili de către operator și nu implică modificări ale modulelor de interpretare :

- numele unei subrutine;
- numele unui parametru formal;
- cuvântul rezervat ‚void’.

Simboluri speciale.

Simbolurile speciale sunt :

- caracterul ‚%’ – utilizat în frazele comentariu, ce nu sunt fraze interpretabile;
- caracterul ‚/’ – utilizat în cadrul instrucțiunilor „execute”;
- caracterul ‚#’ – utilizat în prima frază dintr-o subrutină;
- caracterele ‚(‚, ‚) ’ – utilizate la specificarea parametrilor unei instrucțiuni sau a unei subrutine
- caracterul ‚.’ – utilizat în cadrul specificării proprietății unui obiect.

Separatori

În general atomii lexicali sunt separați prin caracterul ‚ ’ (spațiu), În afară de cazurile :

- specificarea unei liste de parametrii : între parametrii din listă se folosește caracterul ‚, ’;
- utilizarea specificatorului de execuție: între instrucțiunea „execute” și specificatorul de execuție se utilizează caracterul ‚/’;

5.5.2.3.3. Sintaxa și semantica limbajului.

Sintaxa limbajului de aplicație este descrisă în continuare, utilizând notația BNF (Backus-Naur Form).

Instrucțiunile if, is, endif. Se utilizează pentru implementarea deciziilor

```
<instr. if> ::= <clauza if>
                <lista instr.>
                endif
<clauza if> ::= if <obiect.proprietate_obiect>
                is <valoare_proprietate_obiect>
<lista instr.> ::= <lista instr.>
                <instr.>
```

Clauza if compară valoarea memorată în <obiect.proprietate_obiect> (care se actualizează în permanență) cu <valoare_proprietate_obiect> prestabilită la scrierea programului. În cazul returnării valorii logice „adevărat” se execută <lista instr.> , iar în cazul returnării valori logice „fals”, se execută instrucțiunea imediat următoare instrucțiunii *endif*.

Instrucțiunile loop, break, endloop. Se utilizează pentru implementarea buclilor de program.

```
<instr. loop> ::= loop
                <lista instr.>|
                <lista instr.>
                <clauza if>
                <lista instr.>
                break
                endif
                endloop
```

La preprocesare instrucțiunea *loop* se interpretează ca o instrucțiune condițională, ieșirea din buclă realizându-se la cererea operatorului. Din această cauză instrucțiunea *break* de întrerupere a buclei nu este necesară decât în cazul în care bucla trebuie întreruptă pe baza altei condiții decât cererea expresă a operatorului.

Instrucțiunea execute. Se utilizează pentru inițierea unor acțiuni la nivelul obiectelor.

```
<instr. execute> ::=
execute/<specificator_de_execuție>(<lista parametrii>)
<lista parametrii> ::= <nume_obiect_1>,<nume_obiect_2>|
                    void,<nume_obiect>|
                    <nume_obiect>,<nume_fisier>
```

În cazul în care *execute* specifică interacțiunea dintre două obiecte, lista

parametrilor cuprinde *<nume_obiect_1>* și *<nume_obiect_2>*. În cazul în care acțiunea se referă doar la un singur obiect (specificator de execuție „job”), pe locul primului element al listei se specifică cuvântul rezervat *void*. Pentru specificatorul de execuție „eval” parametrii sunt : nume obiect decizie (*<nume_obiect>*) și nume fișier de date (*<nume_fisier>*).

Apelul și specificarea subrutinelor.

Apelul subrutinelor se realizează prin sintaxa :

<nume_subrutina>(<parametru_efectiv_1>,<parametru_efectiv_2>)

Subrutinele sunt salvate în fișiere text separate al căror nume este *<nume_subrutina>*. Prima linie interpretabilă dintr-o subrutină are sintaxa :

#<nume_subrutina>(<parametru_formal_1>,<parametru_formal_2>)

În cadrul preprocesării parametrii formali sunt înlocuiți cu parametrii efectivi.

Instrucțiunea endprogram.

La sfârșitul unui program de aplicație se specifică instrucțiunea *endprogram*.

Astfel orice program de aplicație va avea sintaxa :

*<lista_instr.>
endprogram*

5.5.2.3.4 Exemplu de aplicare a sintaxei.

Se redă pe scurt un exemplu pentru sintaxa prezentată, printr-un program de aplicație simplu. Etichetele utilizate nu sunt parte a programului ci au doar rol explicativ. În text etichetele sunt scrise între paranteze rotunde.

Bucloa *loop* (02–18) este parcursă până când operatorul cere întreruperea programului. Dacă *stocator01* este ocupat și robotul are gripper-ul deschis *robot_h5* descarcă stocatorul (04) și transferă eșantion la postul de analiză a stării suprafeței (06). Dacă rezultatul analizei este „ok” (07) și *stocator11* este liber (09), eșantionul este transferat postului *stocator11* (10) și se realizează condiția de ieșire din bucla *loop* (08-13). Dacă postul *stocator11* este ocupat bucla *loop* (08-13) se execută până când *stocator11* se eliberează. Dacă rezultatul analizei este *slefuire_fina* (15) robotul transferă eșantionul la postul „slefuire_fină” și asistă prelucrarea (16). Ciclurile de analiză și șlefuire continuă până când analiza stării suprafeței generează rezultatul „ok” sau rularea programului este întreruptă de către operator.

```

01 % Program de aplicatie sursa.
02 loop
03     if stocator01.status is full
04         descarcare(robot_h5,stocator01)
05     endif
06     prel_ar(robot_h5,analiza_supr)
07     if analiza_supr.job_result is 'ok'
08         loop
09             if stocator11.status is empty
10                 incarcare(robot_h5,stocator11)
11                 break
12             endif
13         endloop
14     endif
15     if analiza_supr.job_result is 'slefuire_fina'
16         prel_ar(robot_h5,slefuire)
17     endif
18 endloop
19 endprogram

```

În linia (04) este apelată subrutina „descărcare” ce are următoarea sintaxă :

```

01 % Subrutina "descarcare"
02 #descarcare(robot,postdelucru)
03 if robot.status is ready
04     if robot.griper is open
05         execute/go(robot,postdelucru)
06         execute/unload(robot,postdelucru)
07     endif
08 endif

```

Alte tipuri de subrutine sunt date în ANEXA 5.1.

5.5.2.4. Traducerea programelor de aplicație.

Pentru a fi posibilă rularea programului de aplicație a CFLM1, acesta trebuie să fie tradus în format MATLAB. Traducerea se realizează „off-line”, de către un program special numit preprocesor. În cadrul traducerii codul scris de utilizator și salvat într-un fișier text (ASCII), se numește „cod sursă” iar codul MATLAB generat de către preprocesor se numește „cod destinație”

Traducerea programului de aplicație se realizează în două etape, așa cum este arătat în figura 5.15.

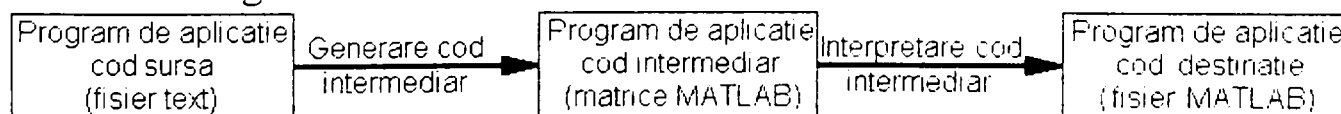


Fig.5.15. Diagrama etapelor preprocesării.

În etapa de generare a codului intermediar se realizează următoarele operații:

- eliminarea liniilor comentariu;

- despărțirea liniilor de cod în identificatori;
- căutarea fișierelor subrutine, înlocuirea parametrilor formali cu parametrii efectivi și inserarea liniilor din subrutine în programul principal.
- regăsirea identificatorului obiectului în funcție de denumirea acestuia (modulul „name2struct”).

Rezultatul generării codului intermediar este o matrice (matricea „main” din figura 5.16), cu patru coloane și un număr de linii egal cu numărul de linii al programului de aplicație plus numărul de linii al subrutinelor utilizate.

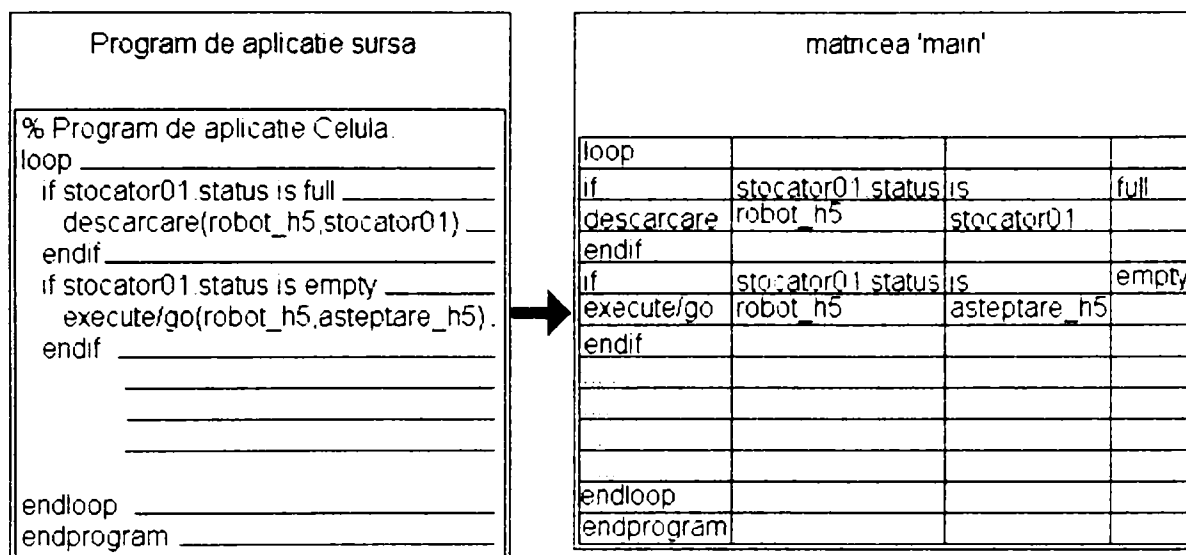


Fig.5.16. Schema generării codului intermediar.

Prima coloană a matricii conține denumirea instrucțiunii, în funcție de care se lansează modulul program de interpretare al instrucțiunii aferente. Fiecare modul de tratare al instrucțiunilor generează una sau mai multe instrucțiuni MATLAB ce în final alcătuiesc codul destinație. Descrierea detaliată a interpretării este redată în paragraful următor.

5.5.2.5. Structura programelor de aplicație MATLAB.

După preprocesare programul de aplicație (în format MATLAB) este gata de rulare. Fiecărei clase de obiecte îi corespunde un modul de tip „interfață_”, care identifică obiectul implicat în interacțiune și în funcție de acesta lansează un modul de tip „com_”.

După fiecare interacțiune se execută un modul de tip „flux_”, selectat de modulul „act_flux” pentru a se realiza reactualizarea diagramei de urmărire a eșantioanelor.

Înainte de fiecare bloc de decizie se execută un modul de tip „conf_” pentru reactualizarea stărilor obiectelor virtuale (figura 5.17) pentru ca decizia să se evalueze pe baza proprietăților reactualizate. Schema explicativă a modului de funcționare a programului de comandă a celulei este arătată în figura 5.18.

Obiectele virtuale sunt declarate într-un fișier separat „obiecte”, împreună cu matricile de definire a interacțiunilor („metode”).

Matricile „metode” pot să conțină :

- numărul unui modul program robot (MPR);
- numărul unui modul program micro-calculator (MPMC);

În cazul specificatorilor de execuție : „go”, „up”, „down”, „load”, „unload” și „assist” matricile "metode" conțin numărul MPR ce se selectează din memoria CNC robot și se execută, specifice fiecărui post de lucru.

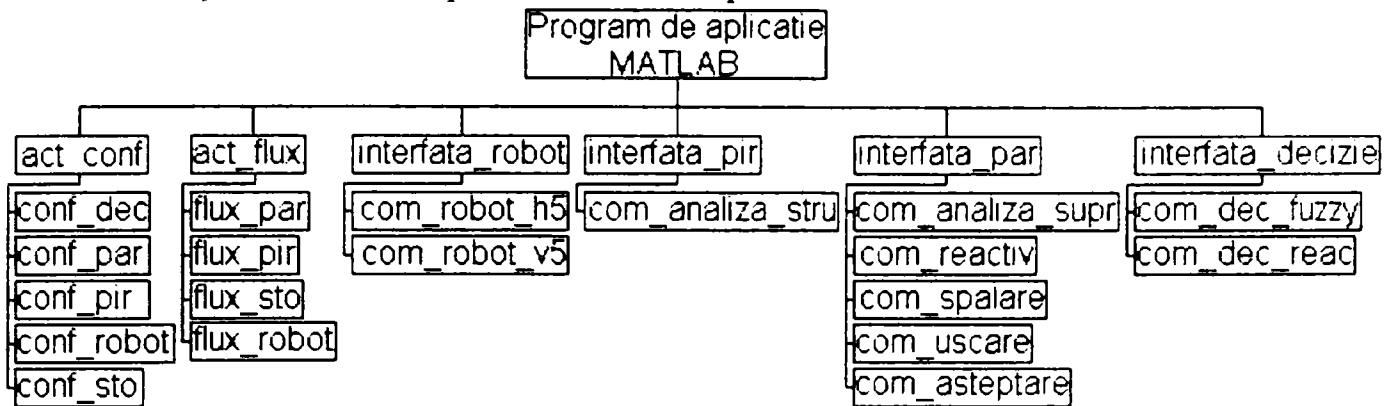


Fig.5.17. Diagrama programului de aplicație în format MATLAB.

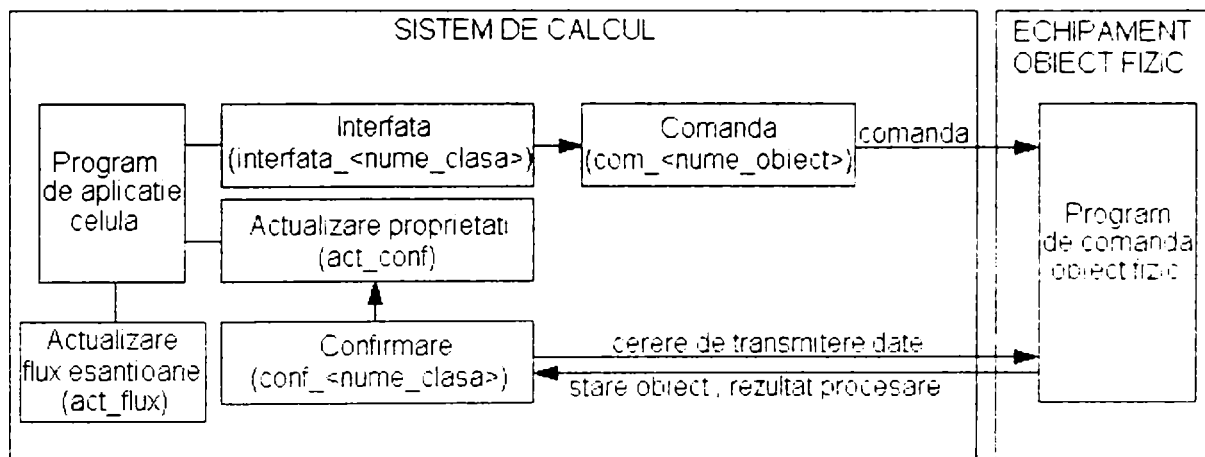


Fig.5.18. Schema funcționării programului de aplicație.

Pentru specificatorii : „go”, „up”, „down”, „load”, „unload” metoda se selectează pe baza identificatorului robot și a identificatorului post de lucru.

Pentru specificatorul „assist” metoda se selectează pe baza identificatorului robot, a identificatorului postului de lucru și a identificatorului șablon („probe_template”) pentru eșantion conectată la postul de lucru respectiv.

În cazul specificatorului „job” matricile "metode" conțin : numărul funcției ce se selectează din memoria micro-calculatorului, specifice fiecărui post de lucru și se selectează pe baza identificatorului postului de lucru și a identificatorului „șablon eșantion” („probe_template”) conectat la postul de lucru respectiv.

În momentul recepției unui eșantion se specifică, prin interfața utilizator :

- itinerarul eșantionului („probe_path”) ce selectează itinerariul pe care îl parcurge eșantion în cadrul sistemului;
- prioritatea inițială („probe_priority”) care este utilizată în cadrul algoritmilor de decizie pentru optimizarea fluxului de eșantioane;
- șablonul („probe_template”) ce specifică un set de parametrii pentru specificatorii de execuție „assist” și „job”;
- codul extern („probe_code”) utilizat pentru identificarea

- eșantionului în afara CFLM1;
- identificatorul intern („probe_id”) utilizat pentru identificarea eșantionului în cadrul CFLM1.

În momentul intrării unui eșantion în sistem (*stocator01.status is full*), se încarcă proprietățile obiectului stocator01 (*sto(1)*):

sto(1).probe_path = <număr itinerar>
sto(1).probe_template = <număr șablon>
sto(1).probe_priority = <număr prioritate>

Când eșantionul este transferat fizic de la un obiect la altul, sunt transferate și valorile proprietății eșantionului în cadrul obiectelor virtuale.

De exemplu :

robot(i).probe_id = *sto(j).probe_id*
robot(i).probe_code = *sto(j).probe_code*
robot(i).probe_path = *sto(j).probe_path*
robot(i).probe_template = *sto(j).probe_template*
robot(i).probe_priority = *sto(j).probe_priority*

unde *i* reprezintă identificatorul robotului iar *j* reprezintă identificatorul stocatorului.

Definirea obiectelor virtuale și a matricilor „metode” sunt date în tabelele din ANEXA 5.2.1.

5.5.3. Prezentarea modulelor program de comandă a celulei flexibile.

5.5.3.1. Descrierea programului de celulă pentru sistemul Sc1

Programul de celulă realizează comanda celulei flexibile și activitățile auxiliare ce asigură buna funcționare a acesteia precum și interfața cu operatorul de celulă.

Programul de celulă este distribuit pe sistemele de calcul Sc1 și Sc2 după cum se arată în figura 5.8.

În figura 5.19. este prezentată interfața programului ce rulează pe sistemul Sc1.

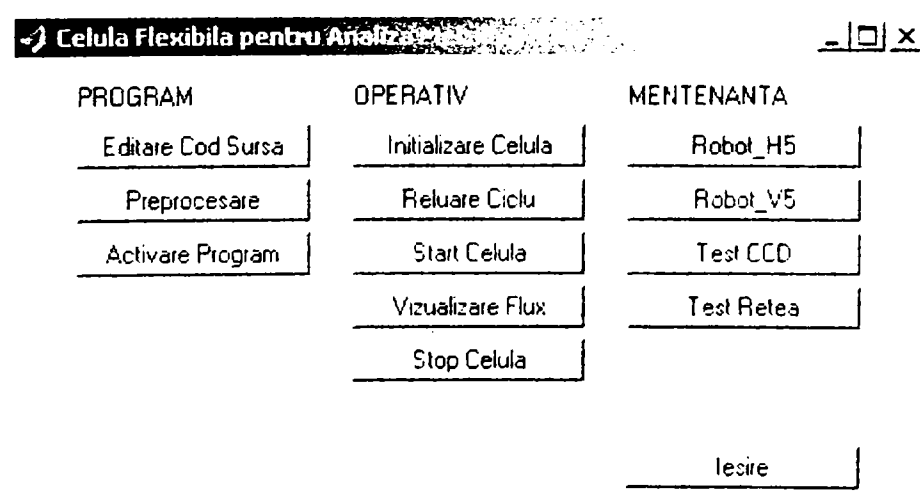


Fig.5.19. Interfața grafică a programului „iosc1”.

Această interfață permite operatorului celulei lansarea în execuție a diverselor module ale programului. Corespondența etichetelor butoanelor interfeței cu modulele ce sunt lansate este prezentată în schema din figura 5.20.

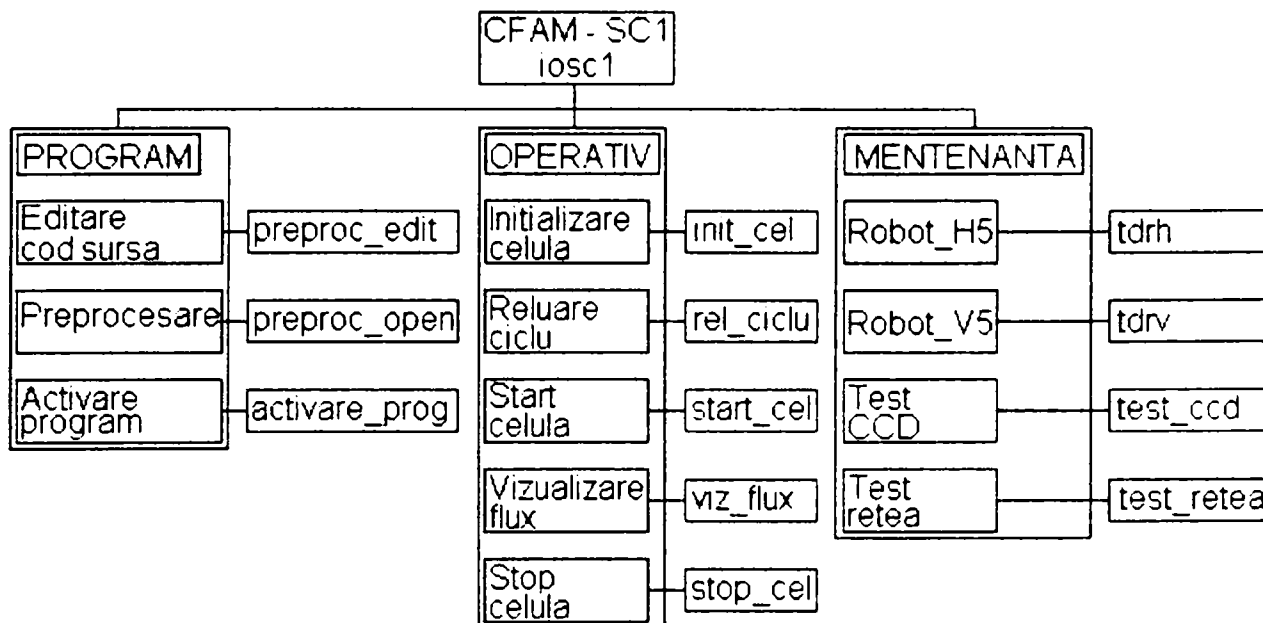


Fig.5.20. Diagrama programului „iosc1”.

Programul principal ‘iosc1’

Programul principal realizează inițializarea variabilelor și obiectelor, setarea acestora la valorile inițiale, și afișarea interfeței grafice arătate în figura 5.19.

Modulele ce pot fi lansate din programul principal se împart în 3 categorii :

- preprocesare - operațiile legate de scrierea, modificarea, și preprocesarea programelor de aplicații;
- operativ - operațiile legate de executarea programului de aplicație celulă și urmărirea fluxului de eșantioane;
- mentenanță – operațiile legate de transmiterea programelor robot și testarea diverselor posturi ale celulei.

Modulul ‘preproc_edit’

Acest modul lansează programul Windows Notepad ce permite generarea și editarea fișierelor text. Cum programele de aplicație ale celulei sunt salvate ca fișiere text, acestea se pot genera și modifica în consecință.

Modulul ‘preproc_open’

Modulul ‘preproc_open’ și modulele subsidiare acestuia realizează generarea codului MATLAB al programului de aplicație celulă. Schema apelării diferitelor module subsidiare este arătată în figura 5.21.

Din programele de aplicație scrise de către operatorul celulei se generează cu ajutorul acestui modul, programe de tip ‘.m’ ce pot fi rulate sub mediul MATLAB. Această operație este realizată de către modulul ‘preproc_open’ în mai multe etape. Se citește fișierul sursă de tip text, linie cu linie, și se analizează fiecare linie de program în parte. Dacă linia citită este un comentariu aceasta se va elimina. Se detectează delimitatorii de cuvinte și se stabilesc cuvintele delimitate.

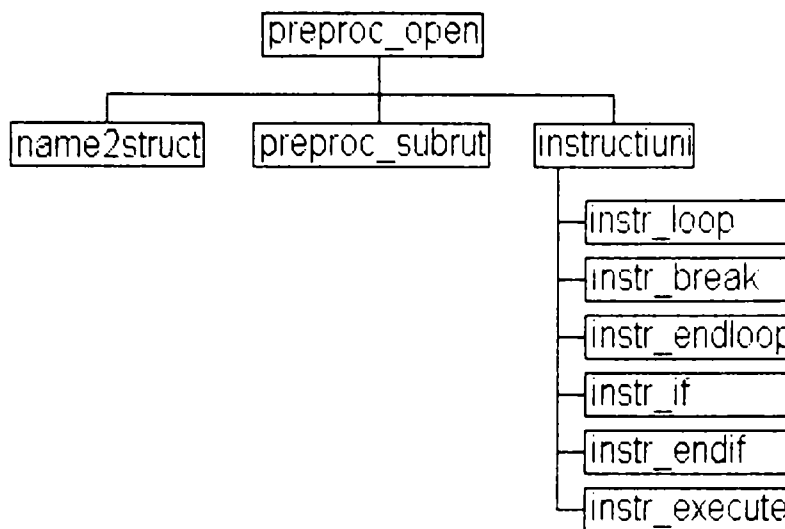


Fig.5.21. Diagrama structurii modulului de preprocesare.

Se testează dacă primul cuvânt delimitat al liniei curente figurează în lista de instrucțiuni. Dacă se regăsește în listă, cuvintele delimitate se înscriu într-o matrice (matricea ,main'). Dacă nu figurează în listă, atunci acesta nu poate fi decât ori numele unei subrutine ori un cuvânt eronat. Se caută deci un nume de subrutină corespunzător și dacă acesta este găsit, atunci se trece la tratarea subrutinei respective. Subrutinele sunt procesate de către un modul separat ,preproc_subrut', și se citesc din fișiere text separate de programul de aplicație principal. Liniile de cod citite din subrutine sunt tratate în mod similar cu cele ale programului de aplicație principal, doar că în acest caz se realizează identificarea parametrilor formali utilizați în subrutine cu parametrii de apel (efectivi) utilizați în cadrul programului de aplicație principal. După realizarea acestor operații se inserează liniile de cod din subrutine în matricea ,main'. Odată ce s-a completat matricea ,main' se trece la analiza fiecărei linii a acesteia, apelându-se modulul ,instrucțiuni', ce lansează în execuție, în funcție de numele instrucțiunii, modulul aferent fiecăreia.

Modulele ,instr_loop', ,instr_break', ,instr_endloop'

Aceste module tratează instrucțiunile ,loop', ,break', ,endllop' din programul de aplicație celulă generând o structură de tip ,while-break-end' compatibilă cu limbajul MATLAB.

Modulele ,instr_if', ,instr_endif'

Tratează structura decizională din programul de aplicație generând structura MATLAB, ,if-end'. Deasemenea înainte de fiecare instrucțiune ,if', generată se introduce o linie de cod ce va apela subrutina ,act_conf'. Această subrutină citește stările obiectelor ce se testează în instrucțiunea „if” pentru ca atunci când se execută blocul decizional, valorile parametrilor pe baza cărora se realizează decizia să fie reactualizate.

Modulul ,instr_execute'

Generează, pe baza parametrilor instrucțiunii „execute” din programul de aplicație text, o linie de cod MATLAB ce conține apelul unei subrutine de tip interfață. Pentru fiecare obiect este definită o subrutină de interfață separată. Pentru obiectele de tip ,transfer' se generează deasemenea și o instrucțiune de apel a unei

subrutine ,act_flux' necesară pentru reactualizarea localizării eșantionului în cadrul celulei flexibile.

Modulul 'activare_prog'

Programul de aplicație în cod MATLAB se salvează după generare într-un fișier cu un nume specificat de către programatorul celulei. Astfel la un moment dat pot exista mai multe variante de programe de aplicație în memoria calculatorului. Pentru eliminarea confuziilor asupra variantei de program ce urmează să ruleze pentru comanda celulei, se lansează modulul ,activare_prog' și se specifică varianta de program de aplicație dorită ce se va copia în fișierul MATLAB cu denumirea ,prog_cel_main' care va fi lansat în execuție la rularea modulului ,start_cel'.

Modulul 'init_cel'

Modulul realizează deschiderea legăturilor de comunicare serială RS232 cu echipamentele de comandă ale roboților H5 și V6 și legătura de comunicare în rețea LAN Win2000, cu sistemul Sc2. În cazul în care stabilirea legăturilor s-a realizat cu succes vor fi afișate mesaje de tip ,OK'. Dacă una sau mai multe legături nu au fost realizate se afișează mesaje de eroare corespunzătoare, de tipul celor arătate în figura 5.22.

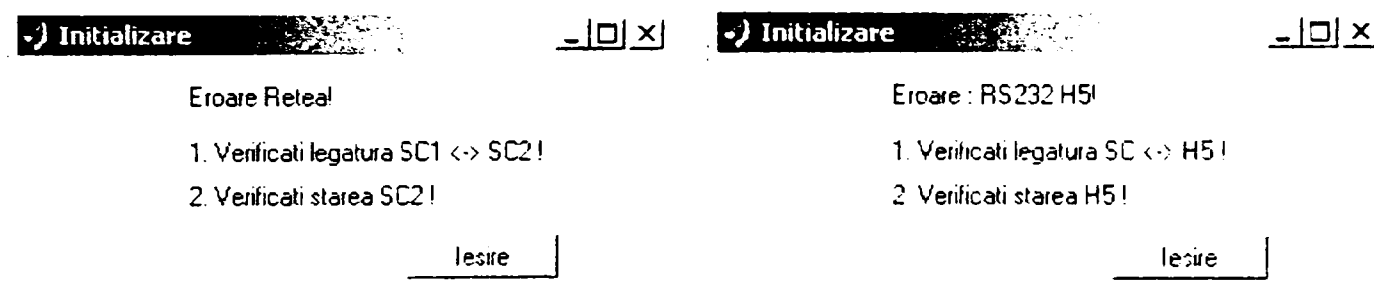


Fig.5.22. Ferestre de mesaje la inițializarea programului.

Modulul 'rel_ciclu'

Permite reluarea ciclului de lucru în cazul în care acesta a fost întrerupt, prin încărcarea stării obiectelor dinaintea întreruperii. În cazul reluării ciclului trebuie eliminate toate neconcordanțele dintre starea obiectelor fizice și cea a obiectelor virtuale.

Modulul 'start_cel'

Modulul dezactivează toate butoanele interfeței grafice ,iosc1' în afară de butonul ,viz_flux' și ,stop_cel' și lansează în execuție forma MATLAB a programului de aplicație celulă. Astfel, în timpul funcționării celulei, utilizatorul poate vizualiza fluxul eșantioanelor și poate opri funcționarea celulei, dar nu îi sunt permise celelalte operații.

Modulul 'viz_flux'

Cu ajutorul acestui modul se realizează vizualizarea pe ecranul sistemului de calcul Sc1 a poziției eșantionului în cadrul celulei flexibile. Fiecărui eșantion, la

intrarea în celula flexibilă, i se atașează un cod de culoare. Numărul codurilor de culoare este egal cu numărul maxim de eșantioane ce pot exista simultan în cadrul celulei. La ieșirea eșantionului din celula flexibilă, codul de culoare este eliberat rămânând disponibil pentru eșantioanele ce urmează să intre în celulă. Schema fluxului de eșantioane afișată este prezentată în figura 5.23.

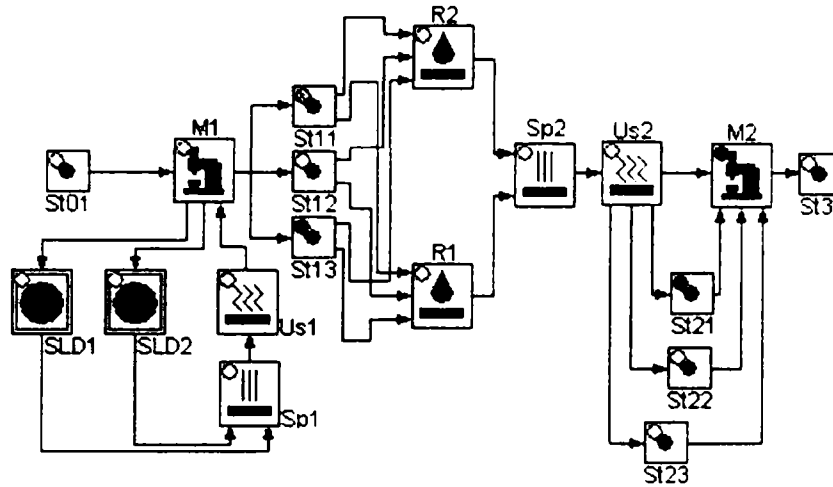


Fig.5.23. Diagrama de vizualizare a fluxului de eșantioane.

Pe lângă această schemă pe ecran este afișată simultan și lista de corespondență a codului numeric al eșantionului cu codul de culoare aferent.

Schema fluxului de eșantioane este actualizată de către modulul ,act_flux' în maniera descrisă mai sus.

Modulul 'stop_cel'

Salvează starea obiectelor virtuale și oprește rularea programului de aplicație celulă.

Modulele 'tdrh'și 'tdrv'

Cu ajutorul acestor module se realizează funcțiile de editare și testare a modulelor program robot (MPR, ANEXA.5.2.2.) cât și a parametrilor mașină ai roboților H5 și V6.

Pentru accesul operatorului la aceste funcții s-a realizat interfața grafică prezentată în figura 5.24.

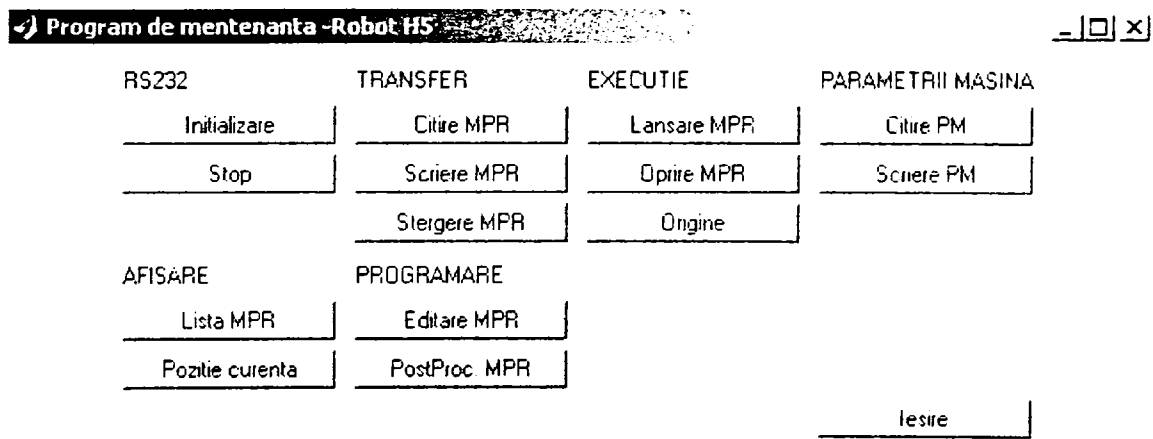


Fig.5.24. Interfața grafică a programului de mentenanță a robotului.

Modulele subsidiare corespunzătoare etichetelor interfeței sunt arătate în figura 5.25.

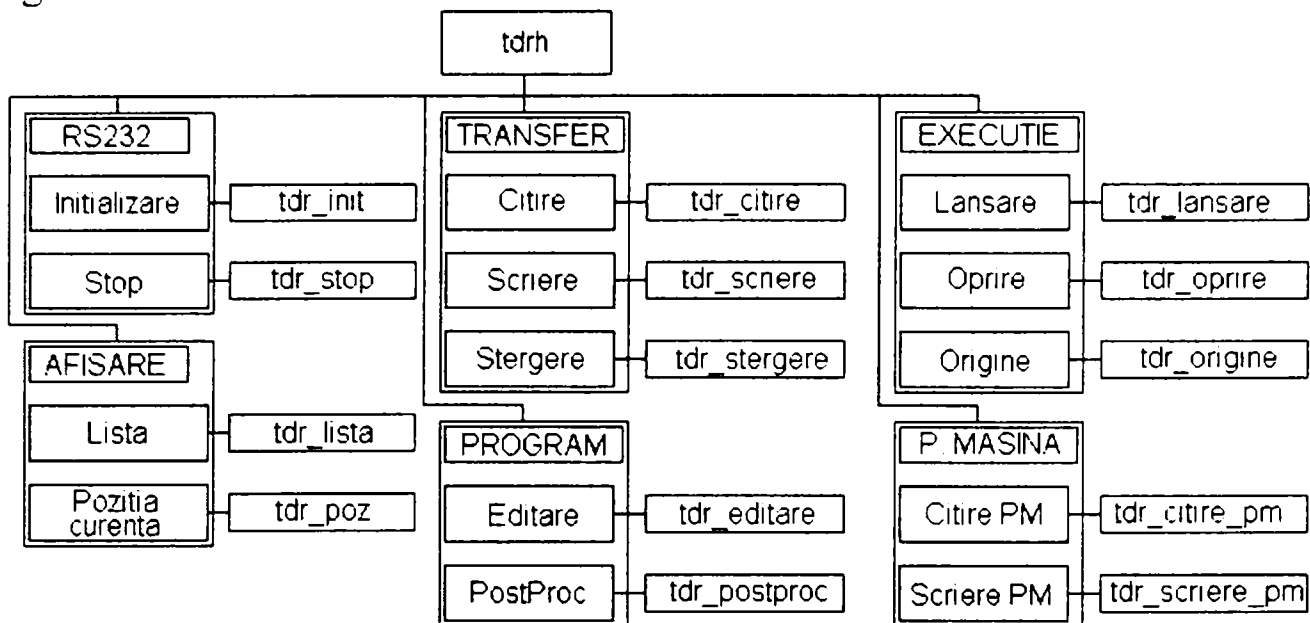


Fig.5.25. Diagrama programului „tdrh”.

Deoarece echipamentele de comandă numerică (CNC-EN5) sunt identice pentru cei doi roboți utilizați modulele ,tdrh’ și ,tdrv’ sunt similare. În continuare sunt descrise modulele corespunzătoare robotului H5 ce sunt similare celor utilizate pentru robotul V6.

Modulul tdr_init

Realizează inițializarea comunicării seriale RS232 (portul COM1) cu echipamentul CNC-EN5. În funcție de succesul sau insuccesul realizării comunicării se afișează un mesaj către operatorul de celulă.

Dacă s-a stabilit linia de comunicare, sunt activate celelalte butoane ale interfeței permițând accesul operatorului la funcțiile de mentenanță.

Modulul tdr_stop

Modulul întrerupe legătura cu echipamentul de comandă al robotului. După întreruperea legăturii, echipamentul trece în modul de comandă manuală.

Modulul tdr_lista

Afișează lista modulelor program robot (MPR) existente în memoria echipamentului CNC-EN5.

Modulul tdr_poz

Citește poziția curentă a axelor robotului și o afișează pe monitorul sistemului de calcul Sc1.

Modulul tdr_citire

Se afișează o fereastră de editare în care se poate specifica numărul MPR, după care se citește MPR din memoria CNC într-un fișier pe sistemul de calcul.

Modulul tdr_scriere

Se afișează o fereastră de editare în care se specifică fișierul tip text ce conține MPR, după care se realizează scrierea programului din fișier în memoria CNC.

Modulul tdr_stergere

Se afișează o fereastră în care se specifică numărul MPR ce urmează să fie șters și se comandă ștergerea acestuia din memoria CNC.

Modulul tdr_editare

Cu ajutorul acestui modul se pot crea și modifica MPR. Acestea se salvează în fișiere tip text în memoria sistemului de calcul.

Modulul tdr_postproc

Fișierele MPR create trebuie completate cu sumele de control cerute de protocolul de comunicație cu CNC-EN5. Acest lucru se realizează cu ajutorul modulului „tdr_postproc”.

Modulul tdr_lansare

Lansează în execuție un program MPR solicitat de către operator.

Modulul tdr_oprire

În cazul în care se constată apariția unor erori în execuția MPR, acesta poate fi oprit prin opțiunea ‚tdr_oprire’.

Modulul tdr_origine

Lansează un MPR dedicat ce comandă robotul într-o situație prestabilită definită ca punct de origine.

Modulele tdr_citire_pm și tdr_scriere_pm

În cazul unor defecțiuni a CNC, după remedierea acestora, este nevoie de reîncărcarea parametrilor mașină. Modulele ‚tdr_citire_pm’ și ‚tdr_scriere_pm’ realizează citirea și scrierea parametrilor mașină.

Modulul 'test_ccd'

Modulul realizează citirea imaginilor captate de camera CCD cuplată la portul USB al sistemului de calcul, și afișarea acestor imagini pe monitor. În cazul în care apar erori la citire se afișează un mesaj prin care se cere ca operatorul să verifice starea camerei și legătura USB aferentă.

Imaginea test a suprafeței unui eșantioane captată cu camera CCD este arătată în figura 5.26.

Modulul 'test_retea'

Modulul testează atât existența legăturii cu Sc2 prin LAN Win2000, cât și starea programului de comandă al celulei ce rulează pe Sc2 (‚iosc2’), dacă acest

program este în derulare sau nu, prin scrierea unor date predefinite într-un fișier pe Sc2 și citirea unui alt set de date modificate de către .iosc2'. În cazul în care datele citite nu corespund se afișează un mesaj de eroare.

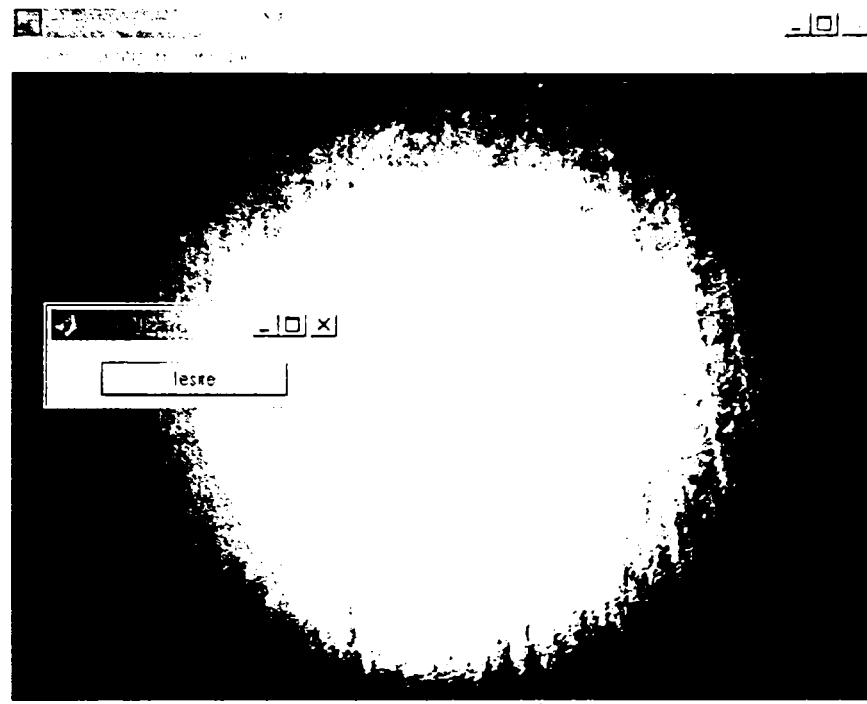


Fig.5.26. Imaginea test pentru camera CCD a postului de analiză a stării suprafeței.

5.5.3.2. Descrierea programului de celulă pentru sistemul SC2

Distribuirea sarcinilor de comandă a celulei flexibile se realizează pe sistemele de calcul SC1 și SC2. Pe sistemul SC2 rulează programul 'iosc2' și modulele subsidiare ce execută sarcinile cerute de către programul de comandă al celulei 'iosc1' ce rulează pe sistemul SC1 și comunică acestuia rezultatele obținute. Interfața grafică a programului 'iosc2' este arătată în figura 5.27, iar modulele lansate în urma activării butoanelor sunt arătate în figura 5.28.

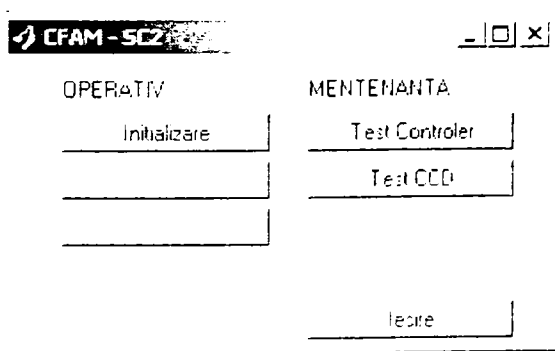


Fig.5.27. Interfața program „iosc2”

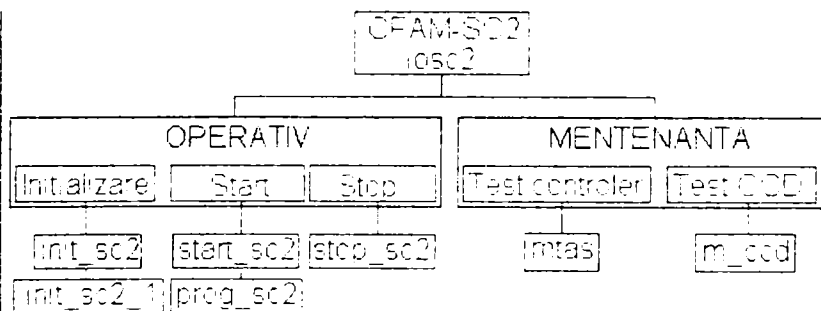


Fig.5.28. Diagrama programului „iosc2”.

Program principal 'iosc2'

Programul principal realizează următoarele sarcini: eliberarea spațiului de memorie utilizat de către variabilele MATLAB; specificarea directoarelor utilizate; inițializarea interfeței grafice utilizator pentru Sc2; definirea obiectelor de tip buton

și text și lansarea la comanda operatorului de celulă a modulelor aferente:

- Grupa 'OPERATIV' :
 - Inițializare - lansează modulul 'init_sc2';
 - Start - lansează modulul 'start_sc2';
 - Stop - lansează modulul 'stop_sc2';
- Grupa 'MENTENANȚĂ' :
 - Test controler - lansează modulul 'mtas';
 - Test CCD - lansează modulul 'm_ccd'.

Modulul 'init_sc2'

Modulul realizează inițializarea ferestrei mesaj arătată în figura 5.29, afișează mesajul 'Cuplati Controler la Sursa!' și lansează modulul 'init_sc2_1'.

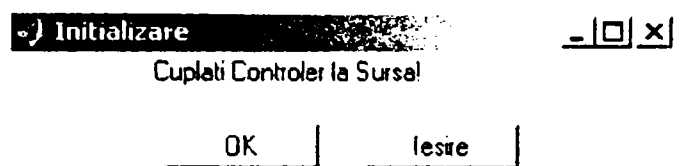


Fig.5.29. Interfața de mesaj către operator.

Modulul 'init_sc2_1'

Modulul inițializează comunicația serială între SC2 și micro-calculatorul BlueEarth 485, prin setarea parametrilor de comunicație și deschiderea canalului serial COM1 (RS232). După deschiderea comunicației seriale se transmite un spațiu (caracterul ASCII 32) și se citește răspunsul de la controler.

În cazul în care controlerul răspunde cu șirul ASCII 'READY', se afișează fereastra de confirmare a validității comunicației seriale COM1 (figura 5.30) și se dezactivează butoanele interfeței 'iosc2', activându-se doar butonul 'Start' ce permite lansarea programului de celulă 'prog_sc2'.

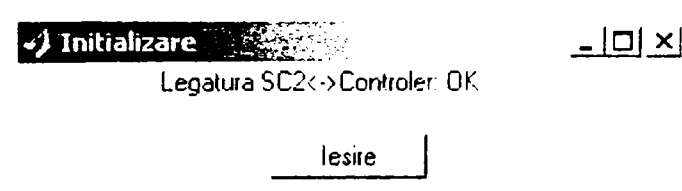


Fig.5.30. Interfața de mesaj a stabilirii legăturii seriale.

În cazul în care controlerul nu răspunde se afișează fereastra mesaj de eroare arătată în figura 5.31, după care se revine în modulul 'iosc2', butonul 'Start' rămânând dezactivat. În acest caz se închide canalul de comunicație serială COM1 (RS232).

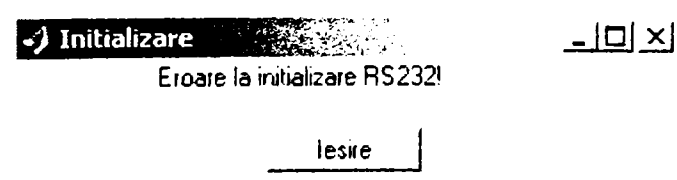


Fig.5.31. Interfața de afișare a erorii.

Modulul 'start_sc2'

La pornirea modului 'start_sc2' se activează butonul 'Stop' pentru a permite operatorului terminarea programului de celulă SC2, când este necesar. Se dezactivează butonul 'Start' pentru evitarea lansării repetate a modului.

Se realizează confirmarea stării 'gata de lucru' prin setarea variabilei 'init_sc2_flag' la valoarea '1' și salvarea acesteia în fișierul de comunicare cu SC1, 'result.mat'.

Se lansează modulul 'prog_sc2'.

Modulul 'prog_sc2'

Programul execută citirea cererilor de sarcini specificate de către SC1 și stocate în vectorul 'request_flag' din fișierul 'request.mat'. Pe baza valorilor elementelor vectorului de cereri, se lansează modulele aferente, așa cum se arată în diagrama din figura 5.32. Rezultatele returnate de către modulele subsidiare se salvează în vectorii ':conf_stocatoare', 'conf_stru', 'job_res_stru', din fișierul de comunicare 'result.mat', pentru a putea fi citite de către SC1. Concomitent, se re setează la '0' valorile vectorului 'request_flag' corespunzătoare modulelor efectuate, pentru a confirma către SC1, executarea cererilor.

Aceste operații sunt executate în buclă atâ timp cât 'init_sc2_flag' are valoarea '1'. 'init_sc2_flag' primește valoarea '0' în momentul în care operatorul apasă butonul 'Stop' din interfața grafică 'iosc2'. Prin această comandă se termină execuția modului 'prog_sc2' și implicit se realizează întreruperea funcționării celulei.

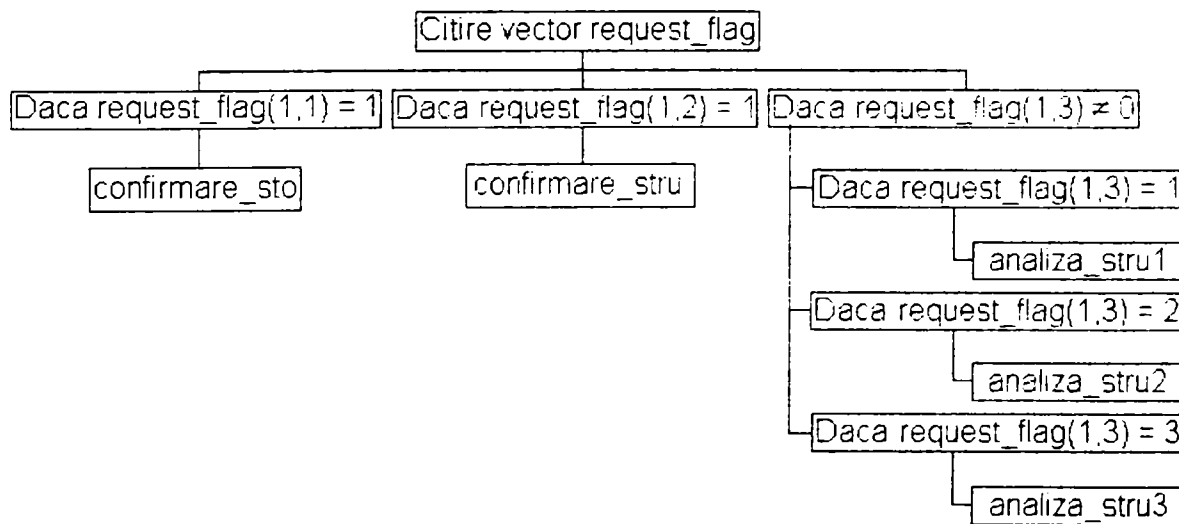


Fig.5.32. Diagrama selectării algoritmului de analiză microstructură.

Modulul 'confirmare_sto'

Modulul 'confirmare_sto' comunică controlerului cererea de citire a valorilor intrărilor digitale ale portului digital pe 8 biți, port A, care este utilizat pentru testarea prezenței eșantionului pe stocatoare, citește valoarea returnată de către controler și o salvează în variabila 'conf_stocatoare'.

Modulul 'confirmare_stru'

Modulul realizează testarea stării limitatoarelor de cursă pe axele mesei microscopului de analiză a microstructurii și returnează confirmarea atingerii punctului de schimbare a eșantionului în variabila 'con_stru'. Aceasta este necesară pentru validarea accesului robotului V5 la eșantion în vederea depunerii sau preluării eșantionului pe/de pe masa microscopului.

Modulele 'analiza_stru1', 'analiza_stru2' și 'analiza_stru3'

Modulele se lansează în funcție de cererea stipulată de programul de celulă a sistemului SC1. În principiu fiecare din module realizează lansarea segmentului de program ce comandă mișcarea mesei microscopului, achiziționează imaginea microstructurii, analizează imaginea și comandă motoarele pas cu pas în consecință. În cadrul algoritmilor de deplasare a mesei microscopului se definesc două puncte semnificative notate PSE (Punct de Schimbare Eșantion) și PL (Punct de Lucru) (figura 5.33).

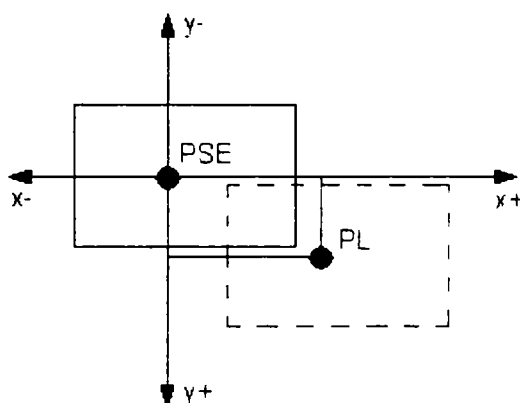


Fig.5.33. Puncte semnificative de poziționare a mesei microscopului

Poziționarea mesei în PSE permite accesul robotului V5 la eșantion în vederea efectuării operațiilor de încărcare/descărcare. Prin poziționarea mesei în punctul PL se aduce eșantion sub obiectivul microscopului. Pornind din acest punct se lansează în execuție unul din algoritmi 'analiza_stru1', 'analiza_stru2' sau 'analiza_stru3' în funcție de sarcina de lucru specifică eșantionului.

Algoritmul 'analiza_stru1', constă în deplasarea mesei microscopului în punctul PL, după care se execută succesiv deplasări (d_2, d_3, d_4, d_5) iar între două deplasări, achiziția imaginilor (A,B,C,D) (figura 5.34).

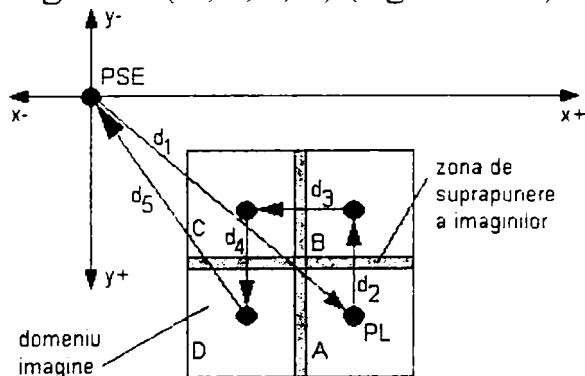


Fig.5.34. Schema de lucru pentru algoritmul „analiza_stru1”.

Deplasările sunt astfel comandate încât imaginile achiziționate să aibă o zonă de suprapunere utilizată la asamblarea acestora într-o singură imagine de dimensiuni mai mari. După achiziția celor 4 imagini se realizează deplasarea în punctul PSE.

Algoritmul 'analiza_stru2', constă din deplasarea în punctul PL, după care se realizează succesiunea de deplasări și achiziții de imagini arătată în figura 5.35. Deoarece, în acest caz, nu este necesară asamblarea imaginilor achiziționate, valoarea deplasărilor este mai mare decât în cazul algoritmului precedent.

După achiziția celor 3 imagini se realizează deplasarea mesei în punctul PSE în vederea descărcării eșantionului de pe microscop (figura 5.35).

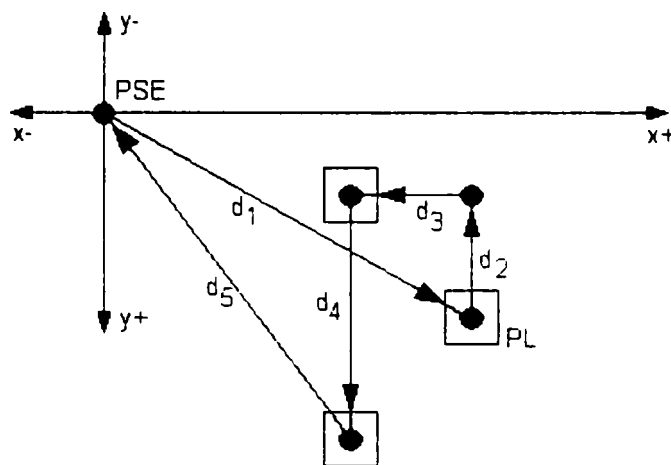


Fig.5.35. Schema de lucru pentru algoritmul „analiza_stru2”.

Algoritmul 'analiza_stru3', constă din deplasarea în punctul PL, achiziția unei singure imagini și revenirea în punctul PSE (figura 5.36). Acest algoritm se poate apela în mod repetat, de câte ori este necesar, schimbând poziția punctului PL.

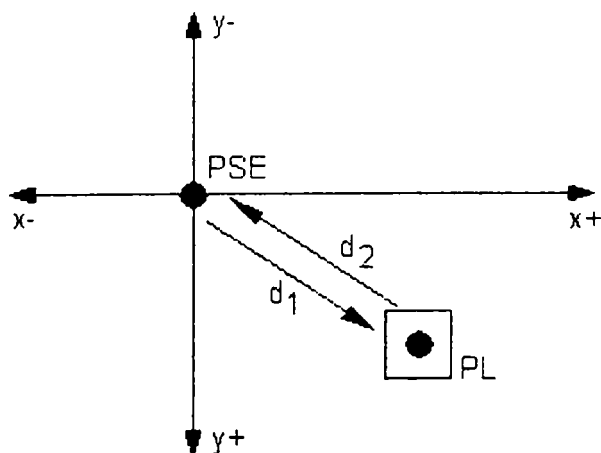


Fig.5.36. Schema de lucru pentru algoritmul „analiza_stru3”.

Modulul 'stop_sc2'

Modulul 'stop_sc2' se execută la comanda operatorului când acesta dorește să întrerupă funcționarea celulei. Se activează butoanele ce au fost active la pornirea programului principal 'iosc2' și se dezactivează butoanele 'Start' și 'Stop' pentru a nu permite pornirea programului 'prog_sc2' înainte de inițializarea

celulei.

Se setează 'init_sc2_flag' la '0' ceea ce întrerupe execuția modului 'prog_sc2' nu înainte însă de a termina orice modul subsidiar în curs de execuție. Se salvează valoarea 'init_sc2_flag' în matricea 'result.mat'. În momentul citirii acestei valori de către SC1 acesta va întrerupe funcționarea programului de celulă ce rulează pe SC1. Se afișează fereastra de mesaj ce solicită operatorului decuplarea controlerului de la sursă (figura 5.37) și în final se întrerupe comunicația serială pe portul COM1.

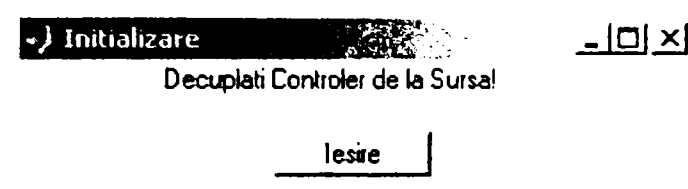


Fig.5.37. Interfață mesaj către operatorul de celulă

Modulul 'mtas'

Acest modul a fost introdus în scopul ajutării operatorului de celulă în cazul în care se dorește crearea de programe controler noi, modificarea unora existente, sau testarea unor funcții realizate de controler.

Modulul 'mtas' cuprinde definirea interfeței grafice și apelarea modulelor subsidiare aferente funcțiilor de mentenanță (figura 5.38).

Acest modul permite :

- editarea unor programe micro-calculator;
- scrierea, ștergerea, citirea și lansarea programelor editate.
- testarea funcțiilor de deplasare a mesei microscopului de analiză structură, a aducerii mesei în poziție de schimbare eșantion și a funcționării senzorilor de prezență eșantion din cadrul stocatoarelor.

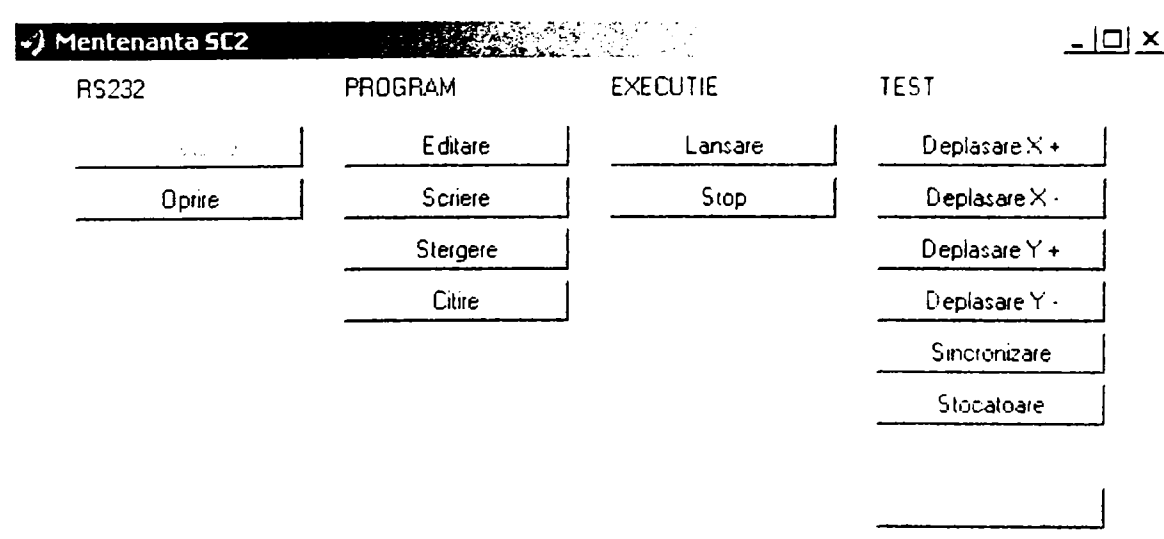


Fig.5.38. Intefafa grafică a programului de mentenanță pe sistemul Sc2.

La pornirea acestui modul, se dezactivează toate butoanele în afară de 'Inițializare' și 'Editare', pentru a permite stabilirea legăturii seriale cu micro-

calculatorul sau după caz editarea programelor.

Corespondența etichetelor butoanelor cu modulele ce sunt lansate la activarea acestora este arătată în figura 5.39.

Modulele 'mtas_init' și 'mtas_init1'

Se realizează inițializarea legăturii seriale cu micro-calculatorul în mod similar cu cele descrise pentru modulele 'init_sc2' și 'init_sc2_1' în scopul funcționării în afara programului de celulă.

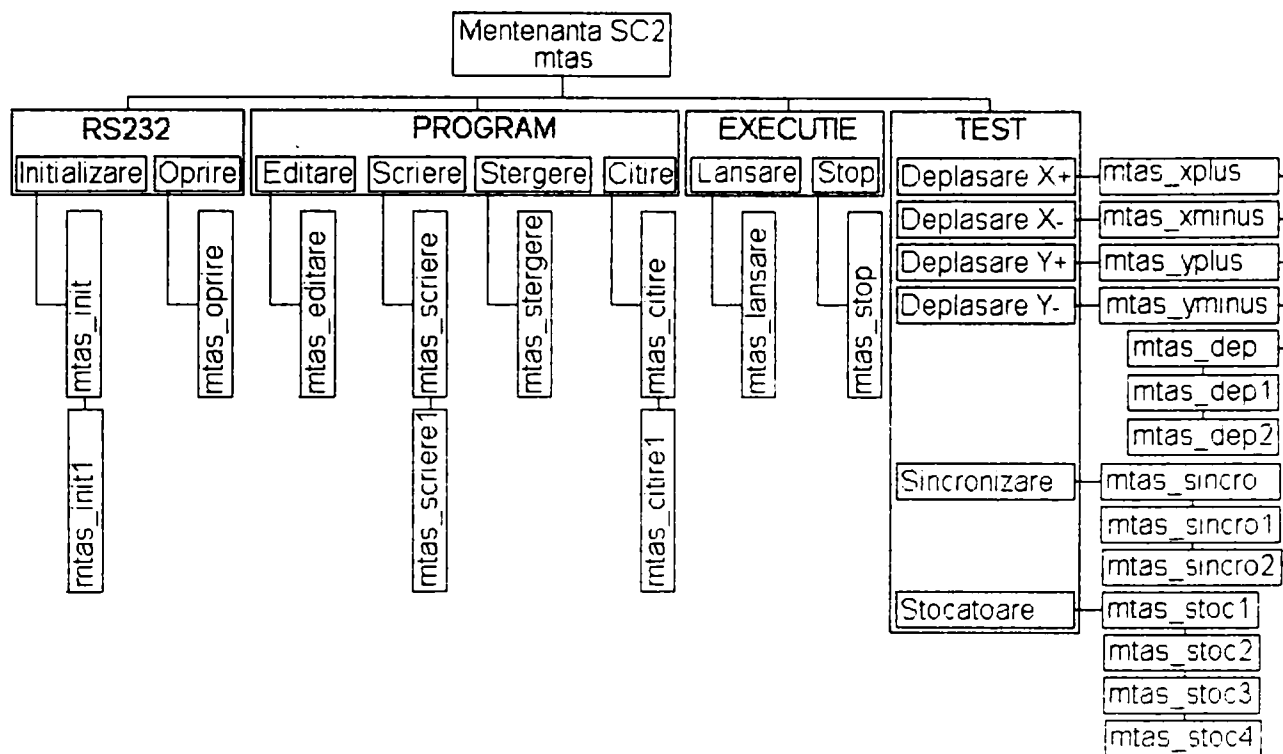


Fig.5.39.. Diagrama programului ..mtas”.

Modulul 'mtas_oprire'

Se afișează fereastra de mesaj ce solicită operatorului decuplarea micro-calculatorului de la sursă (figura 5.37) și în final, după decuplarea de către operator a micro-calculatorului de la sursă și confirmarea acesteia, se întrerupe comunicația serială.

Modulul 'mtas_editare'

Permite editarea programelor și salvarea acestora în format tip text pe hard-discul sistemului SC2.

Modulele 'mtas_scriere' și 'mtas_scriere1'

Modulul de scriere a programelor permite selectarea fișierului program dorit și realizează scrierea în mod automat a acestuia în memoria micro-calculatorului. În momentul începerii transmisiei de date se afișează mesajul 'Scriere....' iar la terminarea cu succes a transmiterii de date mesajul 'Scriere OK', dând operatorului certitudinea corectitudinii transmiterii datelor.

În figura 5.40 este prezentată fereastra de selectare a fișierului ce conține

programul controler.

Modulul 'mtas_stergere'

Modulul lansează comanda 'NEW' spre micro-calculatorului, determinând ștergerea oricărui program existent în memorie.

Modulele 'mtas_citire' și 'mtas_citire1'

Modulul realizează citirea din memoria micro-calculatorului a programului și scrierea acestuia într-un fișier de tip text cu specificarea numelui fișierului.

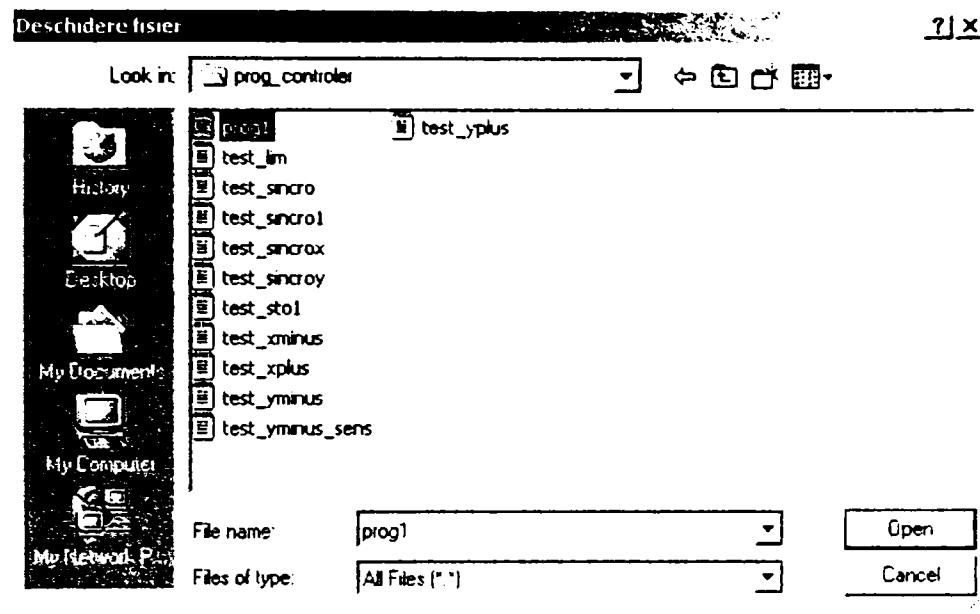


Fig.5.40. Interfața de selectare a fișierului

Modulul 'mtas_lansare'

Modulul transmite comanda 'RUN' către micro-calculatorului în vederea lansării în execuție a programului existent în memoria acestuia.

Modulul 'mtas_stop'

Modulul transmite micro-calculatorului valoarea '0' sau '1' comandând astfel oprirea programului ce rulează în micro-calculator.

Modulele 'mtas_xplus', 'mtas_xminus', 'mtas-yplus' și 'mtas_yminus'

Modulele realizează selectarea fișierului text ce conține programul aferent deplasării pe axele X și Y ale mesei microscopului, în sensurile plus și minus.

Modulele 'mtas_dep', 'mtas_dep1' și 'mtas_dep2'

Fișierul selectat cu ajutorul modulelor 'mtas_xplus', 'mtas_xminus', 'mtas-yplus' și 'mtas_yminus', este transmis la micro-calculator (fișierele text : 'test_xplus', 'test_xminus', 'test_yplus' și 'test_yminus'). După ce s-a realizat transmisia de date se lansează programul și se afișează interfața grafică (figura 5.41) ce permite introducerea numărului de pași ce trebuie realizați de către motorul pas cu pas.

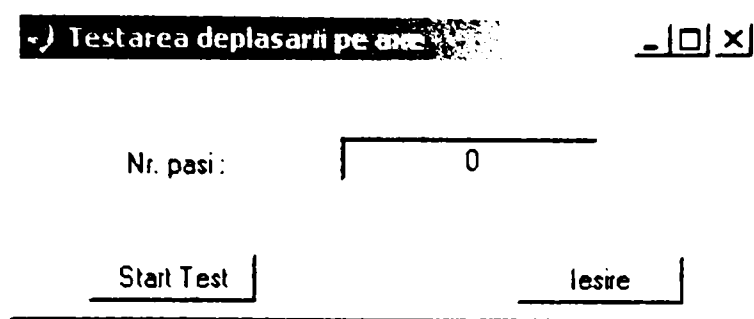


Fig.5.41. Intefața de introducere număr de pași.

Programul controler comandă motorul pas cu pas pe axa selectată, rularea terminându-se la realizarea numărului de pași specificați.

Modulele 'mtas_sincro', 'mtas_sincro1' și 'mtas_sincro2'

Modulele încarcă automat programul controler 'test_sincro' și îl lansează în execuție. Acesta realizează deplasarea mesei microscopului pe cele două axe până în punctul de schimbare eșantion (PSE), sesizat de către limitatoarele de capăt de cursă.

Modulele 'mtas_stoc1', 'mtas_stoc2', 'mtas_stoc3' și 'mtas_stoc4'

Modulele realizează transmiterea și lansarea în execuție a programului 'test_sto1' și afișează interfața grafică prezentată în figura 5.42. Programul citește intrările digitale de la portul A și le returnează sistemului SC2 care setează marcajele din cadrul interfeței grafice. Pentru valoarea '1' a bitului returnat se setează marcajul la culoarea roșie iar pentru valoarea '0' se setează culoarea fundalului.

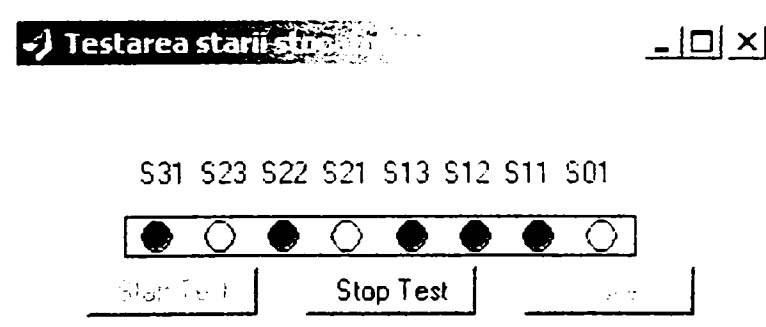


Fig.5.42. Intefața de testare a stării stocatoarelor.

Modulul 'm_ccd'

Modulul realizează testarea camerei CCD montate pe microscopul de analiză a microstructurii în mod similar cu cele descrise pentru modulul 'm_ccd' din cadrul sistemului SC1.

5.6. Procese decizionale în cadrul CFLM1.

5.6.1. Considerații generale

La nivelul programelor de aplicație sursă implementarea deciziilor se realizează prin construcțiile de tip „if....endif” descrise în paragraful anterior. După preprocesare (la nivelul programului de aplicație MATLAB) instrucțiunea if este întotdeauna precedată de către o instrucțiune ce actualizează parametrul specificat în cadrul clauzei „if „. Instrucțiunea de actualizare este generată automat de către preprocesor în cazul în care parametrul actualizat este o proprietate a unui obiect de tip *STO* sau *ROBOT* (starea de ocupare a unui stocator, starea gripper-ului robotului). Un exemplu în acest sens este dat de următoarele linii de cod :

```
if stocator11.status is empty
    incarcare(robot_h5,stocator11)
endif
if stocator12.status is empty
    incarcare(robot_h5,stocator12)
endif
if stocator13.status is empty
    incarcare(robot_h5,stocator13)
endif
```

În acest exemplu se realizează încărcarea primului stocator liber dintre stocatoarele 11,12,13 de către robot_h5. La tratarea instrucțiunilor condiționale din exemplul de mai sus, preprocesorul generează, în mod automat, linii de cod destinație ca de exemplu pentru ‚stocator11’:

```
[sto] = act_conf(sto,'sto','1',stop_program_flag);
if sto(1).status == 1
```

Funcția *act_conf()* realizează actualizarea proprietăților *status* ale obiectelor stocator, pentru ca instrucțiunea „if” să poată fi evaluată corect.

Există însă situații în care decizia nu poate fi luată doar în funcție de proprietatea unui singur obiect, fiind necesară procesarea mai multor date. Astfel de situații se vor analiza pe larg în acest paragraf, pentru fiecare din cele două subsisteme în parte (capitolul 5 figura 5.1.).

Pentru rezolvarea acestor situații la nivelul programului de aplicație în cadrul claselor informaționale definite, a fost inclusă și clasa deciziilor (DEC). Pentru tratarea deciziilor a fost definit specificatorul de execuție „eval”. Sintaxa de utilizare a deciziilor în limbajele de aplicație este :

```
execute/eval(<nume_obiect_dec>,<date>)
if <nume_obiect_dec>.status is ready
    if <nume_obiect_dec>.result is <spec_rezultat1>
```

```

        <lista_instrucțiuni>
    endif
    if <nume_obiect_dec>.result is <spec_rezultat2>
        <lista_instrucțiuni>
    endif

endif

```

În acest caz instrucțiunea de actualizare a parametrilor decizionali nu se generează în mod automat în cursul preprocesării ci trebuie specificată explicit în programul de aplicație sursă, prin instrucțiunea :

```
execute/eval(<nume_obiect_dec>,<date>)
```

în care <nume_obiect_dec> este denumirea modulului program ce elaborează decizia iar <date> este denumirea unui fișier ASCII ce conține parametrii pentru care este evaluată decizia.

5.6.2. Tratarea deciziilor în subsistemul 1.

Deoarece în acest subsistem există doar obiecte din clasa PAR, fluxul eșantioanelor este liniar : ordinea de ieșire a eșantioanelor din subsistem este aceeași cu ordinea de intrare a acestora (FIFO). Pe toată durata procesărilor la diferitele posturi eșantion se află în dispozitivul de prehensiune al robotului. Procesul decizional care prezintă importanță în acest caz se referă la selectarea discului abraziv pe care se va realiza șlefuirea și la durata șlefuirii. Acești parametrii se aleg pe baza analizei imaginii suprafeței eșantionului în funcție de calitatea acesteia (rugozitate) și în funcție de duritatea materialului eșantionului.

Duritatea materialului eșantionului influențează în primul rând durata șlefuirii în sensul că o duritate ridicată va implica alegerea unor durate de șlefuire mai mari.

Rugozitatea suprafeței influențează deasemenea durata șlefuirii (rugozitatea mare implică o durată de șlefuire mare), dar la rugozități mari (peste $R_a = 1,6$), tocmai pentru a scurta durata de șlefuire, se va utiliza șlefuirea pe un disc abraziv cu o granulație grosieră („șlefuire eboș”). Modul în care se realizează decizia în acest caz este descris în Capitolul 8.

Pentru ilustrarea algoritmului decizional se consideră un sistem în care există două unități de șlefuire (ca și în cazul CFLM1) și anume o unitate dotată cu hârtie abrazivă de ganulație „mare” și o unitate dotată cu abraziv cu granulație mică, și că pentru fiecare unitate se definesc două durate de șlefuire : o durată „îndelungată” și una „scurtă”. Granulațiile (respectiv numărul de unități de șlefuire, cât și duratele de șlefuire pot fi mai multe decât două dar aceasta nu schimbă cu nimic logica tratării problemei. Algoritmul de alegere al granulației respectiv al duratei de șlefuire se prezintă în schema din figura 5.43.

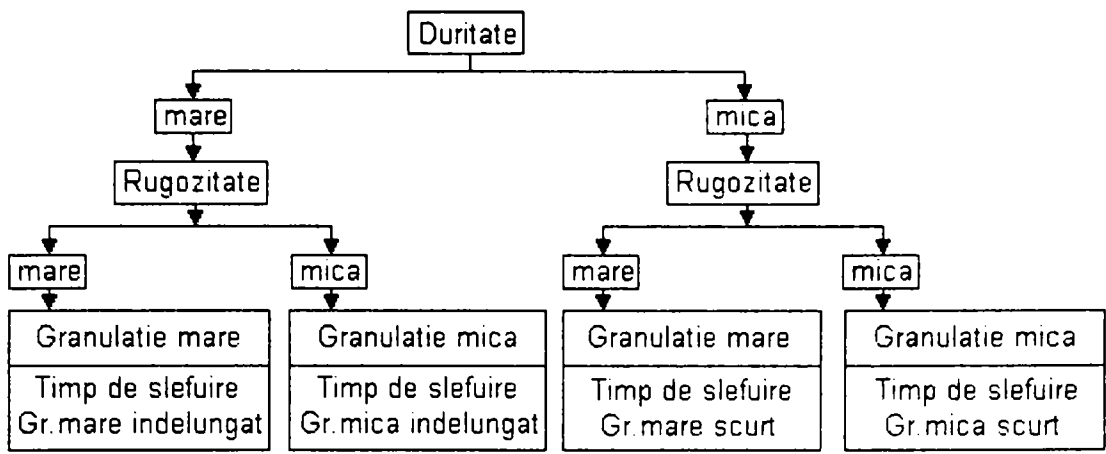


Fig. 5.43. Schema algoritmului de selectare a granulației și duratei de șlefuire în funcție de duritate și rugozitate.

Atributele utilizate („mare”, „mică”, „înelungată”, „scurtă”) pot fi cuantificate, pe domenii convenabile, realizându-se astfel un sistem fuzzyficabil, cu observația că timpii de șlefuire sunt diferiți pentru granulația mare față de cea mică.

5.6.3. Tratarea deciziilor în subsistemul 2.

5.6.3.1. Considerații generale.

În urma studiului experimental al funcționării CFLM1 s-au identificat două probleme ce necesită aplicarea unui algoritm decizional :

- selectarea eșantionului ce urmează a fi procesată în subsistemul 2;
- selectarea reactivului cu care trebuie tratat un anumit eșantion;

Logica de tratare a fluxului de eșantioane în cazul subsistemului 2 (figura 5.44) este stabilită în funcție de stările postului de analiză metalografică M2. Pornind de la aceasta programul de aplicație testează în primul rând stările postului M2, apoi starea stocatoarelor din grupa 2 (St21,St22,St23) și în cele din urmă poate să decidă care dintre stocatoare să fie descărcat (care este eșantionul cu prioritate maximă).

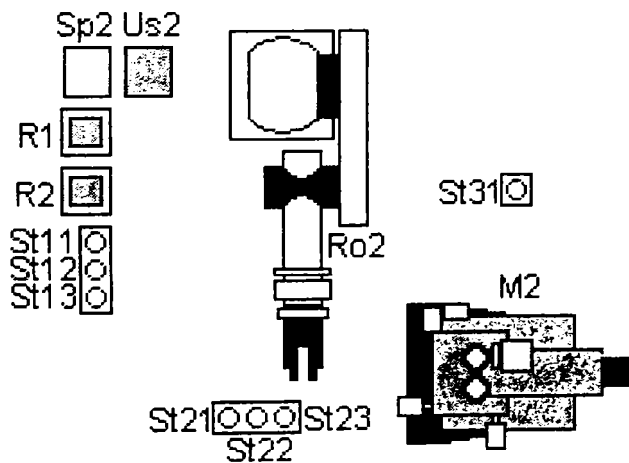


Fig.5.44. Schema subsistemului 2 al CFLM1.

Stările postului M2 sunt definite prin proprietățile „load_status” (stare de

încărcare) și „active_status” stare de activitate, ce pot lua valorile arătate în tabelul 5.1. Funcționarea subsistemului 2 se realizează pe baza analizei următoarelor situații :

- dacă pentru M2 este valabil $load_status = full$ și $active_status = ready$ se descarcă eșantion de pe M2 și se transferă la stocatorul de ieșire St31;
- dacă pentru M2 este valabil $load_status = full$ și $active_status = busy$ se testează starea stocatoarelor tampon :
 - dacă pentru St21 sau St22 sau St23 $status = empty$ se va lua un eșantion de pe unul din stocatoarele St11, St12, St13 și se va încărca stocatorul liber din grupa 2;
 - dacă pentru St21 și St22 și St23 $status = full$ se așteaptă terminarea procesării la M2;
- dacă pentru M2 este valabil $load_status = empty$ și $active_status = ready$ se va lua un eșantion de pe unul din stocatoare și se va încărca M2;

Decizia asupra ordinii în care se procesează eșantioanele din cele două grupe este descrisă în cele ce urmează.

5.6.3.2. Stabilirea ordinii de procesare a eșantionului

În subsistemul 2 există un obiect din clasa PIR (microscopul pentru analiza microstructurii) și stocatoare tampon care implică posibilitatea schimbării ordinii de procesare.

Cea mai simplă abordare a ordonării fluxului de eșantioane este în mod evident similară cu cea din subsistemul 1, și anume să se respecte regula ca ordinea de ieșire a eșantioanelor din subsistem să fie aceeași cu cea de intrare. În practică, apar situații în care trebuie luată în considerare stabilirea unor priorități a procesării eșantioanelor. Pot fi satisfăcute astfel mai bine cererile de urgență ale unor clienți. Pentru rezolvarea acestor cazuri a fost definită proprietatea „probe_priority” în care la intrarea unui eșantion în sistem se înscrie o valoare a priorității inițiale ($PI = 0..1$, - cu cât urgența este mai mare PI este mai aproape de valoarea 1). În consecință, a doua posibilitate de ordonare se poate stabili în funcție de PI a fiecărui eșantion.

În urma studiului funcționării subsistemului 2 se constată însă că dacă se realizează ordonarea numai în funcție de PI pot exista eșantioane cu PI foarte scăzută, dar pentru care timpul de procesare în subsistemul 2 este relativ scurt, ce pot fi reținute în sistem pe o durată nejustificat de lungă. Pentru a rezolva această problemă trebuie aplicat un algoritm de decizie ce să ia în considerare atât PI cât și timpul rămas până la ieșirea eșantionului din sistem (TR).

Un algoritm ce poate rezolva decizia multicriterială (criteriile PI și TR), este algoritmul bazat pe teoria mulțimilor fuzzy, aplicat în maniera prezentată în figura 5.45.

Rezultatul deciziei fuzzy este prioritatea operativă PO pe baza căreia se stabilește ordinea de procesare (va fi procesat eșantionul cu prioritate operativă maximă). Valoarea PO va fi înscrisă în proprietatea obiectului DEC , $dec(1).result$.

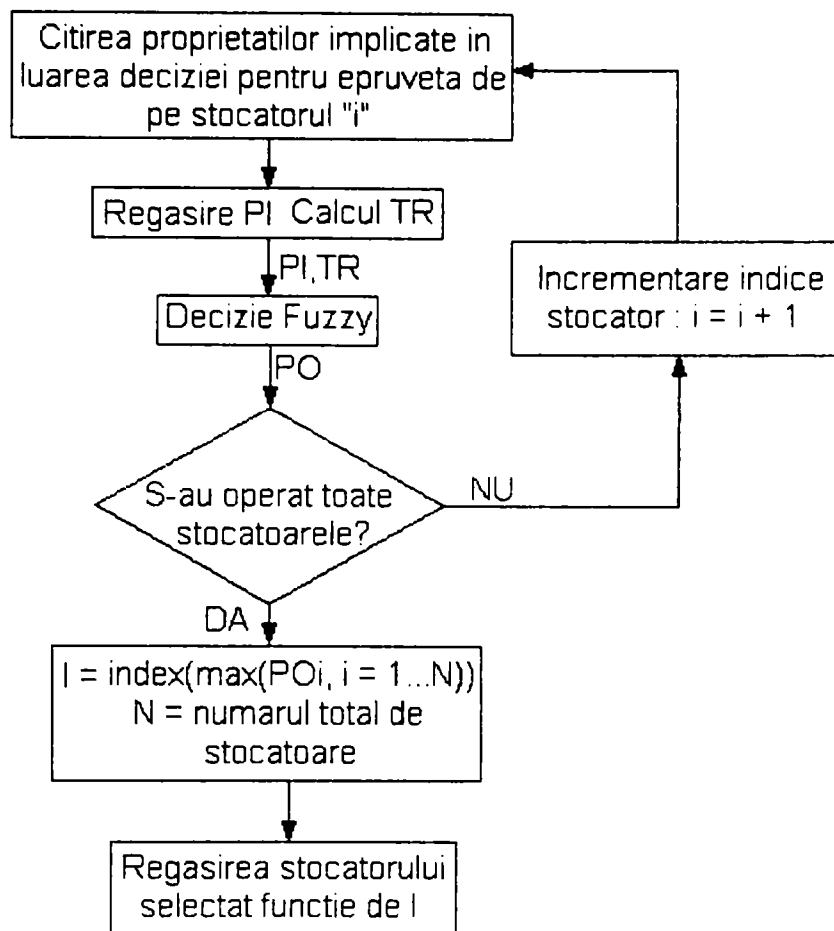


Fig. 5.45. Schema logică a aplicării algoritmului de decizie fuzzy.

Timpul rămas TR de procesare a eşantioanelor se calculează însumând duratele de procesare la fiecare post de lucru la care urmează să fie procesat eşantionul. Specificarea duratelor de procesare a eşantionului la diferitele posturi de lucru se realizează prin declararea (la intrarea eşantionului în sistem), a unui anumit itinerar al acestuia. Itinerariile declarate se concretizează sub forma unor şabloane (*template*), ce la nivel informaţional reprezintă tablouri cu trei dimensiuni, definite pentru obiectele de tip PAR şi PIR şi pentru specificatorii „assist” şi „job” (ANEXA 5.2.1. matricile „temp” şi matricile „metoda”). Indexarea acestor tablouri se realizează prin următorii parametrii :

- pentru specificator „assist” :

assist_par(i_par, i_template, i_robot)

în care : „i_par” – numărul obiectului PAR; „i_template” – numărul şablonului; i_robot - numărul robotului care asistă prelucrarea;

- pentru specificator „job” :

job_par(i_par, i_template)

în care : „i_par” – numărul obiectului PAR; „i_template” – numărul şablonului.

Pentru calculul valorii TR şi regăsirea valorii PI a eşantionului se porneşte de la instrucţiunea din codul sursă :

execute/eval(optimizare, date1)

în care *optimizare* este denumirea modulului program ce elaborează decizia, (în acest caz modulul „com_optimizare”) iar *date1* este denumirea fișierului ASCII ce conține denumirea stocatoarelor implicate în procesul decizional.

Algoritmul de calcul se realizează în următorii pași :

1. se citește linia din fișierul „date1”;
2. se aplică subrutina „name2struct” și se regăsește indicele „i_sto”, numărul obiectului stocator curent;
3. se citește *sto(i_sto).probe_priority* regăsind astfel prioritatea inițială a eșantionului.
4. se declară *i_robot = 2* ;
5. se citește variabila *sto(i_sto).probe_template* – pentru regăsirea numărului șablonului eșantionului, *i_template*;
6. se calculează suma timpilor de prelucrare rămași astfel :
 - dacă $2 \leq i_sto \leq 4$ (stocatoare din grupa 1) :

$$TR = t_assist_par(8, i_template, i_robot) + t_assist_par(9, i_template, i_robot) + t_job_pir(1, i_template)$$
 - dacă $5 \leq i_sto \leq 7$ (stocatoare din grupa 2) :

$$TR = t_job_pir(1, i_template)$$
7. se lansează modulul fuzzy pentru calculul priorității operative PO.
8. se iterează linia din fișier și se reia algoritmul de la punctul 1. până când se epuizează toate liniile din fișierul de date.
9. se regăsește prioritatea operativă maximă și stocatorul corespunzător acesteia.

După terminarea modulului „com_optimizare” se poate trece la preluarea eșantionului de pe stocatorul selectat.

Algoritmul fuzzy elaborat trebuie să răspundă cerințelor de flexibilitate în așa fel încât să se poată adapta la modificări structurale (modificarea numărului de stocatoare) și mai cu seamă la modificări de concepție (de exemplu introducerea unor noi criterii de optimizare). Pentru a răspunde acestor cerințe, s-a realizat descrierea algoritmului pentru cazul general și apoi s-a particularizat pentru aplicația prezentată.

Descrierea algoritmului fuzzy de optimizare pentru cazul general.

Criteriile în raport cu care se realizează determinarea priorităților formează mulțimea criteriilor de evaluare:

$$C = \{C_1, C_2, \dots, C_j, \dots, C_m\} \quad (5.1)$$

Pot fi luate în discuție, la un moment dat, mai multe criterii de evaluare. Aceste criterii vor fi definite ca fiind *intrările* în sistemul decizional. Intrările reflectă starea CFLM1 la momentul în care se pune problema stabilirii ordinii de procesare a eșantioanelor și caracterizează fiecare eșantion care solicită procesarea în sistem.

Fiecărui criteriu i se asociază un domeniu de variație, în interiorul căruia se pot regăsi valorile specifice fiecărui eșantion care face obiectul procesului decizional. Aceste domenii de valori sunt :

$$\begin{aligned} C_1 : D_1 &= [L_1^{\text{inf}}, L_1^{\text{sup}}] \\ &\vdots \\ C_j : D_j &= [L_j^{\text{inf}}, L_j^{\text{sup}}] \end{aligned} \quad (5.2)$$

$$C_m : D_m = [L_m^{\text{inf}}, L_m^{\text{sup}}]$$

unde, $L_j^{\text{inf}}, L_j^{\text{sup}}$ sunt limita inferioară respectiv limita superioară a domeniului de valori asociat criteriului $C_j, j = \overline{1, m}$.

Fiecărui criteriu de evaluare i se asociază o *variabilă lingvistică*. Astfel criteriul C_j devine variabila lingvistică $C_j, j = \overline{1, m}$.

Pentru fiecare variabilă lingvistică asociată unei mărimi de intrare se definesc *gradele lingvistice* [ȘOFRON] sau *termenii lingvistici* [PREITL]. Acestea servesc la caracterizarea “vagă” a informațiilor ferme. Mulțimile gradelor lingvistice asociate fiecărei variabile lingvistice sunt de forma:

$$\begin{aligned} C_1 : GL_1^C &= \{GL_{11}^C, GL_{12}^C, \dots, GL_{1K}^C\} \\ C_j : GL_j^C &= \{GL_{j1}^C, GL_{j2}^C, \dots, GL_{jK}^C\} \end{aligned} \quad (5.3)$$

$$C_m : GL_m^C = \{GL_{m1}^C, GL_{m2}^C, \dots, GL_{mK}^C\}$$

Fiecărui grad lingvistic, corespunzător unei variabile lingvistice, i se asociază o *funcție de apartenență*:

$$\begin{aligned} C_1 \rightarrow GL_1^C &\rightarrow FA_1^C = \{fa_{11}^C, fa_{12}^C, \dots, fa_{1K}^C\} \\ C_j \rightarrow GL_j^C &\rightarrow FA_j^C = \{fa_{j1}^C, fa_{j2}^C, \dots, fa_{jK}^C\} \\ &\vdots \\ C_m \rightarrow GL_m^C &\rightarrow FA_m^C = \{fa_{m1}^C, fa_{m2}^C, \dots, fa_{mK}^C\} \end{aligned} \quad (5.4)$$

Mărimea de ieșire din procesul decizional va fi *prioritatea operativă* asociată fiecărui eșantion $i, i = \overline{1, n}$, aflată în așteptare. Prioritatea operativă va fi determinată prin tehnici specifice mulțimilor fuzzy. Vectorii priorităților operative sunt:

$$PO = \{PO_1, \dots, PO_i, \dots, PO_n\} \quad (5.5)$$

Domeniile de valori ale mărimilor de ieșire sunt următoarele:

$$PO_i : D_i^{\text{out}} = [0, 1]; \quad i = \overline{1, n}; \quad (5.6)$$

Fiecărei mărimi de ieșire: *prioritate operativă* a eșantionului i , i se asociază o *variabilă lingvistică*. Astfel mărimea de ieșire PO_i , $i = \overline{1, n}$, devine variabila lingvistică *prioritatea operativă a eșantionului i* : PO_i , $i = \overline{1, n}$,

Pentru fiecare variabilă lingvistică asociată mărimilor de ieșire se definesc *gradele lingvistice (termeni lingvistici)*. Acestea caracterizează “vag” informațiile rezultate în urma procedurilor de inferență. Mulțimile gradelor lingvistice asociate fiecărei variabilele lingvistice de tipul *prioritatea operativă a eșantionului i* , sunt de forma:

$$PO_i \rightarrow GL^{PO_i} = \left\{ GL_1^{PO_i}, GL_2^{PO_i}, \dots, GL_k^{PO_i} \right\} \quad (5.7)$$

Fiecărui grad lingvistic corespunzător unei variabile lingvistice care descrie o mărime de ieșire i se asociază o *funcție de apartenență*. Pentru gradele lingvistice corespunzătoare variabilei lingvistice PO_i , $i = \overline{1, n}$, mulțimea funcțiilor de apartenență este:

$$PRIORI_i \rightarrow GL^{PO_i} \rightarrow FA^{PO_i} = \left\{ fa_1^{PO_i}, fa_2^{PO_i}, \dots, fa_l^{PO_i}, \dots, fa_k^{PO_i} \right\}; i = \overline{1, n}; l = \overline{1, k} \quad (5.8)$$

Mulțimea variabilelor lingvistice și a gradelor lingvistice, cărora li s-au asociat funcții de apartenență, caracterizează „vag”, valorile ferme ale mărimilor de intrare, respectiv ale mărimilor de ieșire. Conectarea acestora se face prin reguli de inferență care formează *mașina de inferență*. Regulile de inferență sunt de forma:

$$DACĂ (premiza) ATUNCI (concluzia) \quad (5.9)$$

Premiza – este o proprietate constatată, rezultată în urma conectării, prin proceduri specifice teoriei mulțimilor fuzzy, a diverselor grade lingvistice asociate variabilelor lingvistice corespunzătoare mărimilor de intrare. În cazul procedurilor de ordonare care urmează să fie descrise s-a folosit conectorul „ȘI”.

Concluzia – este proprietatea afirmată și este exprimată prin grade lingvistice asociate variabilelor lingvistice corespunzătoare mărimilor de ieșire.

Pentru cazul determinării priorităților operative baza de reguli (mașina de inferență) este de forma:

$$\begin{aligned} RIN_1^{PO_i} : & \text{DACA } (C_1 = GL_{11}^C \text{ SI } \dots \text{ SI } C_m = GL_{1m}^C) \text{ ATUNCI } (PO_i = GL_1^{PO_i}) \text{ SAU} \\ & \vdots \\ RIN_{in}^{PO_i} : & \text{DACA } (C_1 = GL_{i1}^C \text{ SI } \dots \text{ SI } C_m = GL_{im}^C) \text{ ATUNCI } (PO_i = GL_i^{PO_i}) \text{ SAU} \quad (5.10) \\ & \vdots \\ RIN_{IN}^{PO_i} : & \text{DACA } (C_1 = GL_{k1}^C \text{ SI } \dots \text{ SI } C_m = GL_{km}^C) \text{ ATUNCI } (PO_i = GL_k^{PO_i}) \end{aligned}$$

Dacă în cazul general sunt n variabile lingvistice de intrare, fiecărei variabile fiindu-i asociate k grade lingvistice, acestea se pot combina în k^n reguli de inferență (baza de reguli sau mașina de inferență). Adică $IN = k^n$, respective $in = \overline{1, IN}$.

Din relația (5.10) se observă că regulile de inferență sunt conectate la rândul lor prin operatorul fuzzy „SAU”. Modul de agregare al variabilelor de intrare (operatorul „SAU”), respectiv al regulilor de inferență (operatorul „ȘI”), corespunde compunerii *MIN-MAX* a relațiilor fuzzy [PREITL, ȘOFRON].

În aplicațiile practice, la un moment dat, numai anumite reguli de inferență sunt active, adică mărimile de intrare au valori ferme care fuzificate pot fi descrise cu ajutorul unui număr mai mic de grade lingvistice decât cel alocat pentru variabila lingvistică de intrare în cauză.

Rezultatul inferenței fuzzy este o informație fuzzy (vagă) sub forma funcției de apartenență “rezultat”. În cazul stabilirii priorității operative a probei i aceasta este:

$$fa_{rez}^{PO_i}(PO_i); \quad i = \overline{1, n}; \quad (5.11)$$

Prin defuzificare se înțelege operația de obținere a unei valori ferme a mărimii de ieșire, pe baza funcției de apartenență “rezultat” a inferențelor fuzzy.

Astfel pentru prima categorie de mărimi de ieșire este:

$$fa_{rez}^{PO_i}(PO_i) \xrightarrow{\text{DEFUZIFICARE}} PO_i; \quad PO_i \in D_i^{outl}; \quad i = \overline{1, n}; \quad (5.12)$$

Din multitudinea de metode de defuzificare existente, se utilizează *metoda centrului de greutate*, metoda cea mai aplicată în practică.

Principiul metodei constă în determinarea valorii ferme a mărimii de ieșire luând în considerare (în mod ponderat) toate influențele rezultate în urma activării regulilor de inferență de către valorile ferme ale mărimilor de intrare, la un moment dat.

Relația care stă la baza metodei centrului de greutate este:

$$PO_i = \frac{\int_{D_i^{outl}} PO_i \cdot fa_{rez}^{PO_i}(PO_i) \cdot dPO_i}{\int_{D_i^{outl}} fa_{rez}^{PO_i}(PO_i) \cdot dPO_i} \quad (5.13)$$

Valorile ferme ale mărimilor de ieșire rezultate în urma procedurii de defuzificare constituie informațiile care stau la baza ordonanțării propriu-zise a eșantioanelor în CFLM1.

Determinarea priorităților operative.

Criteriile în raport cu care se face determinarea priorităților operative sunt:

1. Prioritatea Inițială (PI);
2. Timpul de procesare Rămas (TR).

Astfel mulțimea criteriilor este:

$$C = \{PI, TR\} \quad (5.14)$$

Pentru criteriile stabilite domeniile de valori sunt următoarele:

$$\begin{aligned} PI : D_{PI} &= [0,1] \\ TR : D_{TR} &= [1;5.5] \quad [min] \end{aligned} \quad (5.15)$$

Criteriului de evaluare *Prioritate initiala* i se asociază variabila lingvistică de intrare *Prioritate initiala* (PI) iar criteriului *Timp de procesare rămas* i se asociază variabila lingvistică de intrare *Timp de procesare rămas* (TR).

Variabilei lingvistice *Prioritate inițiala* (PI) îi corespund următoarele grade lingvistice: *foarte mica* -fm; *mica* - m; *Medie* - Md; *Mare* - M; *Foarte Mare* - FM.

Variabilei lingvistice *Timp de procesare ramas* (TR) îi corespund gradele lingvistice: *foarte mic* -fm; *mic* - m; *Mediu* - Md; *Mare* - M; *Foarte Mare* -FM.

Funcțiile de apartenență asociate fiecărui grad lingvistic în cazul variabilelor lingvistice de intrare sunt de tip triunghiular.

Caracteristicile variabilelor lingvistice de intrare sunt prezentate în figura 5.46.

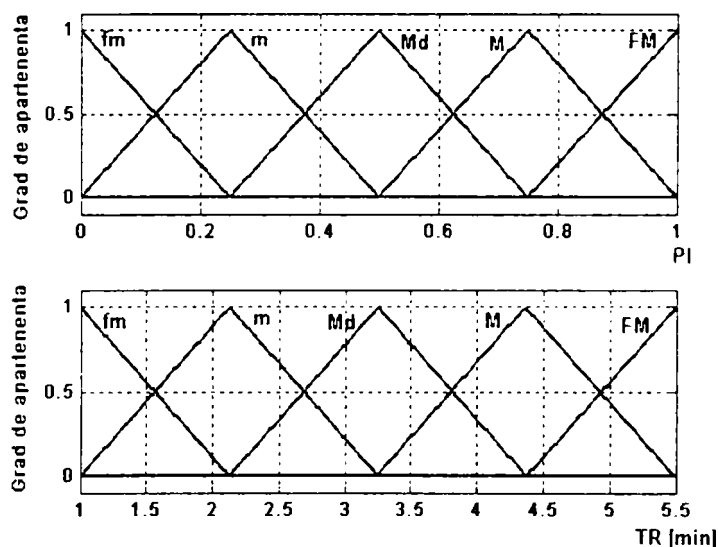


Fig.5.46. Caracteristicile variabilelor lingvistice de intrare

Mărimea de ieșire din procesul decizional va fi *prioritatea operativă* asociată fiecărui eșantion i , $i = \overline{1, n}$, aflată în așteptare.

Domeniul de valori al priorității operative pentru fiecare eșantion i care face obiectul procedurii decizionale este :

$$PO_i : D_{PO_i} = [0,1]; \quad i = \overline{1, n}; \quad (5.16)$$

Fiecărei mărimi de ieșire: *prioritate operativă* a probei i , i se asociază o *variabilă lingvistică de ieșire*. Astfel mărimea de ieșire PO_i , $i = \overline{1, n}$, va deveni

variabila lingvistică *prioritatea operativă a probei i*: $PO_i, i = \overline{1, n}$,

Gradelor lingvistice asociate fiecărei variabile lingvistice de tipul prioritatea operativă a probei i le sunt asociate funcții de apartenență. Acestea sunt: $mf1$ – foarte foarte mică; $mf2$ – foarte mică; $mf3$ – mică; $mf4$ – Medie; $mf5$ - Mare; $mf6$ – Foarte Mare; $mf7$ – Foarte Foarte Mare.

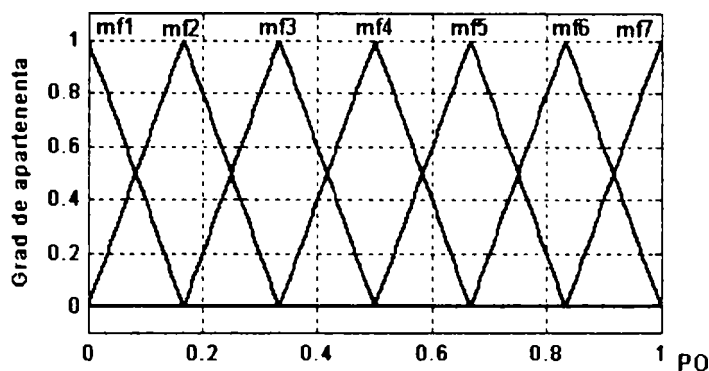


Fig.5.47. Caracteristicile variabilei lingvistice de ieșire

Funcțiile de apartenență asociate gradelor lingvistice corespunzătoare variabilelor de ieșire sunt deasemenea de tip triunghiular. Caracteristicile variabilei lingvistice de ieșire, prioritate operativă, sunt prezentate în figura 5.47.

Pentru cazul determinării priorităților operative *baza de reguli, mașina de inferență*, va fi de forma:

1. If (PI is fm) and (TR is fm) then (PO is mf4)
 2. If (PI is fm) and (TR is m) then (PO is mf3)
 - ⋮
 12. If (PI is Md) and (TR is m) then (PO is mf5)
 13. If (PI is Md) and (TR is Md) then (PO is mf4)
 - ⋮
 24. If (PI is FM) and (TR is M) then (PO is mf5)
 25. If (PI is FM) and (TR is FM) then (PO is mf4)
- (5.17)

Compunerea relațiilor fuzzy s-a făcut prin metoda *MIN-MAX*. Sistemul decizional bazat pe mulțimi fuzzy a fost implementat în Matlab Fuzzy Logic Toolbox®, sub forma unui modul de tip fis: *DECID57.fis*. Acest modul este prezentat schematic în figura 5.48.

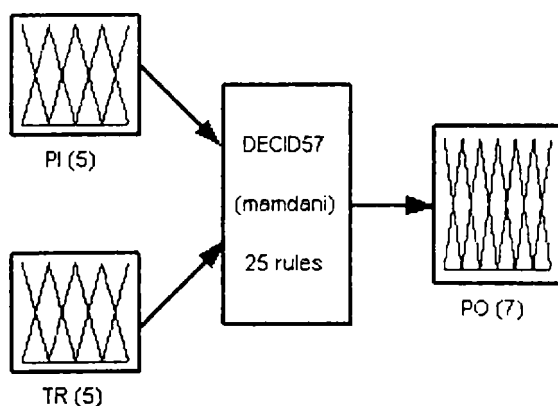


Fig. 5.47. Sistemul DECID57, implementat în Matlab

Suprafața de variație a mărimii de ieșire în funcție de mărimile de intrare este prezentată în figura 5.49.

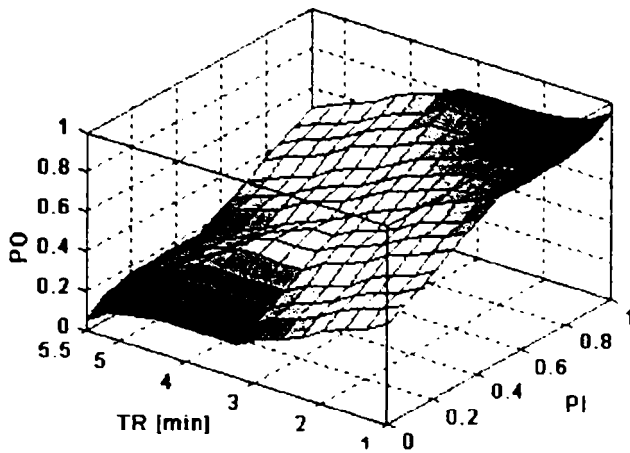


Fig.5.49. Suprafața de variație a mărimii de ieșire în funcție de mărimile de intrare

În figura 5.50 sunt prezentate regulile de inferență și corelația acestora cu funcțiile de apartenență care caracterizează mărimile de intrare, respectiv mărimea de ieșire.

Metoda de defuzificare utilizată este *metoda centrului de greutate*.

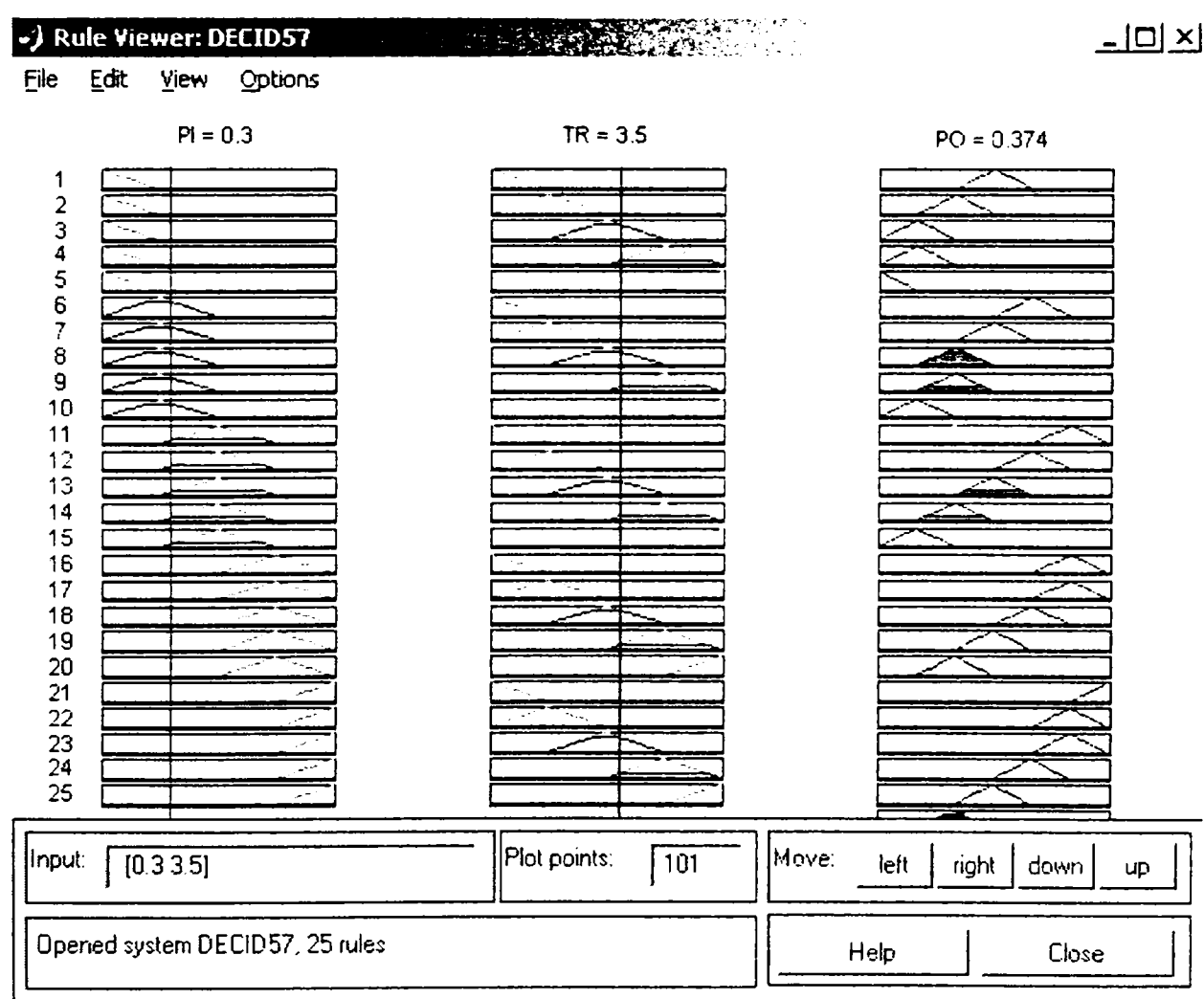


Fig. 5.50. Regulile de inferență

În figura 5.50 se observă că pentru o prioritate inițială mică ($PI = 0,3$) și un timp de procesare rămas mediu ($TR = 3,5$) prioritatea operativă rezultată în urma deciziei fuzzy va fi $PO = 0,374$.

5.6.3.3. Selectarea reactivului

În CFLM1 a fost prevăzută posibilitatea tratării eşantioanelor cu diferiți reactivi, răspunzând astfel cerințelor reale de pregătire a eşantioanelor metalografice.

Mecanismul de realizare a deciziei pentru selectarea reactivului se rezumă la regăsirea valorii corespunzătoare șablonului eşantionului în tabloul *assist_par* (ANEXA 5.2.1., matrice „metoda”, specificator de execuție „assist” pentru robot_v6). Se selectează acel reactiv pentru care valoarea regăsită în tablou este diferită de 0.

Se pornește de la instrucțiunea din codul sursă ce solicită căutarea reactivului :

execute/eval(reactiv,date2)

în care *reactiv* este denumirea modulului program ce elaborează decizia, în acest caz modulul „com_reactiv” iar *date2* este denumirea fișierului ASCII ce conține parametrii „nital” și „persulfat” (denumirea formală a posturilor de atac cu reactivi R1 respectiv R2) .

Algoritmul de selectare se realizează în următorii pași :

1. se citește linia din fișierul „date2”;
2. se lansează subrutina „name2struct” și se regăsește indicele „i_par”, numărul obiectului reactiv;
3. se declară *i_robot = 2* – selectarea reactivului implică asistența robotului cu indentificatorul 2;
4. se citește variabila *robot(2).probe_template* – pentru regăsirea șablonului eşantionului;
5. dacă *assist_par(i_par,i_template,i_robot)* are o valoare diferită de „0” atunci se declară rezultatul selectării reactivului *dec(2).result = <nume_obiect_par>*.
6. se iterează linia din fișier și se reia algoritmul de la punctul 1 până când se epuizează toate liniile din fișierul „date2”.

După terminarea modulului „com_reactiv” se rulează subrutina „select_reac” (ANEXA 5.1.), ce comandă robotul la postul selectat.

5.7. Modelarea celulei flexibile CFLM1

5.7.1. Considerații generale.

Pentru rezolvarea problemelor de proiectare și concepție constructivă s-au realizat pe de o parte schema bloc și matricile de structură iar pe de altă parte modelul cu rețele Petri al celulei flexibile CFLM1.

5.7.2. Schema bloc și matricile de structură

Matricea de structură a unui sistem cu n subsisteme (componente) este o matrice având ca elemente matricile de cuplare ale subsistemelor. Matricea de structură a unui sistem caracterizează în ansamblu mulțimea legăturilor între elementele componente (subsisteme) ale sistemului considerat.

Matricea de structură se realizează pe baza *schemei bloc* și are ca elemente matricile de cuplare ale subsistemelor componente. Schema bloc și matricile de cuplare sunt prezentate în ANEXA 5.3.1.

Schema bloc, matricile de structură și cele de cuplare, servesc atât pentru *analiza* funcționării celulei flexibile, cât și la sinteza celulei flexibile proiectate, permițând stabilirea subsistemelor componente și a legăturilor necesare între acestea. Sinteza structurală stă la baza concepției sistemului de comandă a CFLM1 și a programelor de comandă a acestuia.

5.7.3. Modelarea funcționării cu ajutorul rețelelor Petri.

O Rețea Petri (RP) este un model grafic de tipul grafurilor orientate, care utilizează două categorii de noduri:

- *Poziții* - ce modelează condițiile care se impun pentru realizarea unui anumit eveniment (reprezentate prin cercuri);
- *Tranziții* - ce modelează evenimentele care se desfășoară dacă sunt îndeplinite anumite condiții (reprezentate prin linii orizontale sau prin dreptunghiuri).

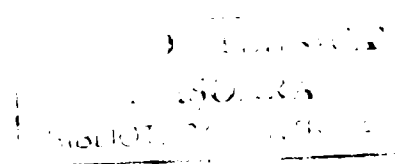
Pozițiile și tranzițiile sunt legate prin *arce*. Arcele sunt *orientate*, ele legând o poziție la o tranziție sau o tranziție la o poziție.

RP *temporizate* permit modelarea unui sistem a cărui funcționare depinde de timp.

O rețea Petri temporizată permite descrierea unui sistem a cărui funcționare depinde de timp. Această categorie de RP sunt utilizate pentru evaluarea performanțelor unui sistem.

Există două posibilități de modelare a temporizării: temporizările sunt asociate pozițiilor - RP P- temporizate; temporizările sunt asociate tranzițiilor - RP T- temporizate.

O rețea Petri T - temporizată este un dublet de forma $\langle R, Tempo \rangle$, unde: R



este o rețea Petri ce utilizează marcaje ; *Tempo* este o aplicație a mulțimii tranzițiilor T în mulțimea numerelor raționale sau nule, definită prin relația :

$$Tempo(T_j) = d_j \quad (5.18)$$

în care d_j este temporizarea asociată tranziției T_j .

În cazul RP T - temporizate, un marcaj poate fi într-unul din următoarele două stadii:

- rezervat pentru execuția tranziției T_j ;
- nerezervat.

Aceste ipostaze ale unui marcaj sunt ilustrate în figura 5.51.

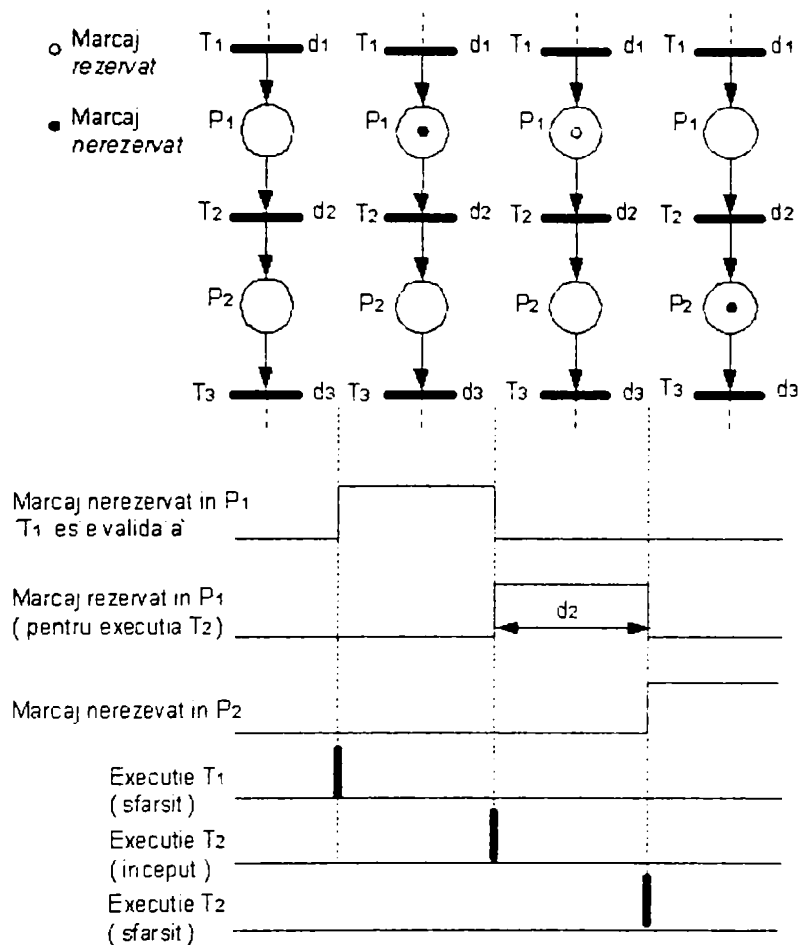


Fig. 5.51. RP T- temporizată.

După ce tranziția T_1 a fost executată, un marcaj a fost depus în poziția P_1 . Din acel moment tranziția T_2 este validată, executarea acesteia putându-se începe oricând. O dată începută executarea tranziției T_2 , marcajul necesar executării acesteia este rezervat. După ce s-a consumat durata d_2 , temporizarea asociată tranziției T_2 , marcajul este retras din poziția P_1 și este depus în poziția P_2 .

La un moment oarecare t , marcajul M este:

$$M = M' + M'' \quad (5.19)$$

În care M este marcajul rezervat și M'' este marcajul nerezervat.

O tranziție este validată pentru marcajul M , dacă este validată pentru marcajul M'' . Dacă t este momentul în care se decide executarea tranziției T_j și $t + d_j$, momentul în care tranziția a fost efectiv realizată, se consideră : t - începutul tranziției; $t + d_j$ - sfârșitul tranziției.

Pentru RP T - temporizate se pot defini două modalități de funcționare:

1. Funcționare cu viteză maximă - de îndată ce o tranziția este validată, marcajele necesare execuției sale sunt rezervate.
2. Funcționare cu viteză proprie - un marcaj depus într-o poziție devine rezervat pentru execuția unei tranziții aflate în aval, după un interval de timp care este diferit de zero (marcajul depus în poziția P_j - figura 5.52).

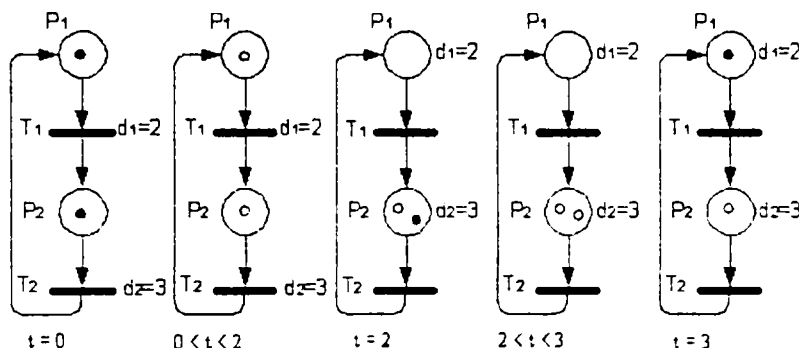


Fig. 5.52. Funcționarea unei RP T- temporizate cu viteză maximă

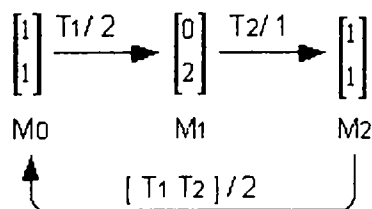
Funcționarea cu viteză maximă, a unei RP T- temporizate, este prezentată în figura 5.52.

La momentul inițial ($t= 0$) tranzițiile T_1 și T_2 sunt validate și poate începe execuția lor.

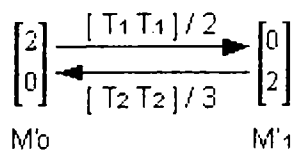
Pentru $M_1 = (1, 1)$ marcajele din pozițiile P_1 și P_2 devin rezervate.

După două unități de timp, tranziția T_1 este executată și marcajul este depus în poziția P_2 , fiind imediat rezervat pentru executarea tranziției T_2 . Pentru $2 < t < 3$ sunt rezervate două marcaje pentru două execuții ale tranziției T_2 .

Graful marcajelor este următorul:



Dacă marcajul inițial ar fi $M_0' = (2, 0)$, funcționarea ar fi cea descrisă de graficul de marcaje următor:



Definirea elementelor modelării cu Rețele Petri a CFLMI este prezentată în tabelul 1 din ANEXA 5.3.2, iar aspectele simulării sunt prezentate în figurile 5.53.5.54.

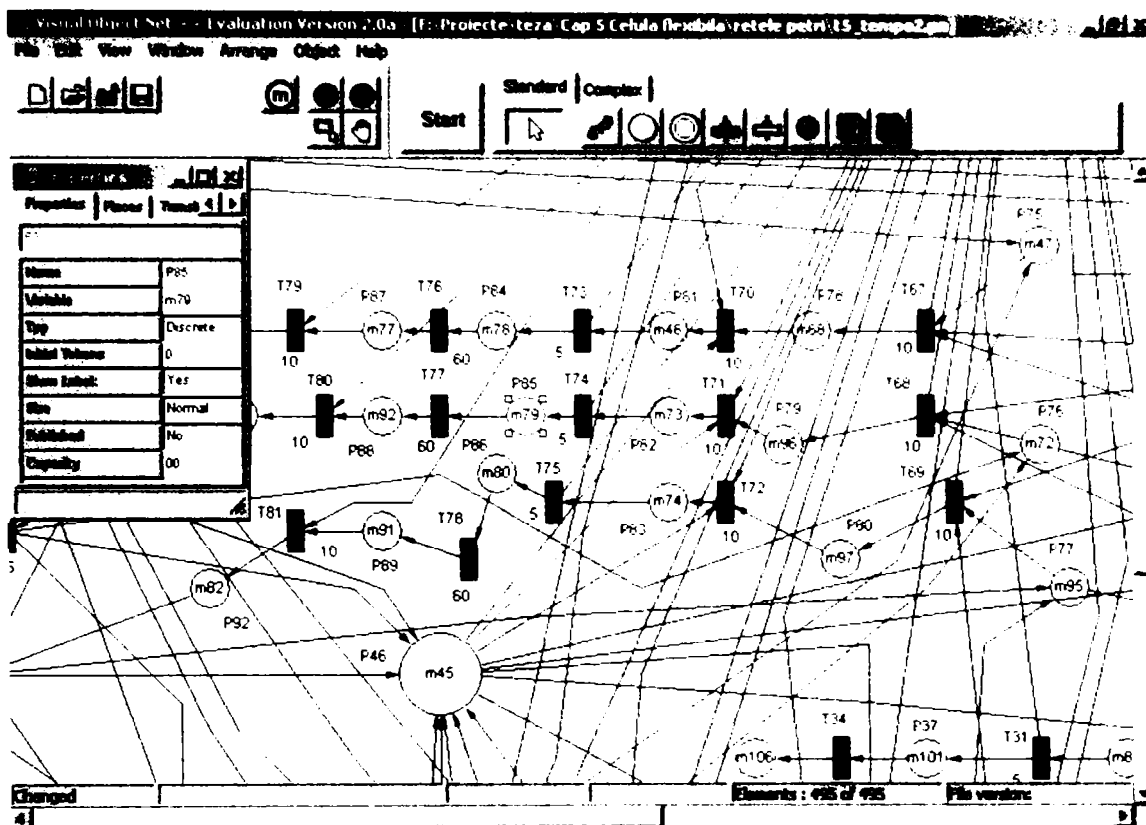


Fig.5.53. Interfața grafică a programului VisualObjectsNet 1.44.

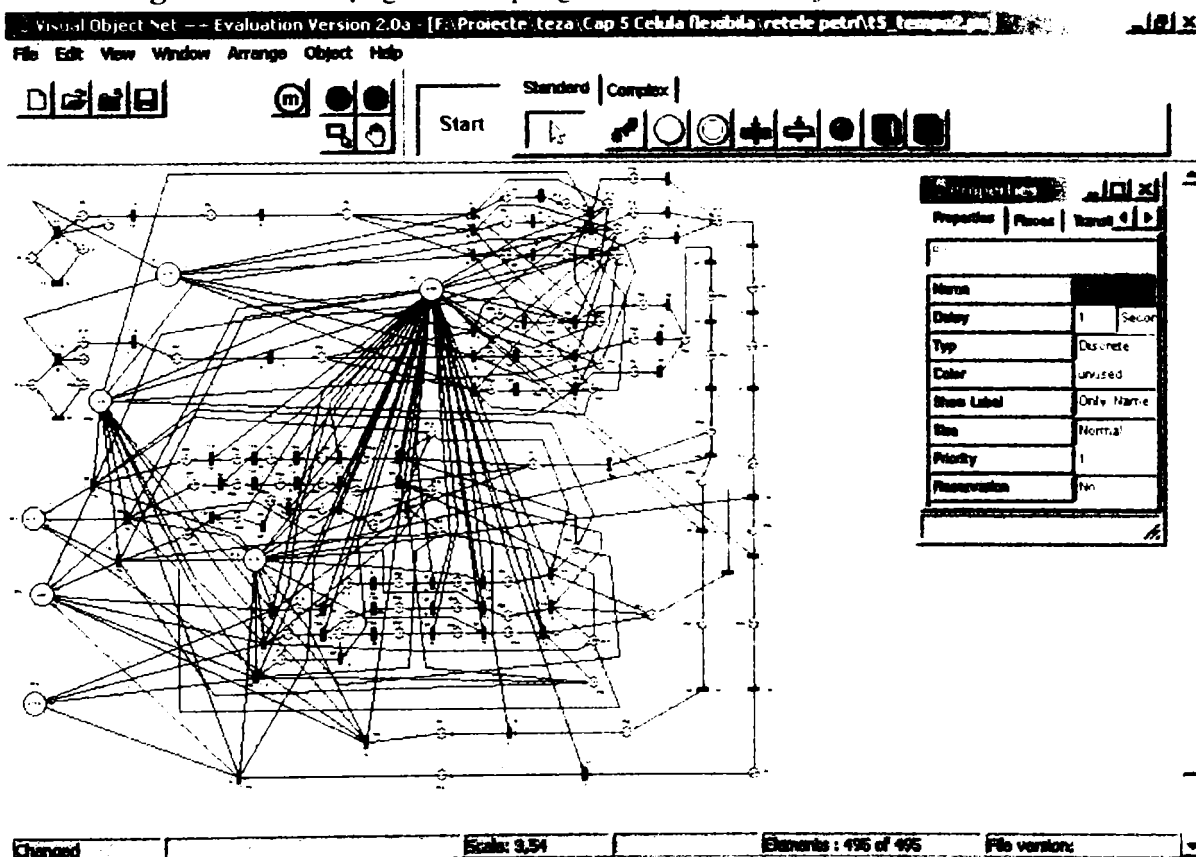


Fig.5.54. Schema rețelei Petri a CFLMI.

5.8. Aspecte privind realizarea CFLM1.

Aspectele practice ale realizării celulei flexibile CFLM1 sunt prezentate în figurile 5.55 – 5.67.

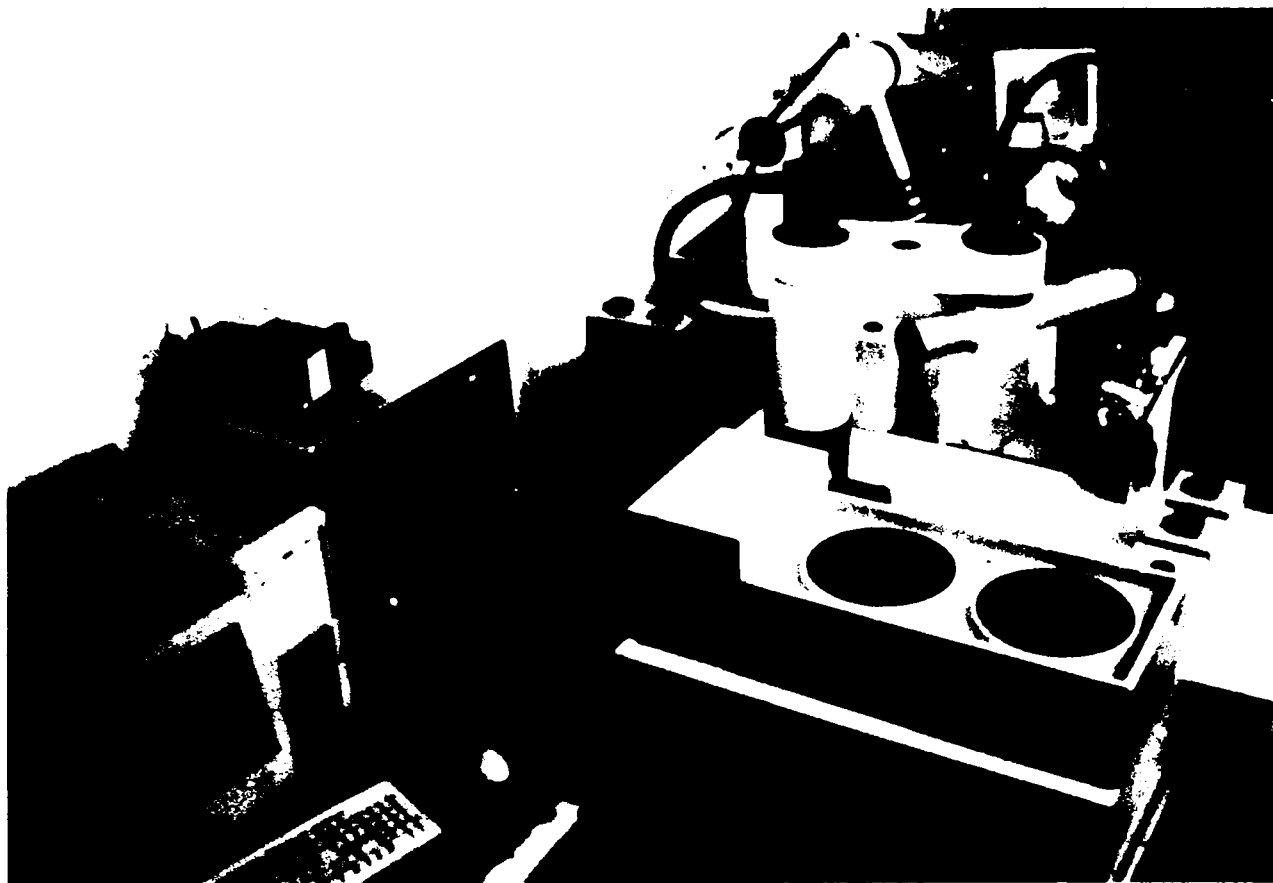


Fig.5.55. Vedere de ansamblu a CFLM1

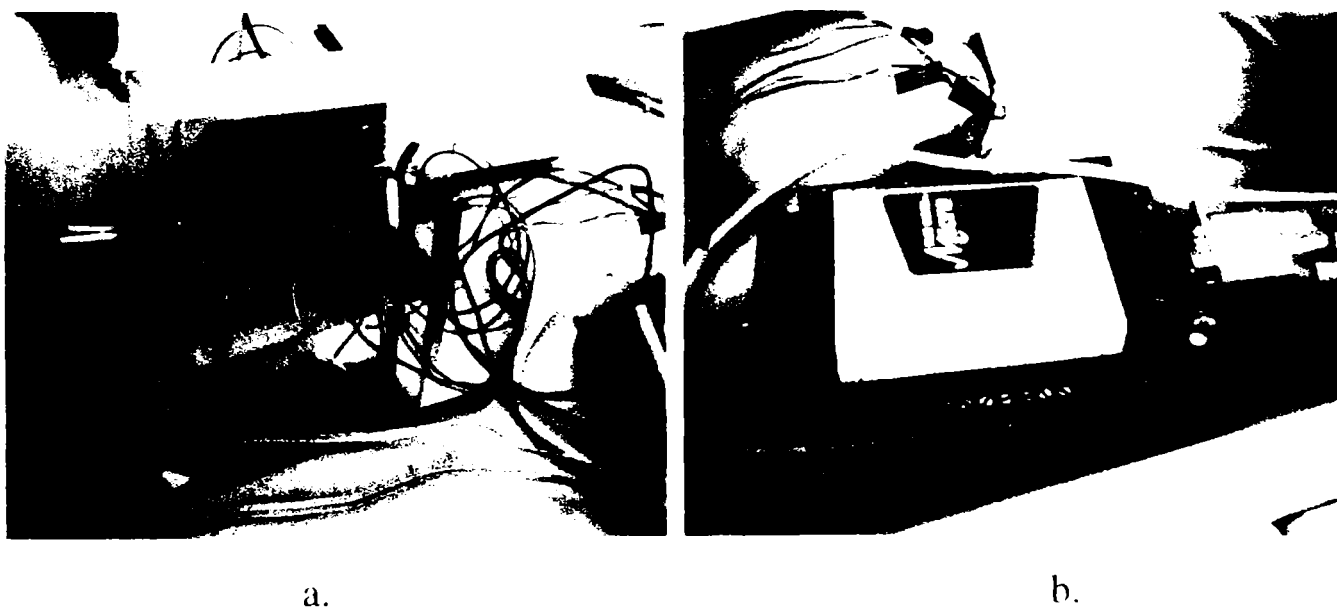
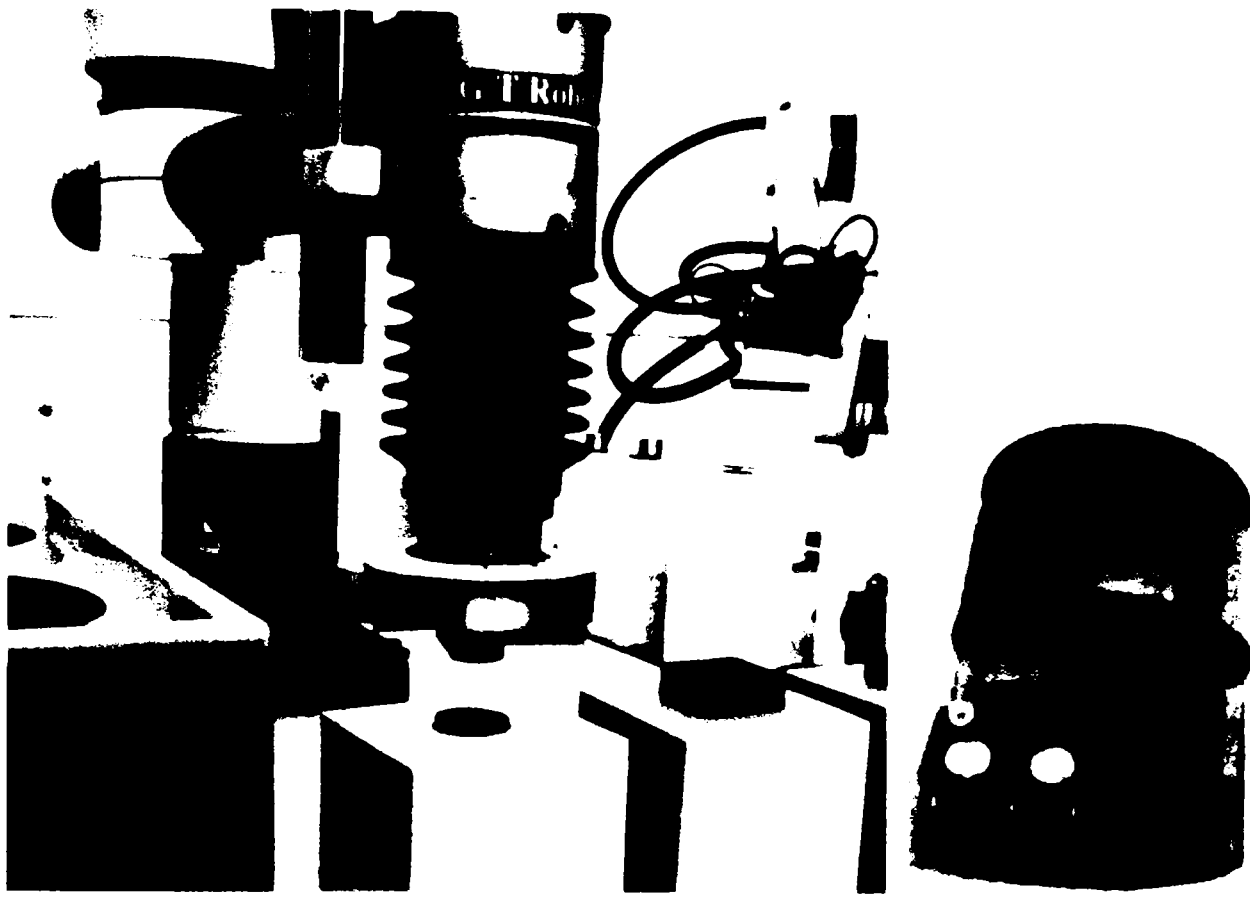


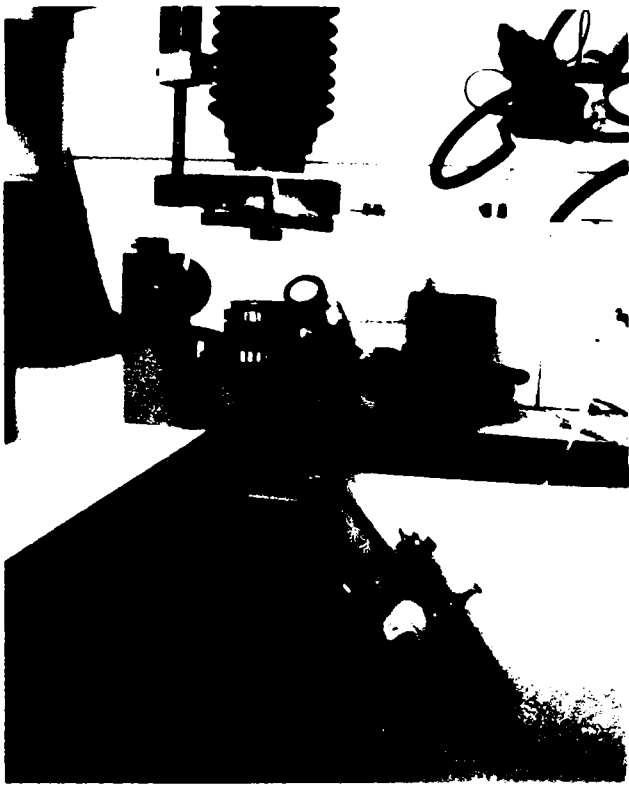
Fig.5.56. a. Interfețe motoare pas cu pas microscop M2;
b. Micro-calculatorul Blue-Earth 485.



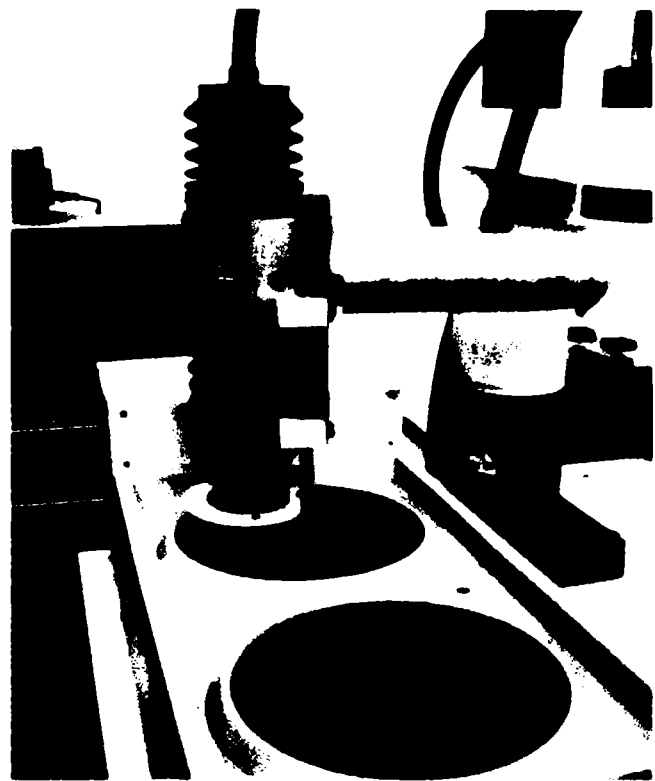
a.
Fig.5.57. a. Stocatorul de intrare St01;
 b. Blocul cu micro-intrerupător al stocatorului



a.
Fig.5.58. a. Robotul Ro1 asistă achiziția de imagine a rugozității suprafeței eșantionului;
 b. Imaginea achiziționată pe monitorul sistemului Sc1.

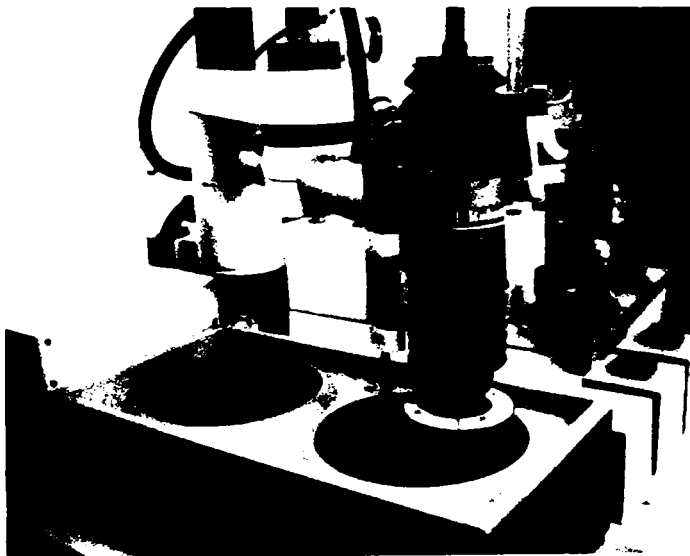


a.

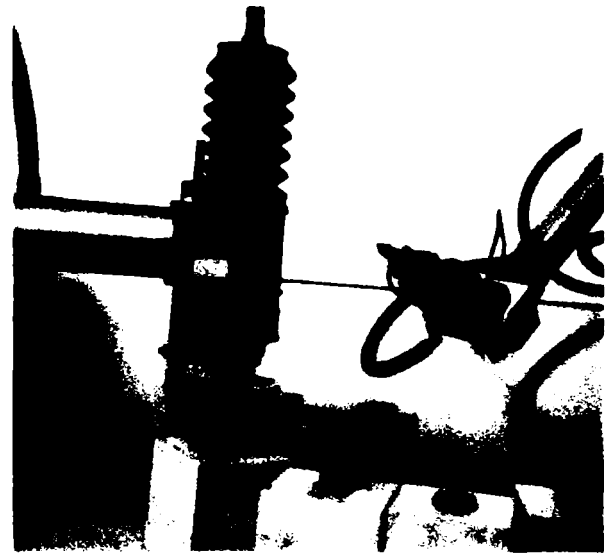


b.

Fig.5.59. a.Postul de analiză a rugozității; b. Accesul robotului RoI la postul de șlefuire La eșantioanelor



a.

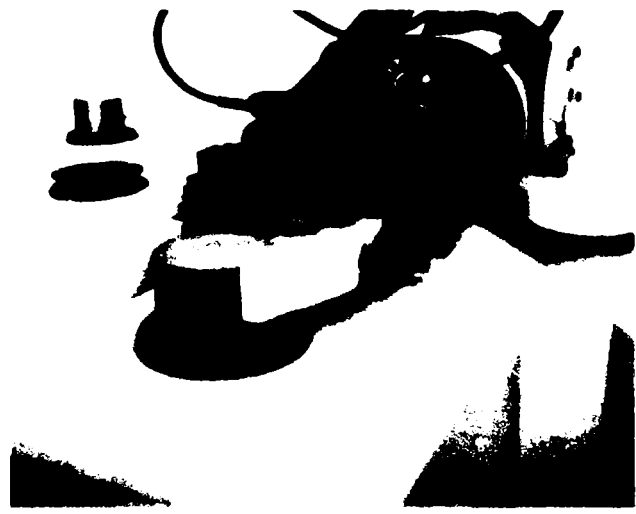


b.

Fig.5.60 a. Accesul robotului RoI la postul de șlefuire 2; b. Robotul RoI transferă eșantionul pe unul din stocatoarele de interfață (STI)

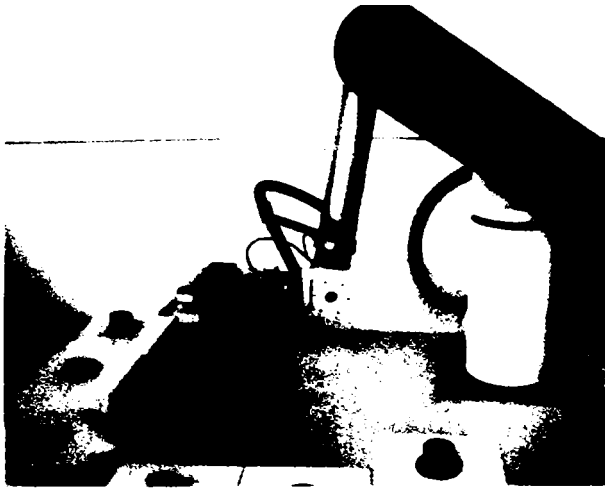


a.

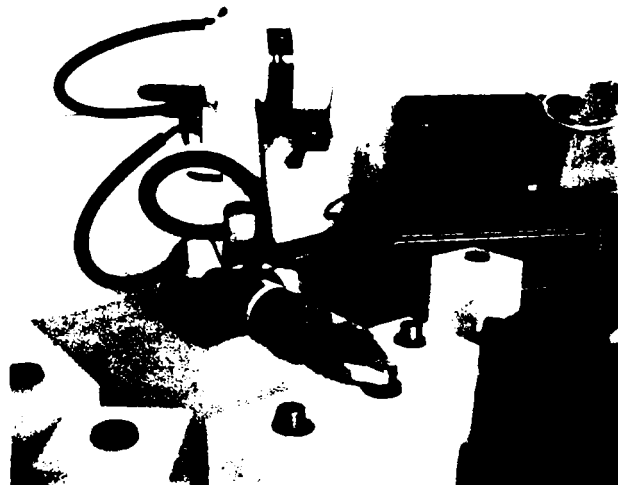


b.

Fig.5.61. a. și b. Robotul Ro2 preia eșantionul de pe stocatorul St11



a.



b.

Fig.5.62. a. Robotul Ro2 asistă postul de atac cu reactiv; b. Robotul Ro2 transferă eșantionul pe postul de stocare St22.

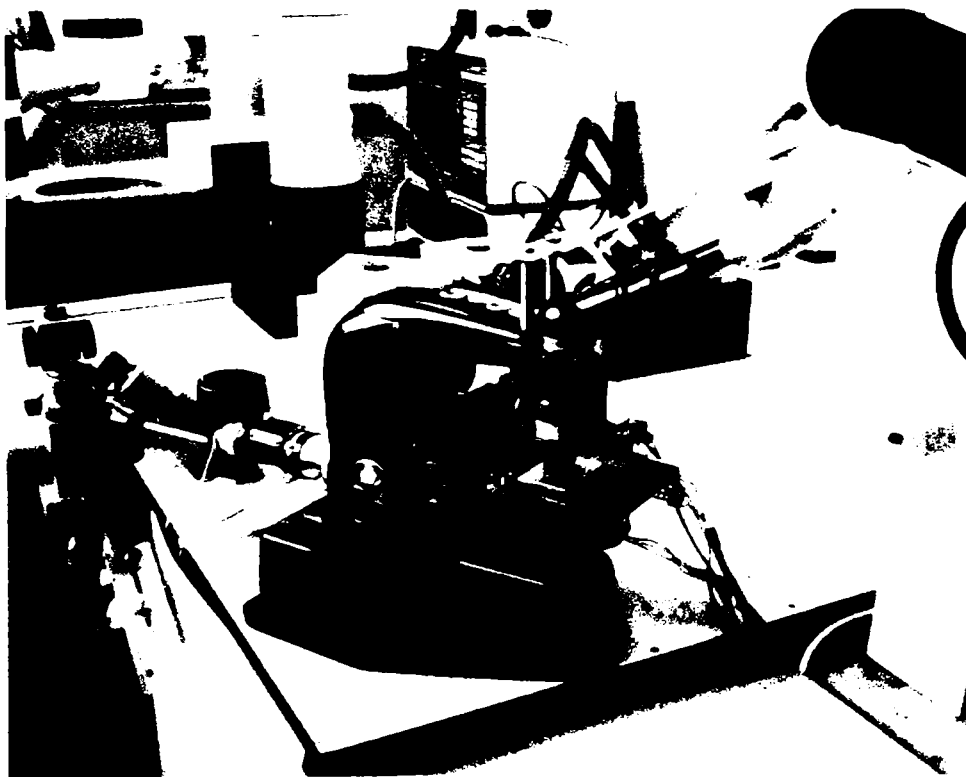
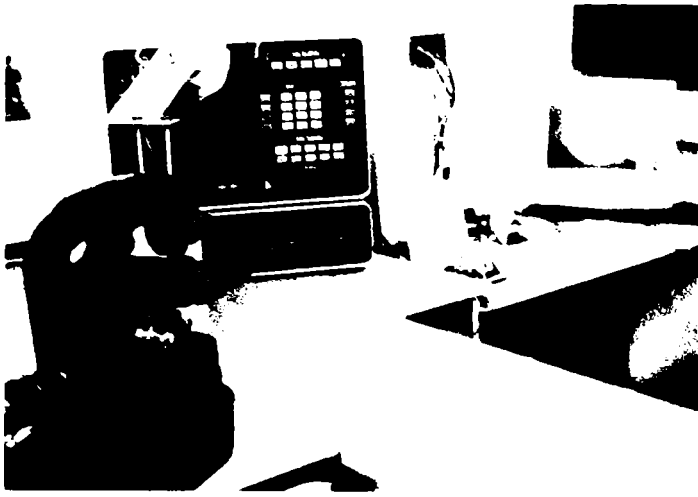


Fig.5.63. Postul de analiză metalografică.



a.

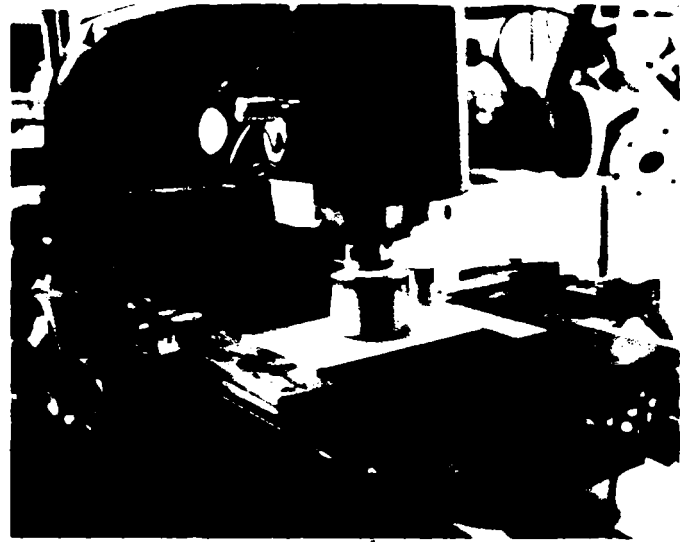


b.

Fig.5.64. a și b. Achiziția imaginii unei microstructuri

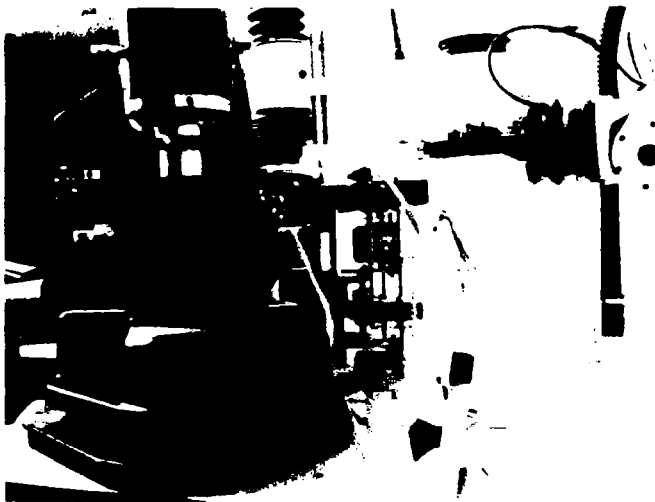


a.

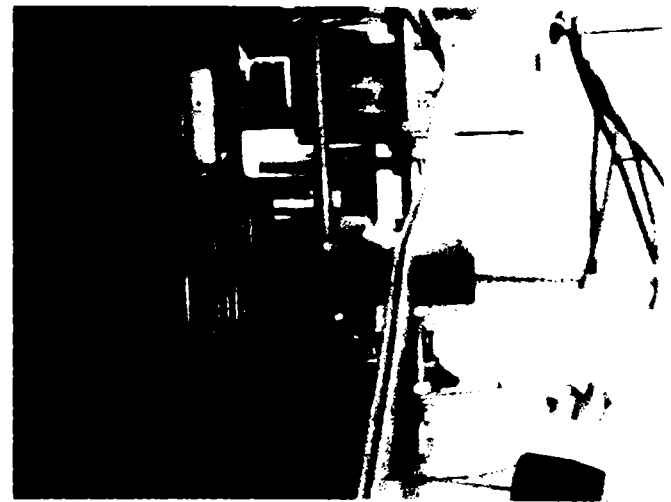


b.

Fig.5.65. a. Accesul robotului Ro2 la postul de analiza metalografică.
b. Eșantion în poziție de analiza



a.



b.

Fig.5.66. a. și b. Detalii ale acționării cu motoare pas cu pas
a mesei microscopului metalografic



Fig.5.67. Robotul Ro2 transferă piesa la stocatorul de reșire St31

6. Dezvoltarea structurii celulei flexibile, CFLM2

6.1. Necesitatea dezvoltării celulei flexibile

În cursul proiectării, realizării și experimentării celulei flexibile experimentale CFLM1 s-au constatat o serie de probleme și neajunsuri, nu atât în ceea ce privește structura fizică a celulei ci mai mult structura informațională a acesteia.

Pe de o parte ambii roboți sunt comandați în cadrul aceluiași program de aplicație iar pe de altă parte structura fizică a celulei favorizează funcționarea independentă a roboților. Acest fapt dă naștere unor contradicții ce trebuie soluționate în conformitate cu principiile programării enunțate în paragrafele anterioare.

6.2. Accesul roboților la posturile comune.

Corelarea acțiunilor roboților este necesară doar la interfața dintre cele două subsisteme, atunci când eșantionul trece din subsistemul 1 în subsistemul 2. Aici problema este de a evita coliziunea dintre roboți adică de a se interzice ca robotul Ro2 să inițieze descărcarea unuia dintre stocatoarele de interfață (St11,St12,St13), în timpul cât robotul Ro1 depune un eșantion pe unul din aceste stocatoare.

O soluție simplă este conectarea echipamentelor celor doi roboți pe porturile I/O și includerea în modulele program robot (MPR) robot Ro1 aferente specificatorilor "go", "load", "unload" pentru posturile St11,St12,St13, a setării unei ieșiri I/O (Bit de Testare a Evitării Coliziunii – BTEC), pe valoarea 1. Deasemenea se vor include în MPR corespunzătoare ale robotului Ro2 citirea intrării aferente BTEC și validarea execuției MPR numai dacă valoarea BTEC este 0. Această soluție are un grad de generalitate scăzut și nu corespunde cu cerințele de flexibilitate integrate deja în CFLM1. Modificările MPR ar duce la restrângerea posibilităților de utilizare a acestora (MPR în cauză vor fi valabile numai pentru stocatoarele St11,St12,St13).

Pentru rezolvarea situațiilor de acest gen, se consideră corectă aplicarea principiului unei conexiuni unice (realizate la nivelul programului de aplicație - figura 6.1.b.) în locul mai multor conexiuni pe diferite nivele ierarhice (figura 6.1.a.).

Gestionarea accesului mai multor roboți la aceleași posturi de lucru (sau stocatoare), prin conexiune de nivel înalt, păstrând totodată principiile programării structurate, orientate obiect, se poate realiza adăugând fiecărui obiect de tip PAR, PIR și STO două proprietăți :

- "<nume_obiect>.sharing" – definește dacă obiectul în cauză este sau nu accesabil de mai mulți roboți, dacă este accesabil valoarea proprietății va fi 1, dacă nu, valoarea este 0.
- "<nume_obiect>.share_status" – definește starea de accesare a obiectului și are valoarea 1 dacă obiectul este în curs de accesare și valoarea 0 dacă obiectul nu este accesat de un robot.

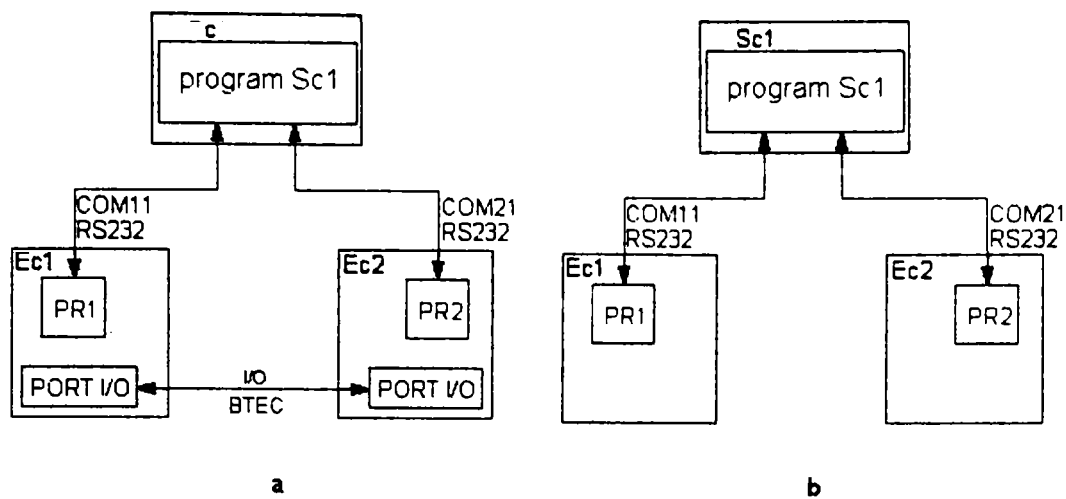


Fig.6.1. Schema partajării conexiunilor pe nivele ierarhice :
cu conexiuni pe mai multe nivel (a.); cu conexiune pe un singur nivel (b.)

Algoritmul de accesare a unui obiect (PAR,PIR sau STO) de către oricare din roboți este prezentat în figura 6.2.

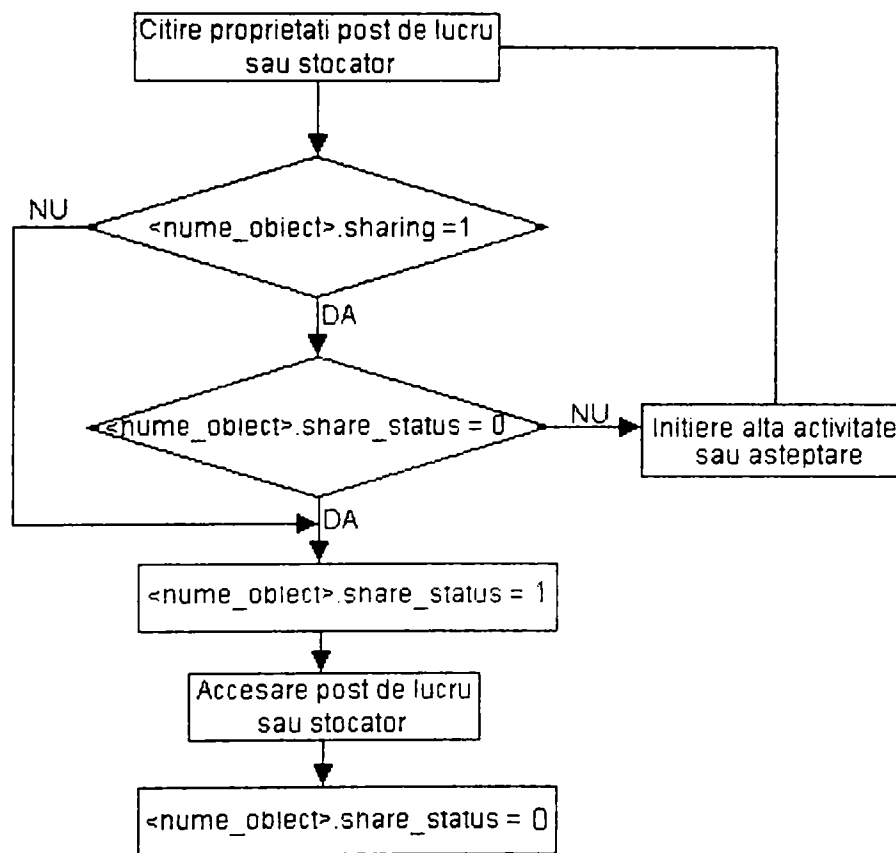


Fig.6.2. Algoritmul de accesare a posturilor de lucru și a stocatoarelor.

Se rezolvă astfel problema accesării posturilor utilizate în comun într-o manieră generalizabilă și în conformitate cu principiile de programare enunțate.

6.3. Execuția simultană a sarcinilor

În afara accesării posturilor comune, funcționarea celor două subsisteme este în principiu independentă, ceea ce are avantajul impunerii a mai puține restricții.

În cursul modelării celulei flexibile cât și în cursul experimentelor realizate s-au studiat posibilitățile de optimizare a încărcării posturilor de lucru și a roboților. Execuția sarcinilor din punctul de vedere al corelării lor în timp este studiată cu ajutorul ciclogramelor ce s-au trasat separat pentru cele două subsisteme (figura 6.3. și figura 6.4.).

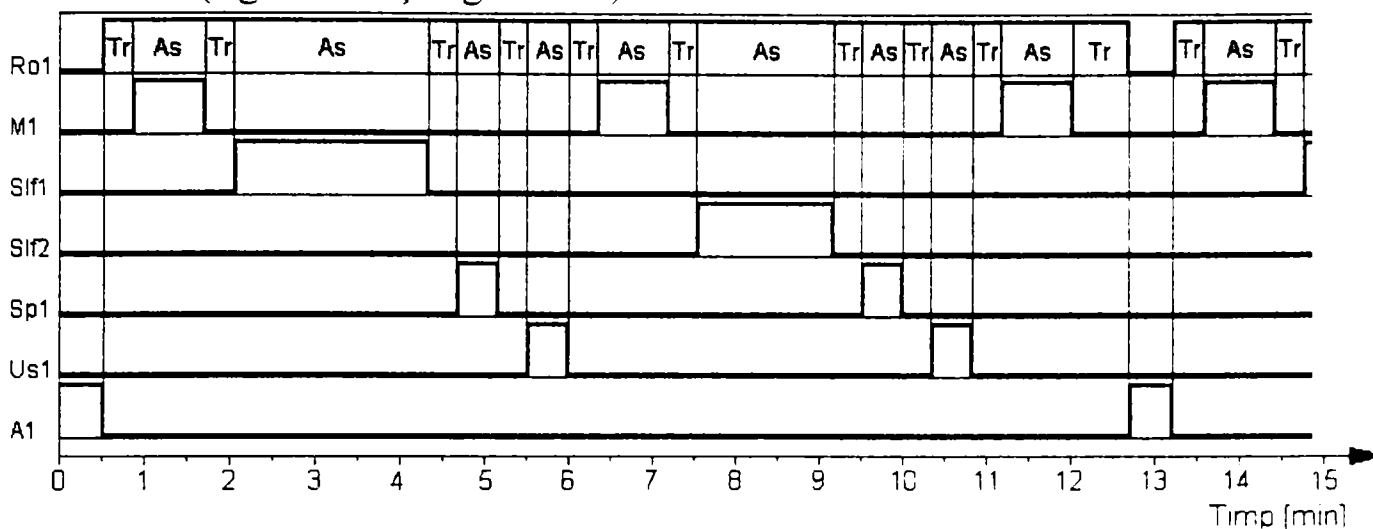


Fig.6.3. Ciclograma funcționării subsistemului 1.

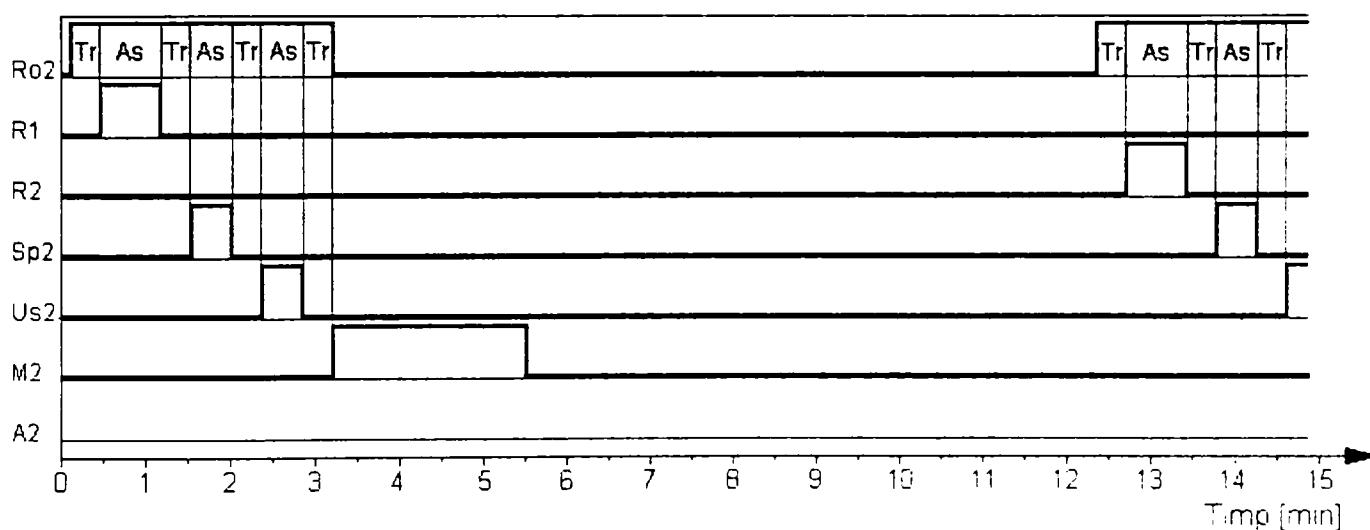


Fig.6.4. Ciclograma funcționării subsistemului 2.

Notățiile utilizate în figurile 6.3. și 6.4. sunt aceleași cu cele din figura 5.1., iar cu Tr se notează transferul și cu As se notează asistența dată de robot, postului de lucru.

Dependența dintre cele două subsisteme se manifestă prin faptul că subsistemul 2 trebuie să aștepte procesarea eșantionului în subsistemul 1 (figura 6.3. de la minutul 5,5 la minutul 12,5). perioada de staționare nereproductivă a subsistemului 2 necesită adoptarea unor soluții de mărire a productivității. Soluțiile ce se pot adopta sunt de două tipuri :

- utilizarea mai multor subsisteme de tipul 1, crescând astfel productivitatea în subsistemul de pregătire a eșantioanelor;
- utilizarea unor posturi de șlefuit autonome care să nu necesite asistența robotului Ro1;
- reducerea timpului de analiză a stării suprafeței și a numărului de analize.

Dintre aceste soluții reducerea timpului de analiză a stării suprafeței este tratată în capitolul 8.

Din dorința de a avea o flexibilitate ridicată o parte din modulele robot sunt definite pentru a realiza mișcări scurte (go, up, down, assist). Acest lucru duce la o solicitare crescută a sistemelor de calcul ce trebuie să realizeze atât lansarea în execuție a comenzilor roboților cât și să supravegheze ducerea la bun sfârșit a acestora. În cazul sistemului de calcul Sc1 acest fapt nu afectează în mod esențial funcționarea subsistemului deoarece toate posturile sunt posturi asistate de robot. În cazul sistemului de calcul Sc2 mișcările robotului Ro2 nu vor putea fi realizate simultan cu analiza metalografică sau cu gestionarea stocatoarelor decât prin utilizarea unei scheme de programare complicate care să implementeze un sistem de întreruperi la nivelul programului de aplicație celulă (figura 6.5).

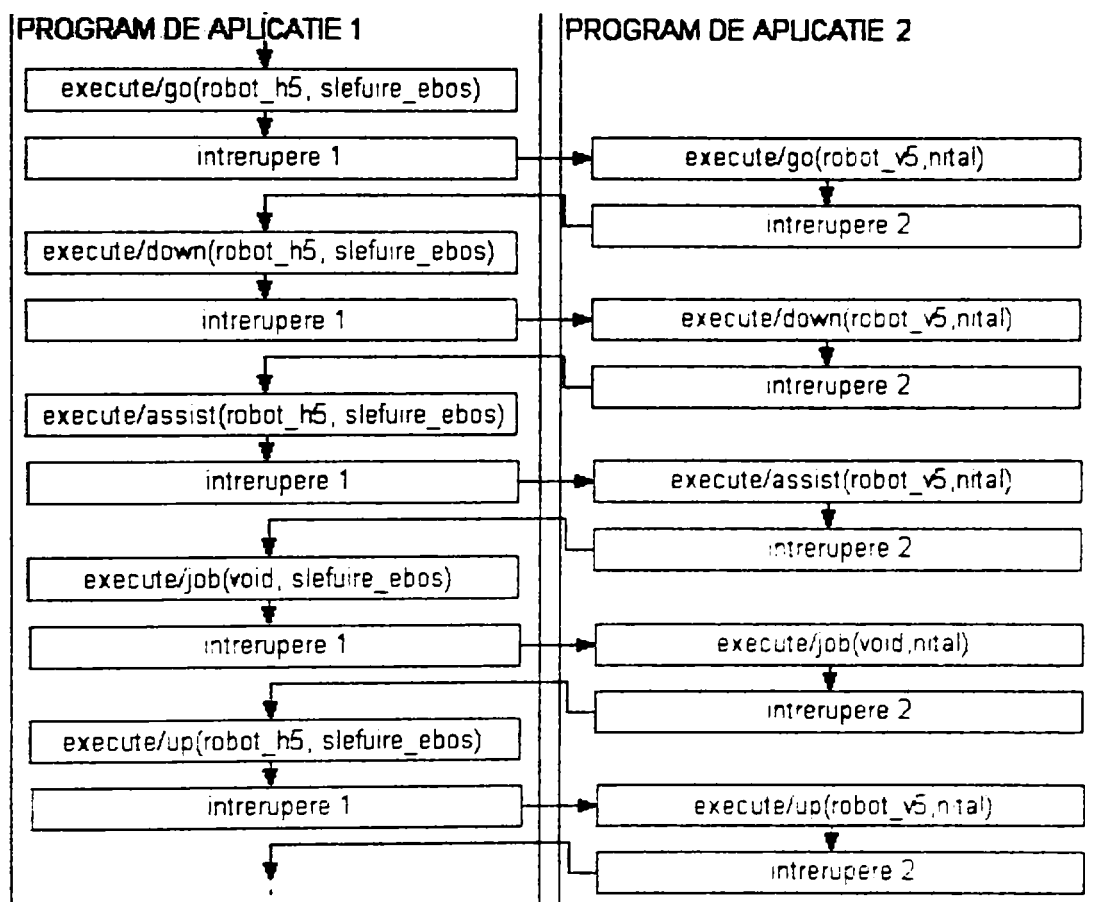


Fig.6.5. Schema rulării cu întreruperi a programului de aplicație celulă.

Pentru a se evita acest tip de funcționare singura soluție este transferul sarcinilor de comandă a postului de analiză metalografică și a gestiunii stocatoarelor pe încă două sisteme de calcul. Se realizează astfel o nouă variantă a celulei flexibile denumită CFLM2. Schema de ansamblu a celulei CFLM2 este arătată în figura 6.6 iar schema structurii informaționale este arătată în figura 6.7. Notățiile utilizate în figurile 6.6 și 6.7 sunt aceleași cu cele utilizate în cazul CFLM1. Din aceste figuri se observă că CFLM2 nu diferă de CFLM1 decât prin cele două sisteme de calcul Sc3 și Sc4 și prin instalarea la sistemul Sc2 a unui modul de achiziție de date (National Instruments PCI-6023E) ce realizează citirea semnalelor numerice de la stocatoare. Conceptele de programare și programele realizate pentru CFLM1 rămân valabile cu observația că acestea vor fi distribuite

de astă dată pe 4 sisteme de calcul în loc de două. Modificări mai importante vor avea loc doar la modulele de comunicare a informațiilor la nivelul rețelei LAN Windows2000, dar și aici metodele de transfer a datelor rămân cele specificate în cazul CFLM1.

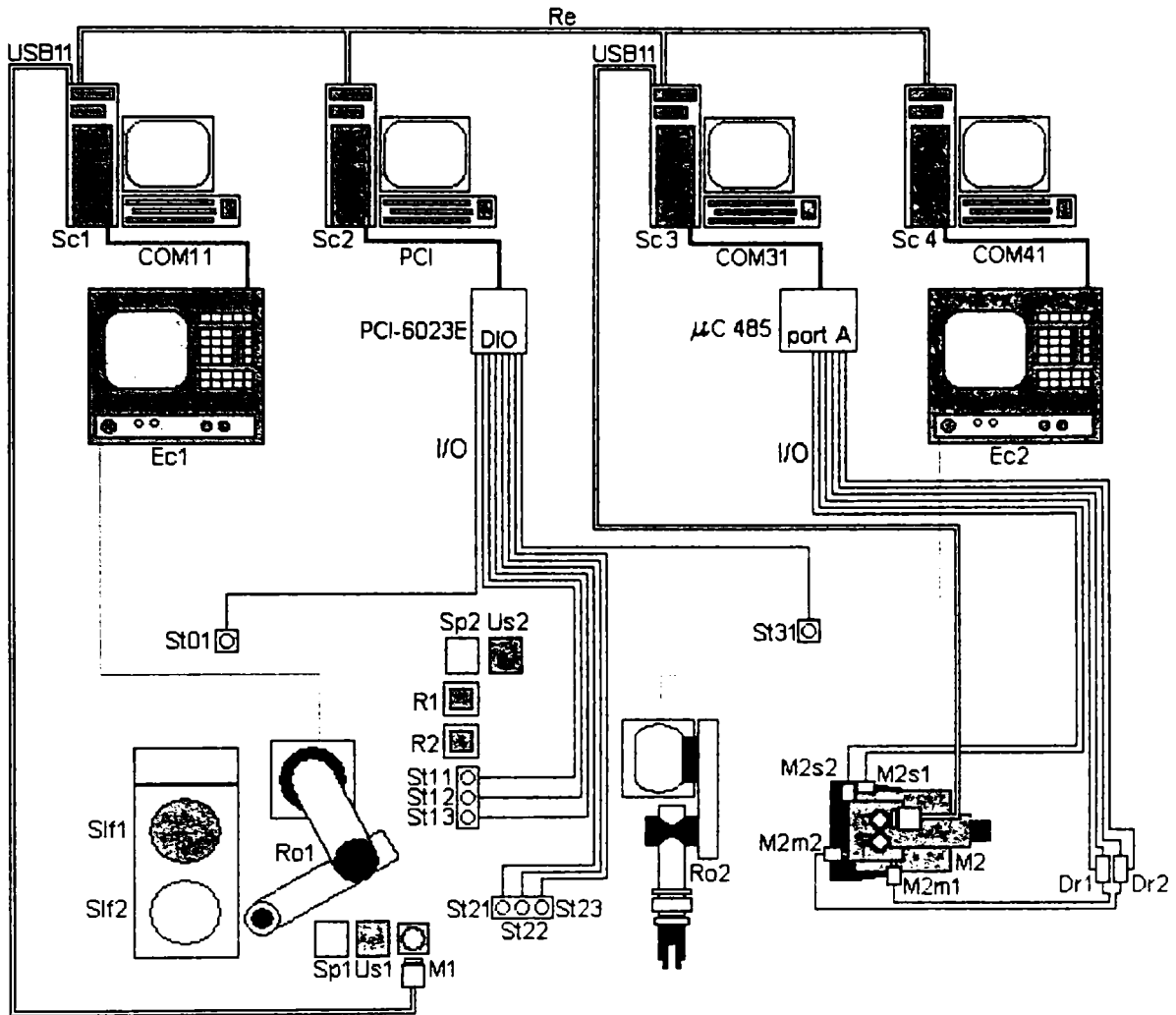


Fig.6.6. Schema de ansamblu a CFLM2.

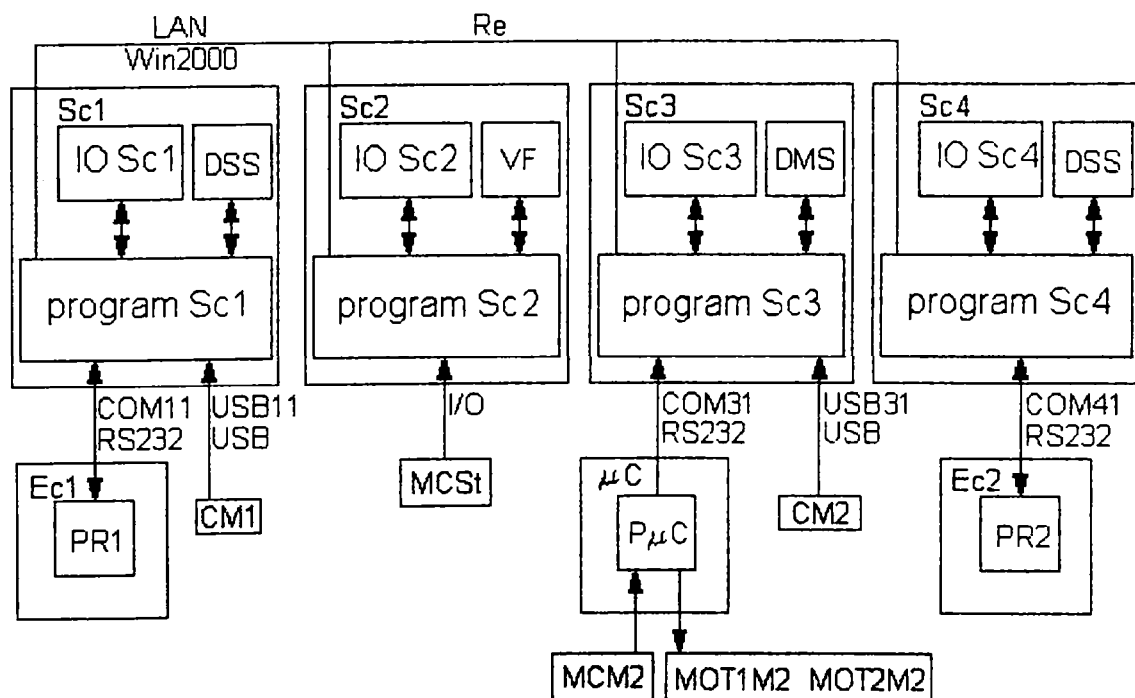


Fig.6.7. Schema structurii informaționale a CFLM2.

Din figura 6.7. se observă că sistemele Sc1 și Sc4 sunt cele care comandă cei doi roboți ai celulei. Pe aceste sisteme vor rula simultan programele de aplicație ce comandă celula flexibilă. Astfel se realizează distribuirea programului de aplicație (care în concepția clasică a celulelor flexibile rula pe un singur calculator), pe două calculatoare ce mențin în permanență un dialog. Această distribuire a programului de aplicație poate fi privită ca un prim pas spre paralelizarea algoritmului de comandă al celulei.

6.4. Transmiterea informațiilor între sistemele de calcul ale celulei CFLM2.

În figura 6.8 se prezintă schema de principiu a programelor de transmitere a informațiilor între cele patru sisteme de calcul.

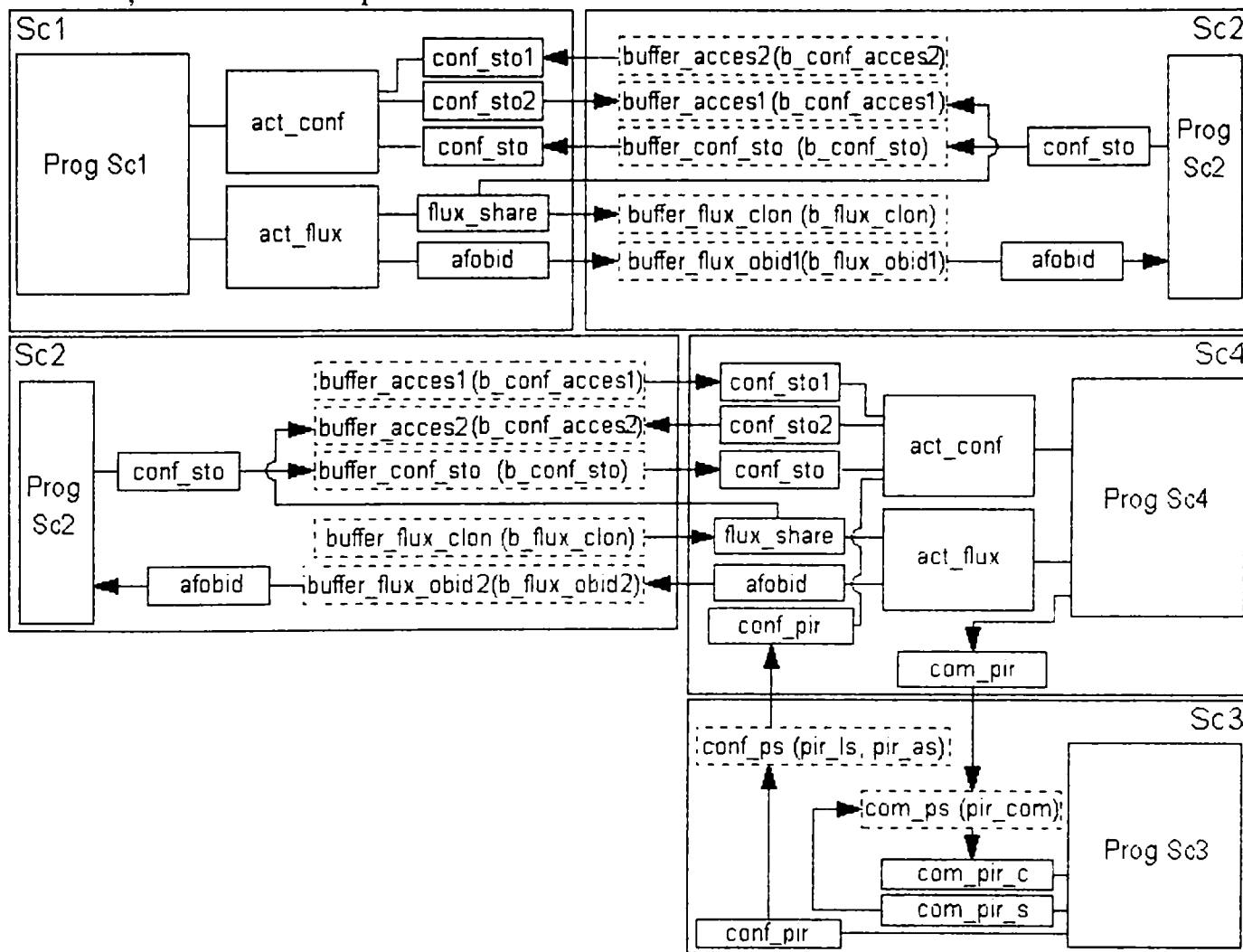


Fig.6.8. Schema transmiterii de informații între sistemele de calcul ale celulei flexibile CFLM2.

Pentru o prezentare mai ușoară a transmiterii informațiilor între sistemele de calcul, schema din figura 6.8. se descrie după rolul funcțional al programelor componente.

Sistemul Sc2 este utilizat pentru vizualizarea fluxului eșantioanelor, din care cauză sistemele Sc1 și Sc4 îi vor transmite informațiile referitoare la eșantioanele aflate în celulă. Este deci convenabil ca Sc2 să fie utilizat pentru stocarea tuturor

buferele de comunicație. Sistemul Sc3 comunică numai cu sistemul Sc4 din care cauză buferele referitoare la comunicația dintre aceste sisteme sunt stocate pe Sc3.

Informațiile transmise între sistemele de calcul ale celulei se pot sintetiza în următoarele categorii:

a. transmiterea informațiilor între sistemele Sc1, Sc2 și Sc4 :

a.1. transmiterea informațiilor referitoare la posturile comune;

a.2. transmiterea informațiilor referitoare la confirmarea prezenței eșantionului pe stocatoare;

a.3. transmiterea informațiilor referitoare la vizualizarea fluxului eșantioanelor.

b. transmiterea informațiilor între sistemele Sc4 și Sc3:

b.1. transmiterea informațiilor de comandă a postului de analiză metalografică;

b.2. confirmarea terminării procesării la postul de analiză metalografică.

Se descriu în continuare algoritmele de transmitere a informațiilor pentru categoriile enunțate.

a.1. Transmiterea informațiilor referitoare la posturile comune.

a.1.1. Accesarea posturilor comune.

Accesarea posturilor comune se realizează conform algoritmului prezentat în figura 6.2. Realizarea practică a algoritmului implică transmiterea informațiilor (a proprietății <nume_obiect>.share_status) între sistemele Sc1 și Sc4. Aceasta se realizează prin intermediul variabilelor „b_conf_acces1” stocat în buferul „buffer_acces1” și „b_conf_acces2” stocat în buferul „buffer_acces2”. La execuția modulului „act_conf” din cadrul sistemului Sc1, se execută instrucțiunile din schema logică (figura 6.2) până la „Accesare post de lucru sau stocator” iar setarea la 0 a proprietății „<nume_obiect>.share_status” se realizează în cadrul algoritmului „flux_share” apelat de către modulul „act_flux” la terminarea execuției procesării la postul comun.

a.1.2. Transferul eșantioanelor.

Urmărirea eșantioanelor în întreaga celulă flexibilă impune transmiterea informațiilor referitoare la eșantioane (în momentul trecerii acestuia din subsistemul 1 în subsistemul 2) între sistemele de calcul Sc1 și Sc4. Pentru realizarea acestui deziderat obiectele fizice comune St11, St12 și St13, sunt reprezentate în memoria fiecăruia din sistemele de calcul Sc1 și Sc4 prin obiectele virtuale sto(2), sto(3) și sto(4). Se poate spune că aceste obiectele virtuale comune existente în memoria sistemului Sc4 sunt clone ale obiectelor virtuale comune din memoria sistemului de calcul Sc1. Actualizarea proprietăților eșantioanelor la aceste posturi se realizează prin intermediul modulelor „act_flux” și respectiv „flux_share” și a variabilei „b_flux_clon” salvată în buferul „buffer_conf_sto”.

a.2. Transmiterea informațiilor referitoare la confirmarea prezenței eșantionului pe stocatoare.

Confirmarea prezenței eșantionului pe stocatoare se realizează la fel ca și în cazul descris pentru CFLM1 cu observația că pentru CFLM2 intrările digitale sunt realizate prin modulul de achiziții de date PCI 6023E. Citirea stării stocatoarelor de la sistemul Sc2 la Sc1 respectiv Sc4 se realizează prin modulul „conf_sto” și prin intermediul variabilei „b_conf_sto” din buferul „buffer_conf_sto”.

a.3. Transmiterea informațiilor referitoare la vizualizarea fluxului eșantioanelor.

Vizualizarea fluxului de eșantioane în celula flexibilă este realizată de către sistemul de calcul Sc2. Sistemele de calcul Sc1 și Sc4 transmit proprietățile eșantionului prezente la fiecare obiect virtual către sistemul de calcul Sc2 cu ajutorul modulelor „act_flux” și „afobid” prin intermediul buferului „buffer_flux_obid1”, variabila „b_flux_obid1” și respectiv a buferului „buffer_flux_obid2” și a variabilei „b_flux_obid2”.

b.1. Transmiterea informațiilor de comandă a postului de analiză metalografică.

Comanda de inițiere a analizei metalografice este realizată prin instrucțiunea execute(pir(1)) din cadrul programului de aplicație ce rulează pe sistemul de calcul Sc4. Această instrucțiune apelează modulul „com_pir” ce încarcă variabila „pir_com” din buferul „com_ps” cu codul corespunzător tipului de analiză necesar. Sistemul Sc3 citește valoarea înscrisă în variabila „pir_com” prin intermediul programului „com_pir_c” și anulează conținutul acesteia după citire prin intermediul modulului „com_pir_s”.

b.2. Confirmarea terminării procesării la postul de analiză metalografică.

Confirmarea terminării procesării eșantionului la postul de analiză metalografică se realizează de către sistemul de calcul Sc3 cu ajutorul modulului „conf_pir” prin intermediul buferului „conf_ps” și a variabilelor „pir_ls” și „pir_as” ce reprezintă proprietățile „load_status” respectiv „active_status” ale postului de analiză metalografică, descrise în cazul CFLM1.

7. Șlefuirea eșantioanelor metalografice

7.1. Considerații generale

În timpul șlefuirii, datorită forțelor de așchiere, dispozitivul de prehensiune al robotului este supus unor solicitări în planul discului abraziv. De asemenea eșantionul trebuie să exercite o forță de apăsare asupra discului de șlefuire. În timpul procesului forțele ce intervin nu sunt constante datorită fenomenelor de acumulare a șpanului și a imperfecțiunilor geometrice atât a eșantionului cât și a discului de șlefuire. Apar astfel două probleme :

- dispozitivul de prehensiune trebuie să permită mișcări elastice de mică amplitudine în planul discului de șlefuire – soluția constructivă propusă fiind descrisă în paragraful 7.2.;
- realizarea unei forțe de apăsare cvasi-constantă prin utilizarea unor regulatoare neuronale și fuzzy - paragraful 7.3.

7.2. Determinarea matricii complianței structurale a unui dispozitiv de prehensiune a eșantioanelor metalografice

Pentru determinarea complianței structurale a dispozitivului de prehensiune elementele cinematice din componența acestuia (lamele elastice, figura 7.1) vor fi considerate (simplificat) ca reprezentând fiecare o grindă cu două noduri, unul încastrat, celălalt liber.

Urmând metoda eforturilor unitare, matricea complianței unui element cinematic, exprimată în sistemul atașat elementului cinematic, este:

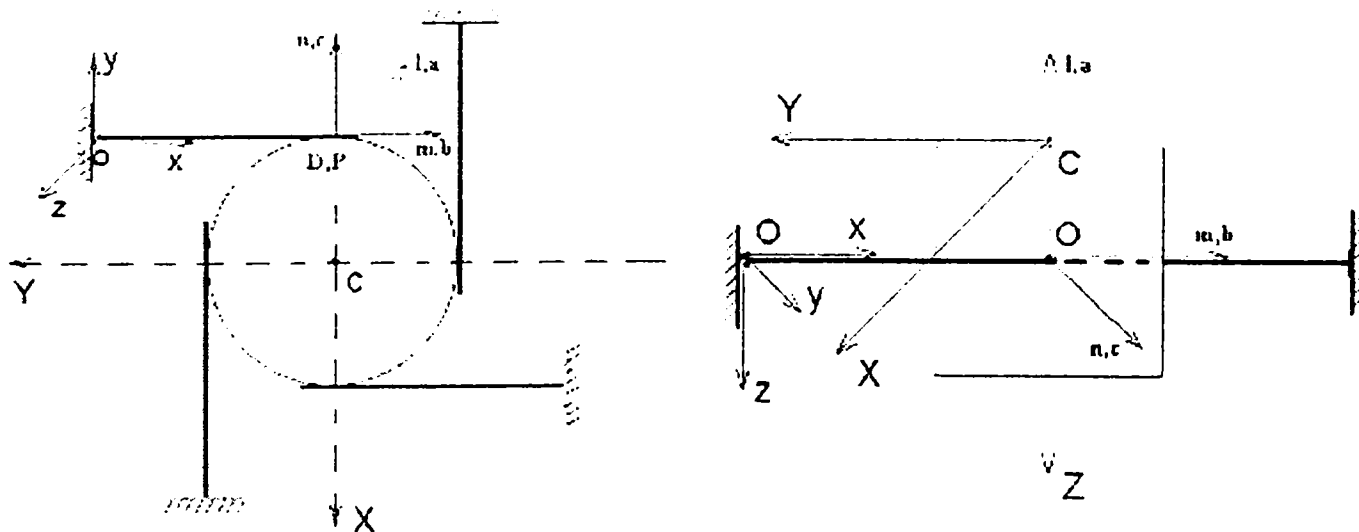


Fig.7.1. Schema principală a dispozitivului de prehensiune compliant propus.

$${}^o[C_1] = \begin{bmatrix} \frac{1}{EA} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{l^3}{3EI_z} & 0 & 0 & 0 & \frac{l^2}{3EI_z} \\ 0 & 0 & \frac{l^3}{3EI_y} & 0 & -\frac{l^2}{3EI_y} & 0 \\ 0 & 0 & 0 & \frac{1}{GI_p} & 0 & 0 \\ 0 & 0 & -\frac{l^2}{3EI_y} & 0 & \frac{1}{3EI_y} & 0 \\ 0 & \frac{l^2}{3EI_z} & 0 & 0 & 0 & \frac{1}{3EI_z} \end{bmatrix}; \quad (7.1)$$

unde: E -modulul de elasticitate longitudinal, G -modulul de elasticitate transversal, A -aria secțiunii elementului cinematic, I_z, I_y - momentele de inerție axiale în raport cu axele z și y , l -lungimea elementului cinematic.

$${}^o[C_1] = \begin{bmatrix} 0,6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 30,37 & 0 & 0 & 0 & 0,61 \\ 0 & 0 & 59,52 & 0 & -1,19 & 0 \\ 0 & 0 & 0 & 0,04 & 0 & 0 \\ 0 & 0 & -1,19 & 0 & 0,02 & 0 \\ 0 & 0,61 & 0 & 0 & 0 & 0,01 \end{bmatrix} \cdot 10^5 N/mm^2 (N \cdot mm^2)$$

Pentru exprimarea matricei complianței structurale a lamei în sistemul de referință $D(a, b, c)$, se va porni de la relația:

$${}^D[C_1] = {}^D[J_1] \cdot {}^o[C_1] \cdot {}^D[J_1]^T, \quad (7.2)$$

unde ${}^D[J_1]$ reprezintă matricea transformării între sistemele $O(x, y, z)$ și $D(a, b, c)$ definite prin ecuația :

$${}^D[J_1] = \begin{bmatrix} {}^o[R]^T & {}^o[R]^T \cdot {}^o[\hat{P}_D]^T \\ [O]_{3 \times 3} & {}^o[R]^T_{3 \times 3} \end{bmatrix} \quad (7.3)$$

unde ${}^o[\hat{P}_D]$ este matricea antisimetrică 3×3 :

$${}^O[\hat{P}_D] = \begin{bmatrix} 0 & {}^O r_{D_2} & {}^D r_{D_1} \\ {}^D r_{D_2} & 0 & {}^D r_{D_3} \\ {}^D r_{D_3} & {}^D r_{D_1} & 0 \end{bmatrix} \quad (7.4)$$

iar ${}^O[R]^T$ reprezintă transpusa matricei de rotație ce exprimă orientarea sistemului $D(a,b,c)$ față de $O(x,y,z)$

$${}^D_o[J_I] = \begin{bmatrix} {}^O_D[R]^T & {}^O_D[R]^T \cdot {}^O[\hat{P}_D]^T \\ [O]_{3 \times 3} & {}^O_D[R]^T_{3 \times 3} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 50 \\ 0 & 0 & 1 & 0 & 50 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

Matricea ${}^D[C_I]$, odată exprimată, se poate asambla în matricea ${}^{\mathcal{C}}[C]$ care reprezintă matricea complianței structurale a dispozitivului de prehensiune:

$${}^{\mathcal{C}}[C] = \begin{bmatrix} {}^D[C_I] & 0 & 0 & 0 \\ 0 & {}^D[C_I] & 0 & 0 \\ 0 & 0 & {}^D[C_I] & 0 \\ 0 & 0 & 0 & {}^D[C_I] \end{bmatrix} \quad (7.5)$$

Dacă matricea ${}^D[C_I]$ este inversabilă (în general acest fapt este posibil pentru că ${}^D[C_I]$ nu este neglijabilă), se poate obține matricea rigidității structurale a dispozitivului de prehensiune ${}^{\mathcal{K}}[K]$.

Pentru a exprima matricea rigidității structurale a dispozitivului de prehensiune în sistemele $P(l,m,n)$ aferente punctelor de contact de pe eșantion, se va lua în considerare matricea ${}^{\mathcal{H}}[H]$ a constrângerilor de contact. Se definește matricea:

$$[H] = \begin{bmatrix} 001000 \\ 000100 \end{bmatrix}$$

pentru contactul liniar lamelă – eșantion. Pentru toate punctele de contact lamelă-eșantion matricea constrângerilor de contact devine:

$$[\mathcal{H}] = \begin{bmatrix} [H] & 0 & 0 & 0 \\ 0 & [H] & 0 & 0 \\ 0 & 0 & [H] & 0 \\ 0 & 0 & 0 & [H] \end{bmatrix} \quad (7.6)$$

Matricea $([\mathcal{H}]^T [\mathcal{E}] \cdot [\mathcal{H}]^T)^{-1}$ reprezintă componentele rigidității lamelelor transmise prin contact. Se poate extinde această rigiditate la toate punctele de contact de pe eșantion:

$${}^p[\kappa] = [\mathcal{H}]^T \cdot ([\mathcal{H}]^T [\mathcal{E}] \cdot [\mathcal{H}]^T)^{-1} \cdot [\mathcal{H}] \quad (7.7)$$

Se poate exprima vectorul coloană ${}^c\bar{F}$, reprezentând vectorul rezultat și vectorul momentului rezultat al sistemului de forțe aplicate de lamele asupra eșantionului, sub forma:

$${}^c\bar{F} = {}^c_j[\mathcal{J}]^T \cdot {}^c_j[\bar{F}] = {}^c_j[\mathcal{J}]^T \cdot {}^c_j[\kappa] \cdot [\delta \cdot \mathcal{X}] = ({}^c_j[\mathcal{J}]^T \cdot {}^c_j[\kappa] \cdot {}^c_j[\mathcal{J}]) \cdot \delta^c \bar{X} \quad (7.8)$$

În ecuația (7.8) matricea:

$${}^c[\mathcal{K}] = {}^c_j[\mathcal{J}]^T \cdot {}^c_j[\kappa] \cdot {}^c_j[\mathcal{J}] \text{ reprezintă matricea rigidității unei prinderi.}$$

Pentru prehensiunea cu 4 lamele, matricea ${}^c_j[\mathcal{J}]$ se exprimă prin ecuația:

$${}^c_j[\mathcal{J}] = \begin{bmatrix} p_1 \\ c \\ J_1 \end{bmatrix}$$

unde

$${}^{p_1}_c[J_1] = \begin{bmatrix} 0 & 0 & -1 & 10 & 0 & 0 \\ 1 & 0 & 0 & 0 & 10 & 10 \\ 0 & -1 & 0 & -10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}$$

$${}^{p_2}_c[J_2] = \begin{bmatrix} 0 & 0 & -1 & 0 & 10 & 0 \\ 1 & 0 & 0 & 0 & 10 & 0 \\ 0 & -1 & 0 & -10 & 0 & 10 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}$$

$${}^c_c [J_3] = \begin{bmatrix} 0 & 0 & -1 & 10 & 0 & 0 \\ 1 & 0 & 0 & 0 & 10 & -10 \\ 0 & -1 & 0 & -10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}$$

$${}^c_c [J_4] = \begin{bmatrix} 0 & 0 & -1 & 0 & -10 & 0 \\ 1 & 0 & 0 & 0 & 10 & 0 \\ 0 & -1 & 0 & -10 & 0 & -10 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}$$

S-au utilizat următoarele notații:

- $\delta^D \bar{X}$ - vectorul coloană 6×1 , ce exprimă variațiile infinitesimale în orientarea și poziția punctului de contact al lamelei în raport cu baza dispozitivului de prehensiune, exprimate în sistemul $D (a, b, c)$, $\delta^D \bar{X} = J(\bar{\theta}) \cdot \delta \bar{\theta}$;
- $\delta^P \bar{X}$ - vectorul coloană 6×1 , ce exprimă schimbările infinitesimale în poziția și orientarea punctului de contact de pe eșantion față de centrul dispozitivului de prehensiune, exprimat în sistemul $P (l, m, n)$;
- $\delta^P \bar{X}_r$ - mișcările transmise prin contact, $H \delta^P \bar{X} = H \delta^D \bar{X} = \delta \bar{X}_r$;
- $\delta^C \bar{X}$ - vectorul coloană 6×1 , ce exprimă schimbările infinitesimale în poziția și orientarea eșantion în raport cu sistemul $O(x, y, z)$ și exprimate în sistemul $C(x, y, z)$;
- $[\delta^C \mathcal{X}]$ - matricea coloană exprimând schimbările infinitesimale în poziție și orientare, extinsă asupra tuturor lamelor, $[\delta^C \mathcal{X}] = [\delta^D \bar{X}_1, \delta^D \bar{X}_2, \dots, \delta^D \bar{X}_m]^T$;
- ${}^c \bar{F}$ - vectorul coloană 6×1 , reprezentând forța și momentul rezultat al sistemului de forțe, aplicate de lamele asupra eșantionului și exprimat în sistemul $C(x, y, z)$;

Dispozitivul de prehensiune propus a fost realizat practic fiind montat pe brațul robotului Ro1 din cadrul celulei flexibile CFLM. Fotografia dispozitivului de prehensiune proiectat și realizat este prezentată în capitolul 5.8 figura 5.58.

7.3. Utilizarea regulatorului neuronal cu linearizarea recției în comanda sistemului automat de șlefuit eșantioane metalografice.

7.3.1. Introducere.

În domeniul conducerii roboților există cercetări ce vizează aplicarea metodelor fuzzy și neuronale în rezolvarea problemelor sistemului de conducere care din diferite cauze sunt dificil de rezolvat prin metode analitice. În mai multe lucrări, cum ar fi [PHAM1], [LAU1] sau [ZHANG1] și [JANG1], sunt prezentate soluții ale unor probleme de comandă având la bază metode combinate de logică fuzzy și rețele neuronale, numite metode neuro-fuzzy.

Domeniile ce intră cel mai des în atenția specialiștilor sunt :

- soluții privind rezolvarea problemelor de cinematică inversă ;
- realizarea unor regulatoare neuronale locale sau globale prin :
 - reglare cu model predictiv;
 - reglarea prin liniarizarea reacției;
 - reglarea cu model de referință.

Structuri de conducere neuronale.

Abilitatea rețelelor neuronale de a modela sisteme neliniare, este folosită, în diverse cercetări [WENMEI1], [ZHANG1], în vederea realizării unor sisteme de reglare automată a unor procese industriale.

În cazul utilizării rețelelor neuronale în conducerea unor procese, există două faze importante: identificarea sistemului (realizarea modelului neuronal al procesului); proiectarea comenzii (realizarea sistemului de reglare neuronal);

În faza de identificare a sistemului, se proiectează un model neuronal al procesului condus iar în faza de proiectare a comenzii se folosește modelul neuronal realizat anterior pentru a antrena rețeaua ce va regla procesul.

Antrenarea modelului neuronal al procesului se realizează “offline” (sau “batch processing”), dar pentru optimizarea regulatorului neuronal sunt totuși necesare calcule și în timpul procesului.

Comanda neuronală cu model predictiv

Acest tip de comandă folosește un model al procesului pentru a observa modul în care va răspunde procesul la potențiale semnale de comandă. Se folosește apoi un regulator ce calculează mărimile de comandă optime pentru un anumit orizont de timp.

1. Identificarea.

În figura 7.2. se prezintă schema structurală a procesului de identificare a sistemului, în care erorile de predicție, formate ca diferențe între ieșirile generate de proces și cele generate de modelul neuronal, sunt folosite ca semnale de antrenare a rețelei, iar în figura 7.3 este prezentată rețeaua neuronală, ce modelează procesul, și folosește intrările și ieșirile anterioare pentru a genera semnale de comandă.

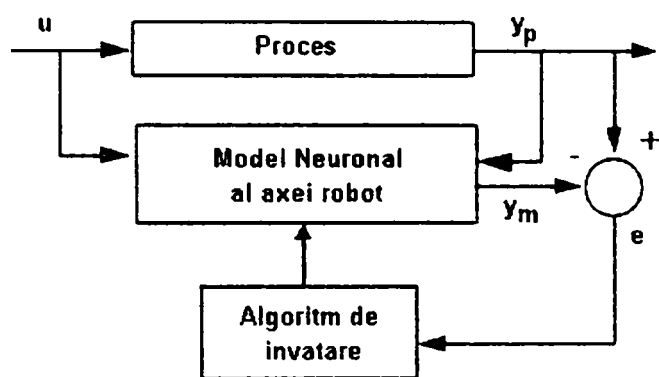


Fig.7.2. Schema de identificare a modelului neuronal al procesului.

Notățiile din figura 7.2 sunt : u – mărime de comandă; y_p – mărime de ieșire proces; y_m – mărime de ieșire model; $e = y_p - y_m$ – eroarea de ieșire.

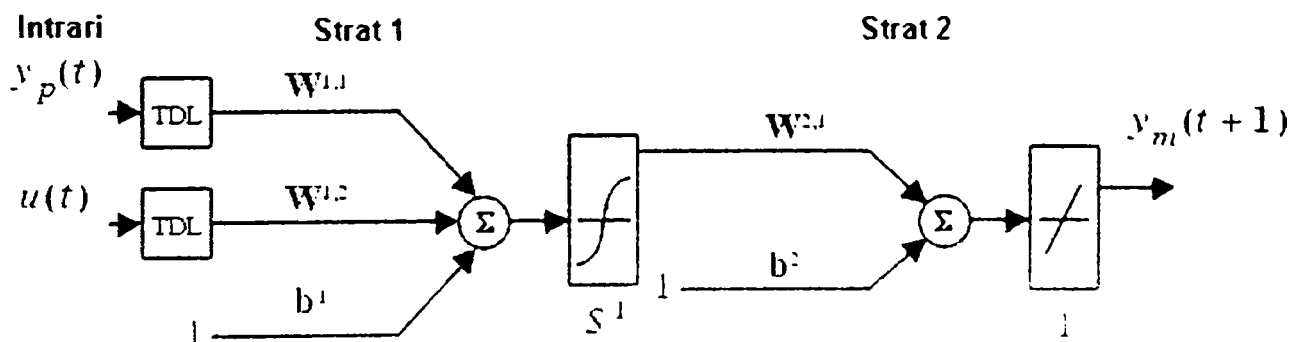


Fig.7.3 Structura modelului neuronal al procesului.

Notățiile din figura 7.3 reprezintă : $u(t)$ – mărime de comandă la momentul t ; $y_p(t)$ – mărime de ieșire proces la momentul t ; $y_m(t+1)$ – mărime de ieșire model pentru momentul $t+1$ (estimata mărimii de comandă); $w^{i,j}$ – ponderea intrării j din stratul i ; b^k – ponderea intrării constante (bias) pentru stratul k ; TDL – întârziere (Time Delay); S^l – funcție de transfer sigmoidală pentru stratul 1 ; l – funcție de transfer liniară pe porțiuni.

2. Reglarea

Metoda reglării cu model predictiv se bazează pe ideea orizontului regresiv (Soloway și Haley, 1996). Modelul neuronal generează estimări ale răspunsului

procesului într-un anumit orizont de timp. Estimările sunt folosite într-un algoritm numeric de optimizare, pentru a determina acea valoare a mărimii de comandă care minimizează funcția criterială :

$$J = \sum_{j=N_1}^{N_2} (y_r(t+j) - y_m(t+j))^2 + \rho \sum_{j=1}^{N_u} (u'(t+j-1) - u'(t+j-2))^2 \quad (7.9)$$

în care :

- N_1 , N_2 și N_u definesc orizonturile de timp peste care sunt evaluate mărimile erorii și a pașilor (incrementelor) de comandă;
- u' - este semnalul de comandă de încercare;
- y_r - este răspunsul dorit;
- y_m - este răspunsul modelului neuronal;
- ρ - definește contribuția variației valorilor mărimilor de comandă încercate în timp anterior;

Schema structurală a sistemului este arătată în figura 7.4., în care se observă că regulatorul este compus din modelul neuronal al procesului și din blocul de optimizare. Blocul de optimizare determină care dintre valorile u' minimizează funcția criterială J . Odată găsită valoarea optimă u este folosită ca valoare de comandă în proces.

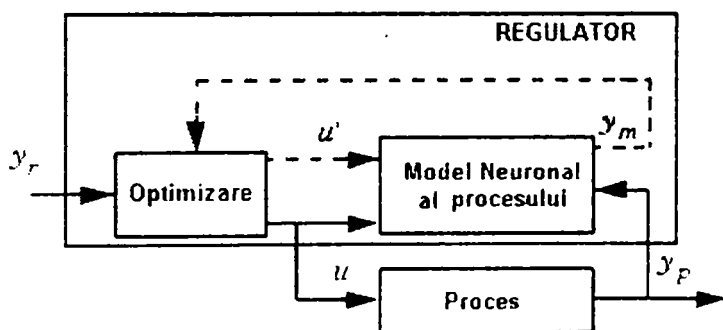


Fig.7.4. Schema structurală a procesului .

Datorită faptului că regulatorul execută calcule laborioase online (la fiecare pas trebuie evaluată funcția criterială de mai multe ori, pentru a putea alege valoarea optimă a mărimii de comandă), se poate afirma că aplicabilitatea în domeniul roboților a acestei metode este limitată. În literatura de specialitate studiată nu s-au întâlnit exemple de aplicare decât în procese mult mai lente cum ar fi de exemplu reglarea amestecului unor substanțe în reactoare chimice.

Comanda neuronală cu linearizarea reacției. (sau NARMA-L2).

1. Identificarea.

La fel ca și în cazul reglării cu model predictiv, mai întâi se va realiza identificarea sistemului ce urmează să fie reglat. Va trebui, deci, antrenată o rețea neuronală pentru a modela dinamica sistemului. Primul pas în procedura de

identificare este acela de a găsi o structură convenabilă pentru model. Un model standard pentru reprezentarea sistemelor neliniare discrete este modelul NARMA, (Nonlinear AutoRegressive-Moving Average), dat de relația :

$$y(k+d) = N[y(k), y(k-1), \dots, y(k-n+1), u(k), u(k-1), \dots, u(k-n+1)] \quad (7.10)$$

în care : $u(k)$ este intrarea sistemului și $y(k)$ este ieșirea. În scopul identificării se antrenează rețeaua pentru a aproxima funcția neliniară N . Procedura de antrenare este identică cu cea folosită în cazul comenzii cu model predictiv.

2. Reglarea

Dacă se definește ca scop urmărirea unei traiectorii de referință, $y(k+d)=y_r(k+d)$, va trebui dezvoltat un regulator neliniar de forma :

$$u(k) = G[y(k), y(k-1), \dots, y(k-n+1), y_r(k+d), u(k-1), \dots, u(k-m+1)] \quad (7.11)$$

Pentru a genera funcția G care minimizează eroarea pătratică medie, va trebui folosită, ca algoritm de învățare, metoda dinamică de propagare înapoi a erori, care este o metodă dificil de implementat și este foarte lentă (Hagan și De Jesus, 1999). Din această cauză, pentru a reprezenta sistemul, se folosesc modele aproximative (Narendra și Mukhopadhyay, 1997). Un astfel de model aproximativ, este dat de relația :

$$y(k+d) = f[y(k), y(k-1), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1)] + g[y(k), y(k-1), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1)] \cdot u(k) \quad (7.12)$$

Acest model este într-o formă, în care intrarea $u(k)$ nu este conținută în termenul neliniar, și dacă $y(k+d)=y_r(k+d)$, se poate scrie sub forma :

$$u(k) = \frac{y_r(k+d) - f[y(k), y(k-1), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1)]}{g[y(k), y(k-1), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1)]} \quad (7.13)$$

care însă ar necesita determinarea valorii intrării $u(k)$, bazată pe ieșire la același pas de calcul $y(k)$, din care cauză se folosește modelul de forma :

$$y(k+d) = f[y(k), y(k-1), \dots, y(k-n+1), u(k), u(k-1), \dots, u(k-n+1)] + g[y(k), \dots, y(k-n+1), u(k), \dots, u(k-n+1)] \cdot u(k+1) \quad (7.14)$$

în care $d \geq 2$.

Structura rețelei neuronale ce reprezintă modelul procesului, este dată în figura

7.5.

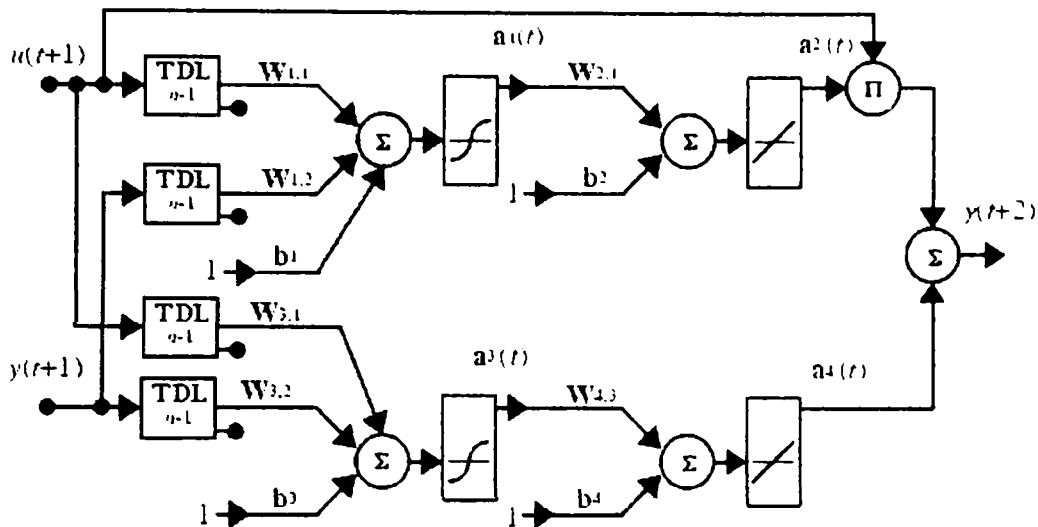


Fig.7.5. Structura modelului neuronal al procesului, construit pe baza relației (7.13)

Din relația 7.13, expresia structurii regulatorului va fi :

$$u(k+1) = \frac{y_r(k+d) - f[y(k), \dots, y(k-n+1), u(k), \dots, u(k-n+1)]}{g[y(k), \dots, y(k-n+1), u(k), \dots, u(k-n+1)]} \quad (7.15)$$

în care $d \geq 2$.

În figura 7.6 este prezentată schema de principiu a sistemului de reglare, în care mărimea y_r este generată de modelul neuronal (“model de referință”).

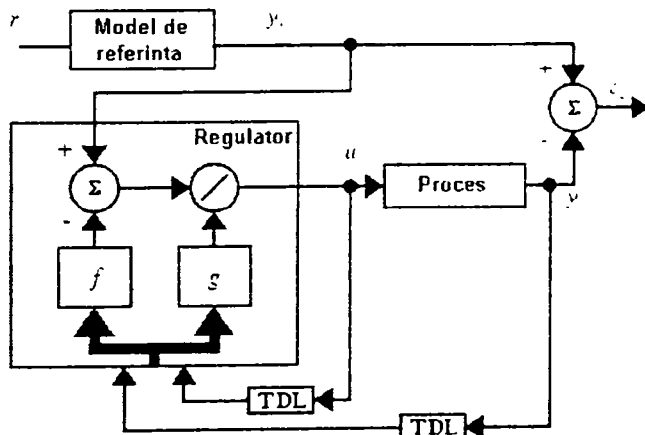


Fig.7.6. Schema sistemului de reglare, construit pe baza relației 7.15.

Schema structurală a rețelei neuronale ce reprezintă regulatorul prezentat în figura 7.6, este arătată în figura 7.7.

În cazul acestei metode, regulatorul execută un număr extrem de mic de calcule (calculul realizat poate fi urmărit pe diagrama din figura 7.7), iar modelul neuronal este antrenat offline. Din această cauză metoda poate fi folosită cu succes în cadrul comenzii roboților industriali. O aplicație directă poate fi realizarea unor regulatoare neuronale care să înlocuiască regulatoarele clasice pentru comanda unei axe de robot.

Avantajul folosirii unor regulatoare neuronale pentru roboți industriali, pe lângă avantajele date de o stabilire mai simplă a parametrilor, ar putea fi și mai mare dacă acestea ar fi folosite în sisteme în care și generatorul de traiectorii ar funcționa pe baze neuronale (sistem de calcul paralel).

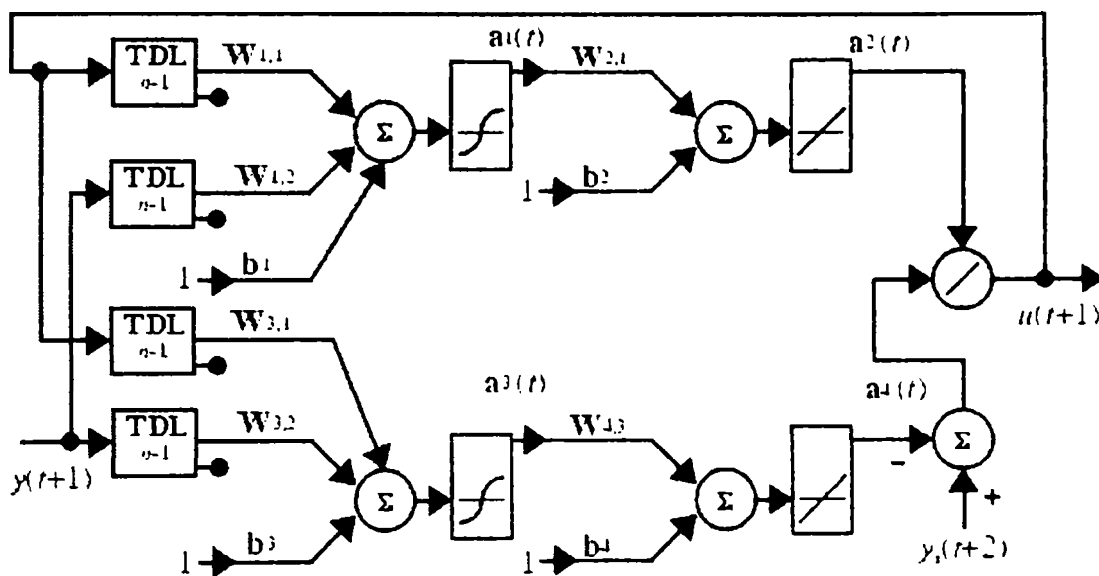


Fig.7.7. Structura rețelei neuronale reprezentând regulatorul, bazat pe relația (7.15)

Comanda neuronală cu model de referință.

1. Identificarea.

Pasul de identificare este realizat în mod similar cu cel din celelalte două metode prezentate.

2. Reglarea

După identificarea modelului, este antrenată rețeaua neuronală ce realizează reglarea procesului. Antrenarea rețelei necesită folosirea pentru învățare a metodei dinamice de propagare înapoi a erori, dificil de realizat și cu o durată lungă de realizare, însă cu un domeniu mai larg de aplicare decât metoda linearizării reacției (nu este necesară scrierea expresiei regulatorului în forma dată de relația 7.12). Rețelele neuronale, atât ale modelului cât și ale regulatorului sunt realizate pe două straturi. Numărul de întârzieri se selectează în funcție de ordinul sistemului, și anume cu cât sistemul este de ordin mai mare cu atât numărul de întârzieri va trebui să crească.

Și în acest caz (ca și la metoda cu linearizarea reacției), volumul de calcule online este mic. Ambele rețele, atât modelul cât și regulatorul sunt antrenate offline, aici însă volumul de calcule este mare.

Volumul mic de calcule necesare în timpul funcționării, face ca această metodă

să poată fi folosită în cadrul unor sisteme de conducere a roboților industriali, în mod similar cu metoda linearizării reacției.

7.3.2. Considerații privind modelarea sistemului de șlefuire.

Schema sistemului automat conceput pentru șlefuirea eșantioanelor metalografice este prezentată în figura 7.8.

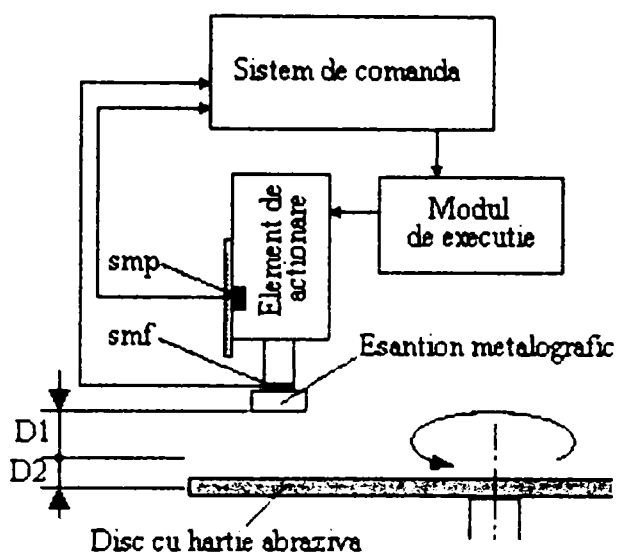


Fig.7.8. Schema sistemului automat de șlefuit eșantioane metalografice

Sistemul de comandă – reprezintă o unitate de calcul programabilă pe care rulează algoritmul de comandă ce înglobează regulatorul neuronal sau fuzzy (calculator de tip PC) și are posibilitatea de a transmite semnale către elementul de execuție, și de a primi semnale de la senzori (prin intermediul unei plăci de achiziție dotată cu un convertor A/N și un port de ieșiri digitale).

Modulul de execuție – conține elemente de execuție (ventile de cale și ventile de reglarea presiunii proporționale).

Smp, smf – reprezintă senzori de poziție respectiv de forță, necesari realizării buclelor de reacție.

Funcționarea sistemului.

Sistemul prezentat are ca scop realizarea apropierii eșantionului de abraziv, cu o viteză mare pe domeniul $D1$ și cu viteză mică pe domeniul $D2$. Domeniul $D2$ reprezintă o zonă de incertitudine în care nu se cunoaște exact locul în care va avea loc contactul dintre eșantion și abraziv (datorită poziției incerte a abrazivului cât și a dimensiunii variabile a eșantioanelor). După stabilirea contactului dintre eșantion și abraziv, regulatorul va trebui să mențină nivelul de presiune al acționării pneumatice în așa fel încât să realizeze o eroare cât mai mică față de forța de referință programată.

Pentru experimentarea reguletoarelor neuronale și fuzzy, forța dată de contactul dintre eșantion și abraziv se presupune a fi o forță elastică, și se înlocuiește printr-un resort.

Ca aplicație de modelare în ce privește reguletoarele neuronale s-a ales metoda reglării cu linearizarea reacției (NARMA) și reacție în poziție pentru rezolvarea problemei în domeniul în care s-a realizat contactul între eșantion și abraziv (se cunoaște poziția abrazivului). Ca suport al realizării modelului și a simulării s-a folosit modulul SIMULINK® al mediului de programare MATLAB®R12. SIMULINK, conține blocuri predefinite pentru generarea modelului neuronal cât și o interfață grafică ce permite setarea parametrilor, antrenarea, și validarea reguletorului neuronal.

În cele ce urmează este prezentată pentru comparație și modelarea variantei clasice de reglare on-off [CHEN].

Modelarea unei axe de translație pneumatice cu reglare on-off.

Pentru modelarea variantei clasice a axei pneumatice s-a ales sistemul prezentat în figura 7.9., în care : *smp* este elementul de măsură al poziției, *A1* și *A2* reprezintă ariile pistonului, *mpt* este motorul pneumatic liniar, *m* este masa inerțială, *el1* și *el2* reprezintă ventile de cale cu comandă electrică, *Ps* este presiunea de lucru iar *Pe* reprezintă presiunea atmosferică. Notățiile și valorile parametrilor sistemului sunt prezentați în tabelul A7.1.1. din ANEXA 7.1.

Ecuția dinamică a sistemului este dată de relația :

$$m\ddot{x} = P_1 A_1 - P_2 A_2 - b\dot{x} \tag{7.16}$$

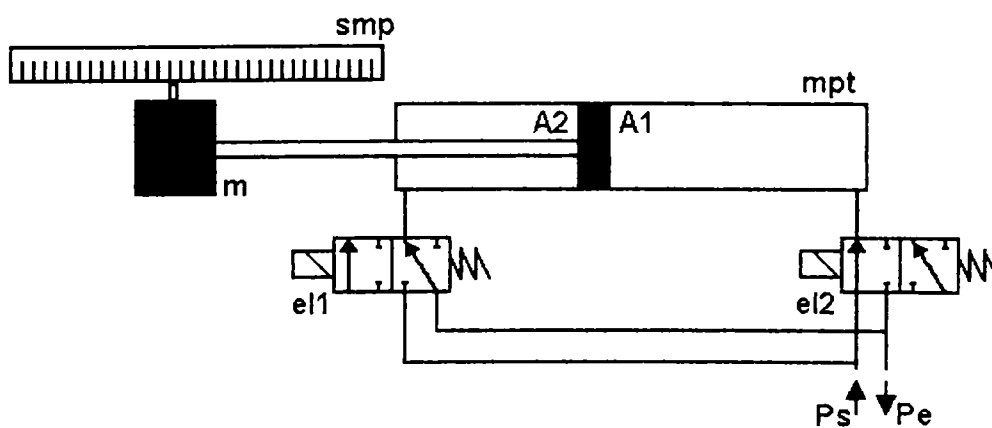


Fig.7.9. Schema sistemului pneumatic pentru comanda on-off.

Relațiile (7.17) și (7.18.) reprezintă expresia matematică a legii conservării energiei pentru debitul masic din camerele cilindrului pneumatic:

$$q_1 = A_1 \dot{x} \frac{P_1}{RT} + \frac{V_{1min} + A_1 x}{\gamma RT} \dot{P}_1 \quad (7.17.)$$

$$q_2 = -A_2 \dot{x} \frac{P_2}{RT} + \frac{V_{2min} + A_2(S-x)}{\gamma RT} \dot{P}_2 \quad (7.18)$$

în care primul termen din partea dreaptă reprezintă energia necesară deplasării masei inerțiale și al doilea termen din partea dreaptă reprezintă compresibilitatea aerului.

Relațiile (7.19) respectiv (7.20) reprezintă expresia legii lui Bernoulli :

$$q_1 = A_{pd1} \sqrt{2\rho(P_s - P_1)} \quad (7.19)$$

$$q_2 = -A_{pd2} \sqrt{2\rho(P_2 - P_e)} \quad (7.20)$$

în care A_{pdi} este aria orificiului ventilului de cale i .

Modelele SIMULINK realizate cât și rezultatele experimentale sunt arătate în :

- ANEXA 7.2 – modelul sistemului cu reglare “on-off”;
- ANEXA 7.3 – rezultatele simulării obținute la modelul cu reglare “on-off”.

7.3.3. Modelarea axei de translație pneumatice cu regulator neuronal.

Pentru sistemul cu reglare neuronală schema din figura 7.9, s-a modificat prin introducerea unui ventil regulator de presiune vr . S-a folosit un singur ventil regulator de presiune și s-au păstrat ventilele de cale. Schema astfel modificată este arătată în figura 7.10.

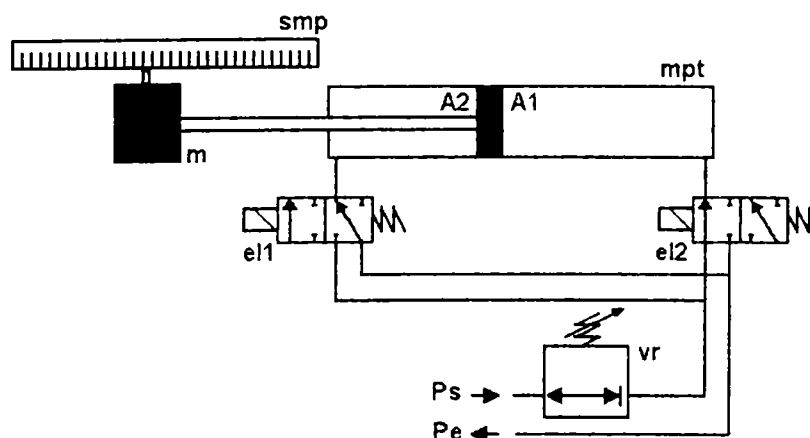


Fig.7.10. Schema sistemului pneumatic cu ventil regulator de presiune proporțional.

Modelul sistemului prezentat în ANEXA 7.2 este un model complex și deoarece simularea se realizează pe un calculator PC echipat cu un procesor PIII-

800MHz, ciclurile de antrenare a rețelelor neuronale durează foarte mult. Din această cauză, pentru simularea reglării neuronale s-a folosit un model simplificat al procesului, arătat în figura 7.11. Simplificarea constă în neglijarea factorilor ce țin de compresibilitatea aerului, și eliminarea aerului din camera opusă celei acționate.

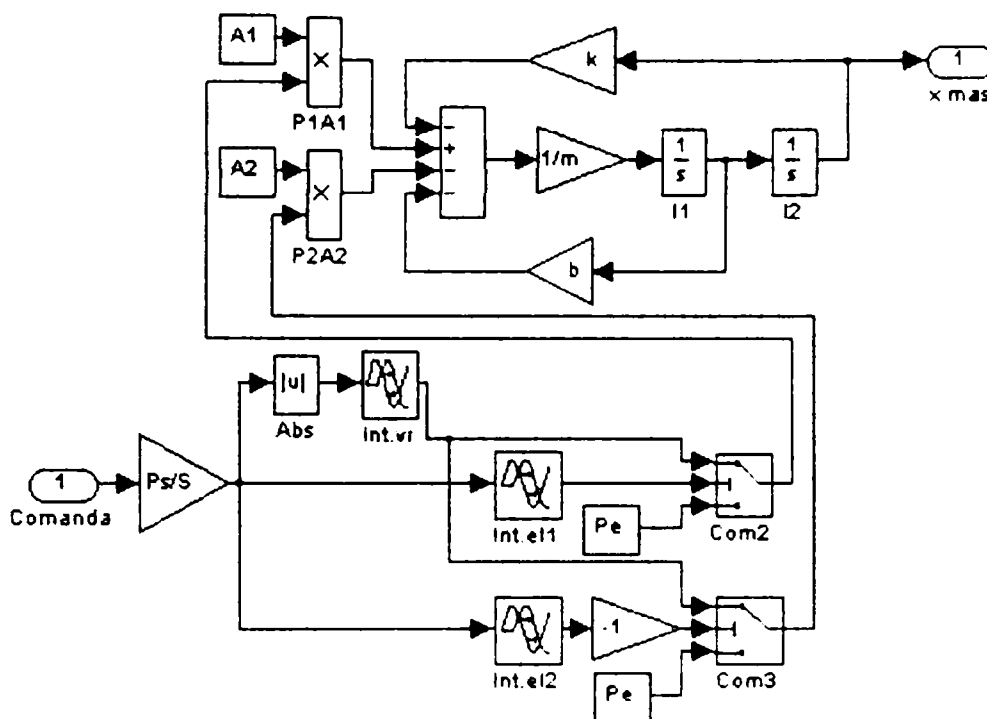


Fig.7.11. Schema simplificată a procesului („motor pneumatic liniar”).

Bazele teoretice ale metodei de reglare neuronală cu linearizarea reacției (NARMA) au fost prezentate în paragraful anterior.

Modelul procesului este integrat apoi în modelul sistemului cu regulator neuronal prezentat în figura 7.12.

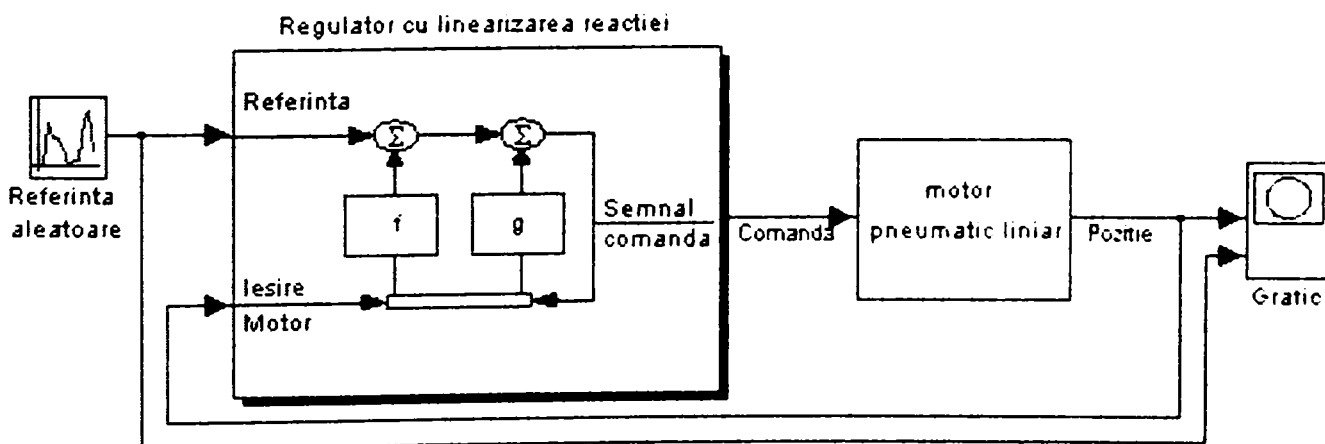


Fig.7.12. Schema sistemului pneumatic cu regulator neuronal.

După realizarea modelelor, s-au generat seturile de antrenare, validare și testare cu ajutorul modelului procesului. Pe baza acestora a fost generat modelul neuronal al procesului, regulatorul obținându-se în mod automat pe baza acestui model, după cum s-a descris în paragraful anterior. În figurile 7.13, 7.14 și 7.15 sunt prezentate seturile de intrări-ieșiri generate cu ajutorul modelului procesului din figura 7.12, ieșirile modelului neuronal cât și erorile dintre ieșirile celor două modele, pentru seturile de date de antrenare, validare și testare.

În figura 7.16 se prezintă evoluția performanței rețelei în funcție de numărul de epoci de antrenare. Pentru antrenare s-au folosit seturi de câte 3000 de date în 300 de epoci.

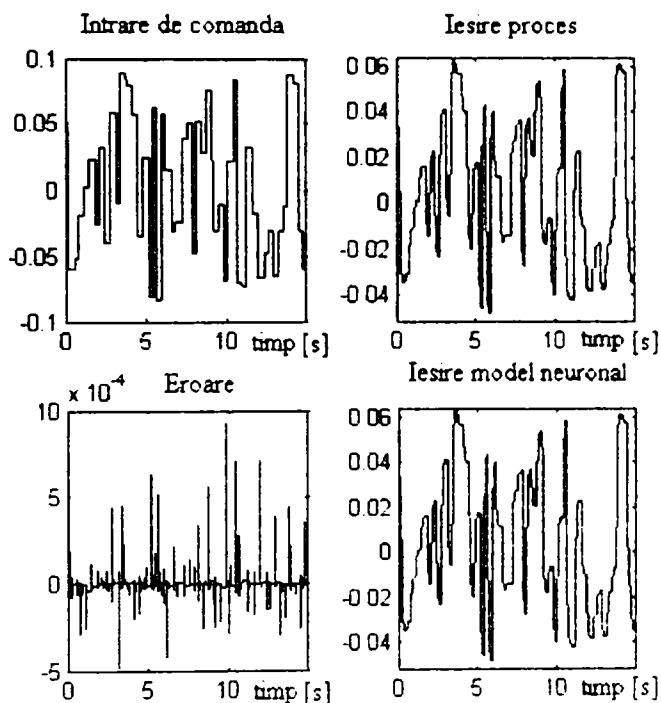


Fig.7.13. Reprezentarea grafică a datelor de antrenare.

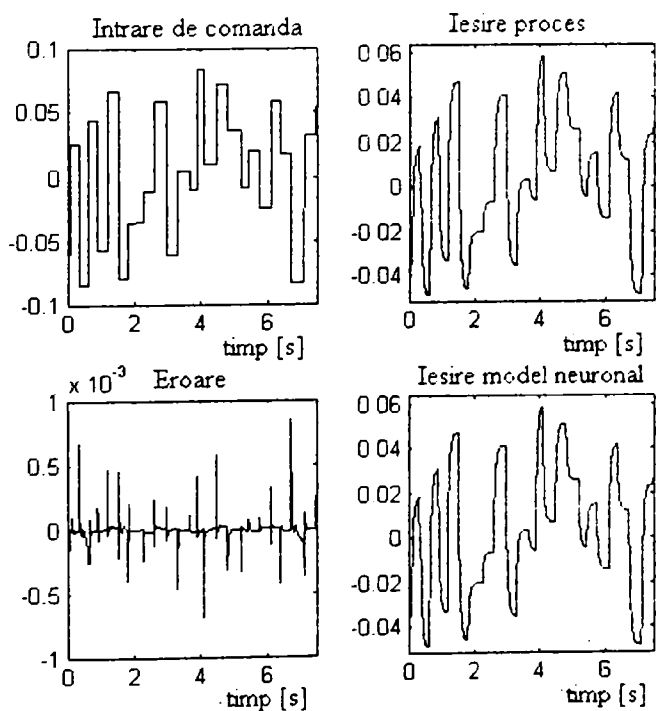


Fig.7.14. Reprezentarea grafică a datelor de validare

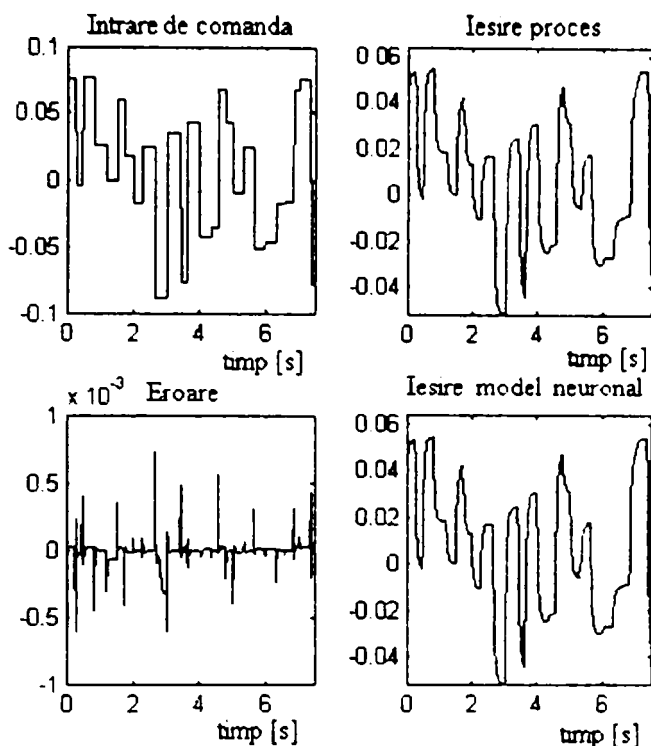


Fig.7.15. Reprezentarea grafică a datelor de testare

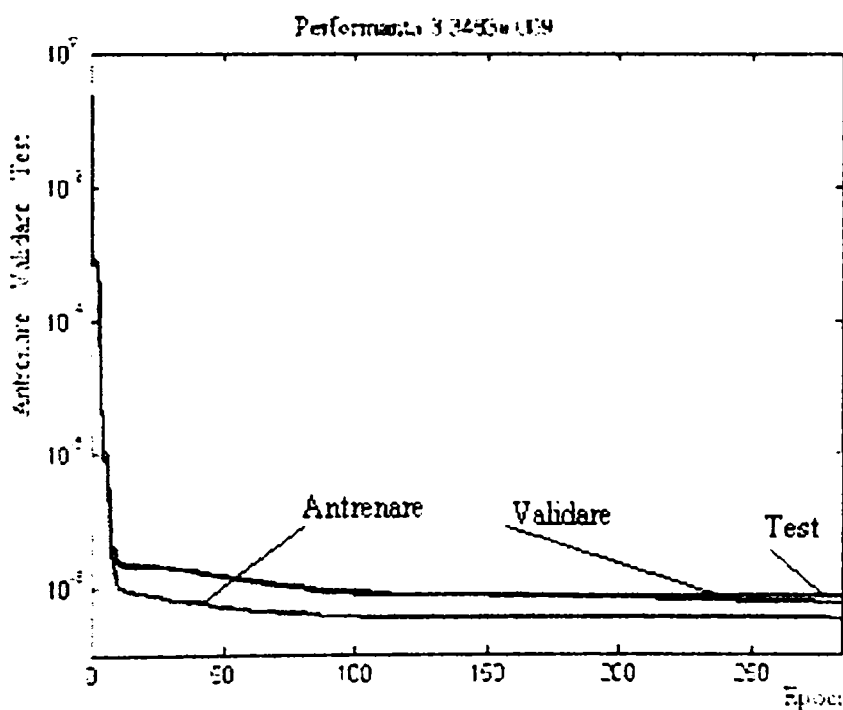


Fig.7.16. Evoluția performanțelor rețelei neuronale a modelului procesului în 300 de epoci.

După antrenarea rețelor neuronale, s-a efectuat simularea folosind un set aleator de comandă (*sc*) ce au generat semnalele de răspuns (*ps*) (figura 7.17.). Pentru a scoate în evidență mărimea erori s-a efectuat o simulare cu o referință 0,04 (semnal tip treaptă, figura 7.18.), pentru care răspunsul sistemului este de 0.042, rezultând o eroare de 0.002 (5%).

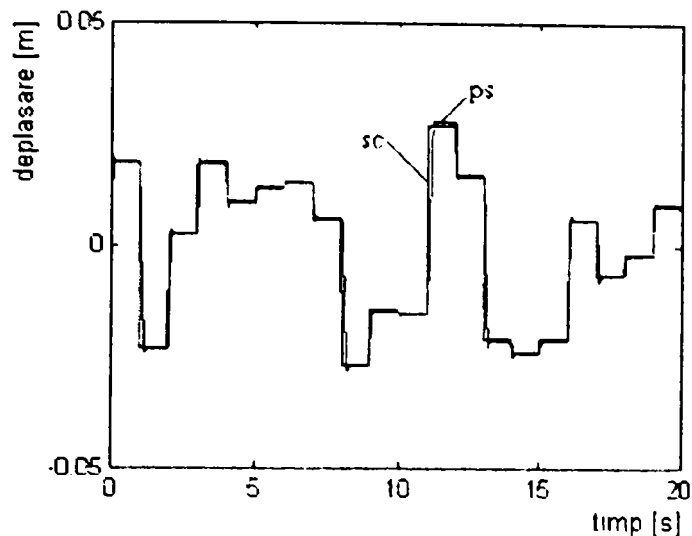


Fig.7.17. Referință aleatoare (sc),
răspunsul sistemului (ps).

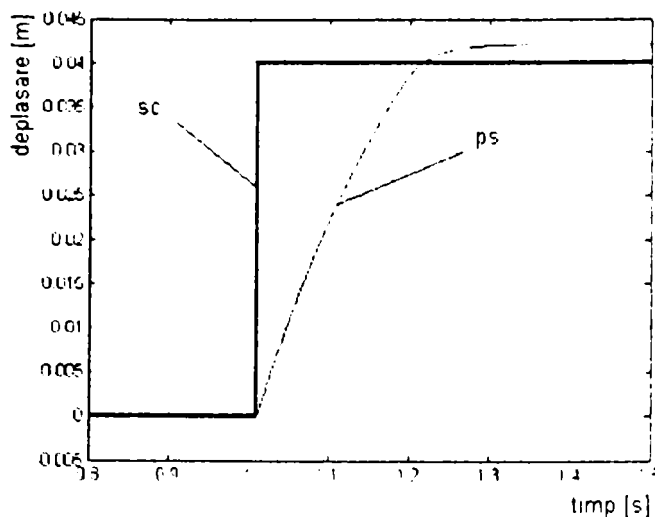


Fig.7.18. Răspuns la un semnal treaptă,
al sistemului.

7.3.4. Utilizarea unui regulator fuzzy pentru comanda în forță/poziție a unei axe de translație cu acționare pneumatică pentru șlefuirea eșantioanelor.

7.3.4.1. Considerații generale.

Există o mare diversitate de moduri în care se pot folosi sisteme de inferență fuzzy în comanda roboților, atât în privința locului în care se aplică în schema de conducere, cât și în privința formei sistemelor de inferență. Din bibliografia studiată se desprinde concluzia că sistemele de inferență fuzzy sunt folosite, printre altele, la generarea traiectoriei [JANG1], definirea modelului robotului [EMAMI1], înlocuirea reguletoarelor clasice [NOVAK1] sau în combinație cu acestea [REZNIK1]. Bazele reglajului fuzzy pentru sisteme de reglare automată sunt prezentate în [PREITL1]. În aceeași lucrare se arată că se pot realiza reguletoare fuzzy cu funcționare similară celor clasice (cvasi-PID). În alte cercetări se subliniază importanța ajustării parametrilor reguletorului, și faptul că față de reguletoarele clasice, reguletoarele fuzzy se pot ajusta într-o manieră mult mai convenabilă [REZNIK1]. Din cauza faptului că o mare parte din sistemele de inferență fuzzy au fost implementate pe baza cunoștințelor formale, euristice (funcțiile de apartenență și setul de reguli sunt definite după bunul simț tehnic al specialiștilor), nu există garanția unei funcționări sigure sau a stabilității sistemului în condiții neprevăzute. O serie de cercetări în domeniu au ca scop elaborarea unor metodologii de sinteză și analiză a sistemelor de inferență fuzzy, în domeniul roboticii [EMAMI1], [NOVAK1], sau în domeniul mai larg, dar aplicabil în robotică, al sistemelor cu reglare automată [PREITL1] (studiul stabilității sistemelor cu reguletoare fuzzy).

Deasemenea există preocupări privind elaborarea unor modele fuzzy a roboților (folosite în cinematica directă și inversă [JANG1] sau în dinamica inversă [BAPTISTA1]) care să înlocuiască modelele analitice, care necesită calcule îndelungate. Multe dintre cercetările în domeniu încearcă o abordare sistematică a proiectării sistemelor de inferență fuzzy (elaborarea unei metodologii de proiectare), care să elimine incertitudinile în alegerea funcțiilor de apartenență și a setului de reguli, ca de exemplu în [EMAMI1], în care se pune problema stabilirii riguroase a parametrilor sistemelor de inferență fuzzy, pe baza unei metodologii clare.

Mediul de programare MATLAB dispune de funcții predefinite pentru realizarea diferitelor etape ale calculului fuzzy (fuzzyficare, inferență, defuzzyficare).

Aceste funcții sunt legate de două module externe (realizate în C++) și anume: “sistemul de inferență” și “motorul fuzzy”. Prin intermediul funcțiilor predefinite există posibilitatea apelării și a mediului de simulare SIMULINK. Aplicațiile utilizator se leagă de modulele amintite prin intermediul funcțiilor predefinite. Schema legăturii dintre aceste module este arătată în figura 7.19.

Structura de bază tipică a sistemelor de inferență fuzzy reprezintă un model ce realizează o anumită corespondență:

valoare fermă de intrare - funcții de apartenență de intrare - reguli de inferență - caracteristici de ieșire - funcții de apartenență de ieșire - valoare fermă de ieșire.

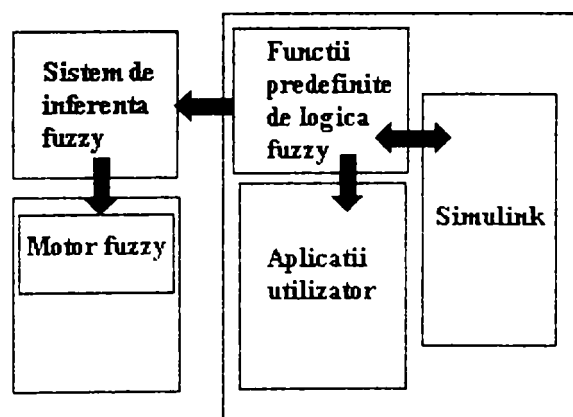


Fig. 7.19. Schema legăturii dintre modulele fuzzy și SIMULINK în mediul MATLAB.

Deasemenea, un sistem de inferență fuzzy tipic, presupune o structură de reguli predefinite de către utilizator, ce încearcă să interpreteze caracteristicile variabilelor modelului. Dacă însă se dorește realizarea unui sistem de inferență fuzzy pentru modelarea unui proces la care există seturi de date intrare-ieșire stabilite experimental, acestea se pot utiliza pentru generarea automată a funcțiilor de apartenență, deci implicit pentru identificarea sistemului. În acest caz strategia de identificare folosită poate fi o strategie neuro-fuzzy, care are la bază învățarea informațiilor din setul de date pentru a genera parametrii funcțiilor de apartenență. În cadrul mediului MATLAB ajustarea parametrilor amintiți se poate realiza cu un modul ce funcționează

în mod similar cu o rețea neuronală, numit ANFIS (Adaptive Neural Fuzzy Inference System). Ca algoritm de învățare se folosește propagarea înapoi a erorii iar pentru optimizarea procesului de învățare ANFIS utilizează metoda gradientului, urmată de un algoritm de optimizare bazat pe suma pătratică a erorilor dintre mărimea de ieșire dorită și cea generată.

7.3.4.2. Modelarea sistemului de șlefuire cu regulator fuzzy.

Aplicația ce folosește regulator fuzzy a fost dezvoltată pe sistemul descris în paragraful 7.3.2, modificat prin introducerea unui senzor de forță (figura 7.20.), pentru a se realiza aplicația completă a șlefuirii eșantioanelor după ciclul de lucru descris în paragraful 7.3.2.

S-au specificat anterior cerințele impuse funcționării sistemului pneumatic. Sub formă grafică, modul în care trebuie să funcționeze sistemul este arătat în diagramele din figura 7.21.a. pentru poziție și 7.21.b. pentru forță.

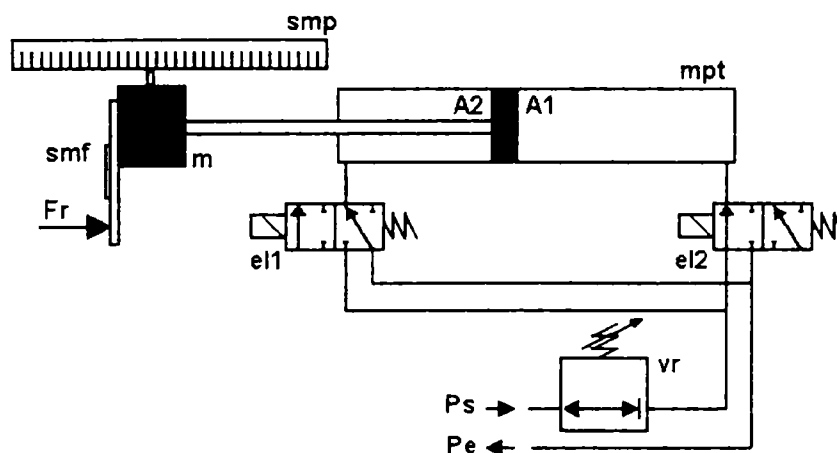


Fig.7.20. Schema sistemului pneumatic cu reacție în forță-poziție.

Din cele expuse până acum se observă următoarele aspecte :

- se vor defini două variabile lingvistice de intrare și anume "Poziție" și respective "Forță" și o variabilă de ieșire "Semnal de comandă";
- referința de poziție este dată la limita superioară a domeniului de incertitudine ($D2$);
- reglarea finală se va realiza după variabila "Forță";
- pentru poziție vor exista două domenii ce trebuiesc definite prin termeni lingvistici (Mare, Mică), iar pentru forță șapte termeni lingvistici (Mare Negativă, Medie Negativă, Mică Negativă, Zero, Mică Pozitivă, Medie Pozitivă, Mare Pozitivă);

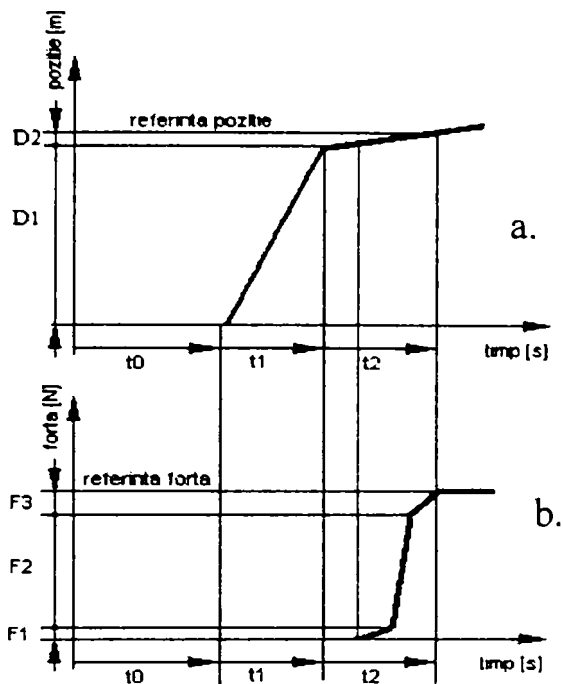


Fig.7.21. Diagramele funcționării în poziție (a) și în forță (b) a sistemului pneumatic.

Modelarea axei de translație pneumatice cu regulator fuzzy.

S-au realizat două modele cu regulator fuzzy : cu reacție în poziție, și cu reacție în forță-poziție, ce vor fi prezentate în continuare.

Reacție în poziție.

Modelul pentru reglarea fuzzy cu reacție în poziție este prezentat în figura 7.22. În acest caz s-a folosit sistemul fără reacție în forță, prezentat în paragraful 7.3.3. Acest model a fost realizat doar ca un studiu preliminar și acoperă doar domeniul de lucru în care se presupune că eșantionul este în contact cu abrazivul.

În figura 7.23. este arătată diagrama de inferență a sistemului, ce conține :

- funcțiile de apartenență de intrare de tip “clopot Gaussian” ce descriu cinci termeni lingvistici ai variabilei de intrare “Eroare de poziție”;
- funcțiile de apartenență de ieșire de tip triunghiular de asemenea descriind cinci termeni lingvistici ai variabilei de ieșire “Semnal de comandă”;
- defuzzificarea se realizează cu funcția “centroidă”.

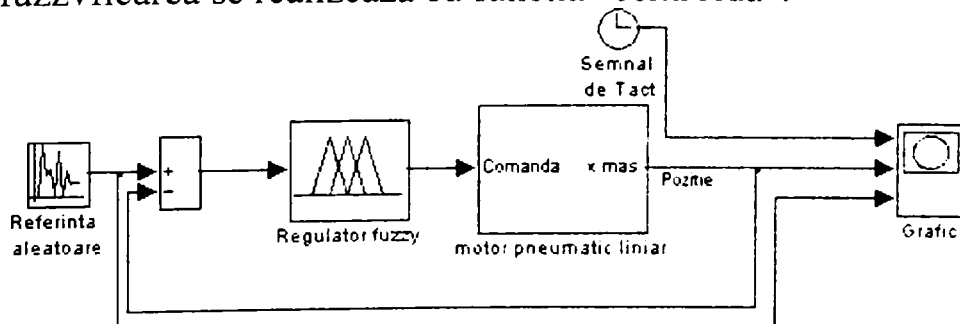


Fig7.22. Modelul sistemului cu reglare fuzzy după poziție.

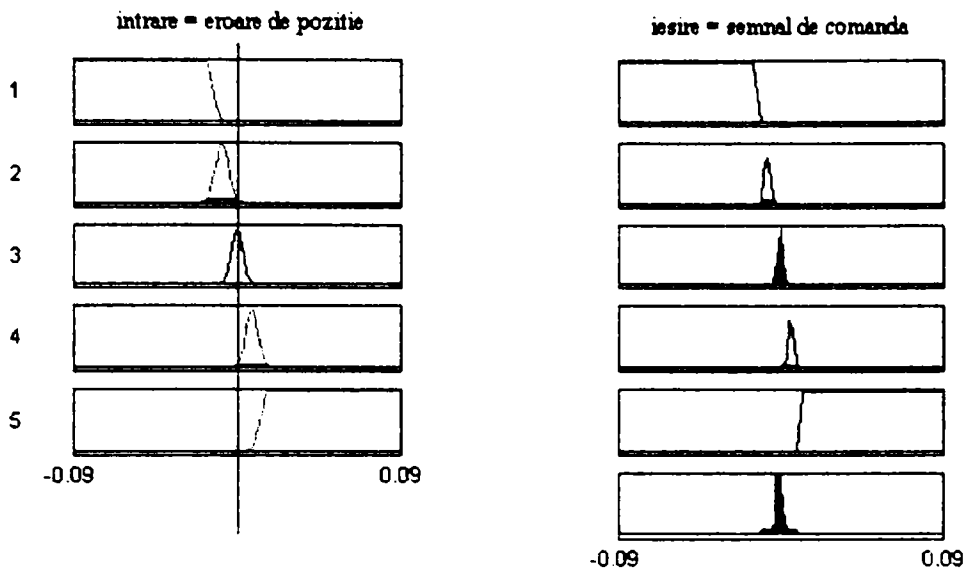


Fig.7.23. Diagrama de inferență pentru sistemul cu regulator fuzzy și reacție în poziție.

Rezultatul simulării pentru o referință aleatoare este arătat în figura 7.24.

Din analiza rezultatelor simulării se poate observa că parametrii regulatorului fuzzy necesită ajustări, deoarece întârzierile sunt mari (de ordinul secundelor).

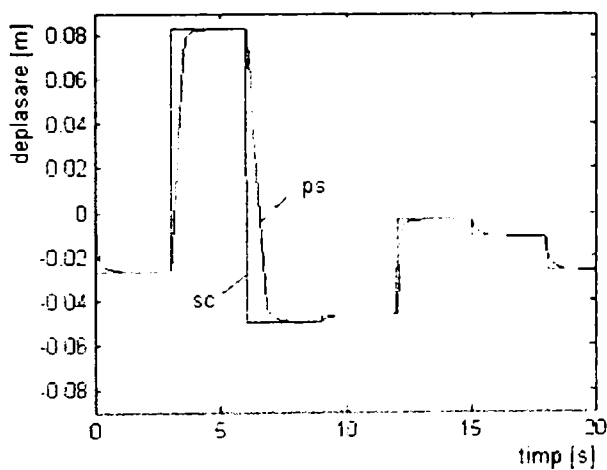


Fig.7.24. Diagrama semnalelor de comandă și răspuns pentru sistemul cu reglare fuzzy după poziție.

Reacție în forță-poziție.

Modelul sistemului cu regulator fuzzy care a fost conceput pentru a rezolva cerințele reglării în forță-poziție este prezentat în figura 7.25.

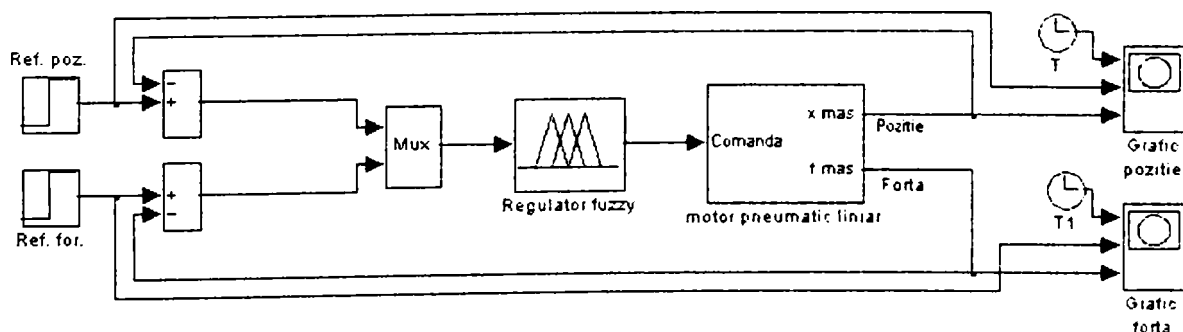


Fig.7.25. Modelul sistemului cu reglare fuzzy după reacție în forță- poziție.

Diagrama de inferență a sistemului este prezentată în figura 7.26.

Pentru ambele intrări s-au folosit funcții de apartenență de tip “clopot Gauss” iar pentru ieșire s-au folosit funcții de tip triunghiular.

Rezultatele simulării pentru excitație de tip treaptă sunt arătate în figurile 7.27. și 7.28. Din forma semnalului de răspuns se observă că acestea corespunde în mare măsură cerințelor specificate.

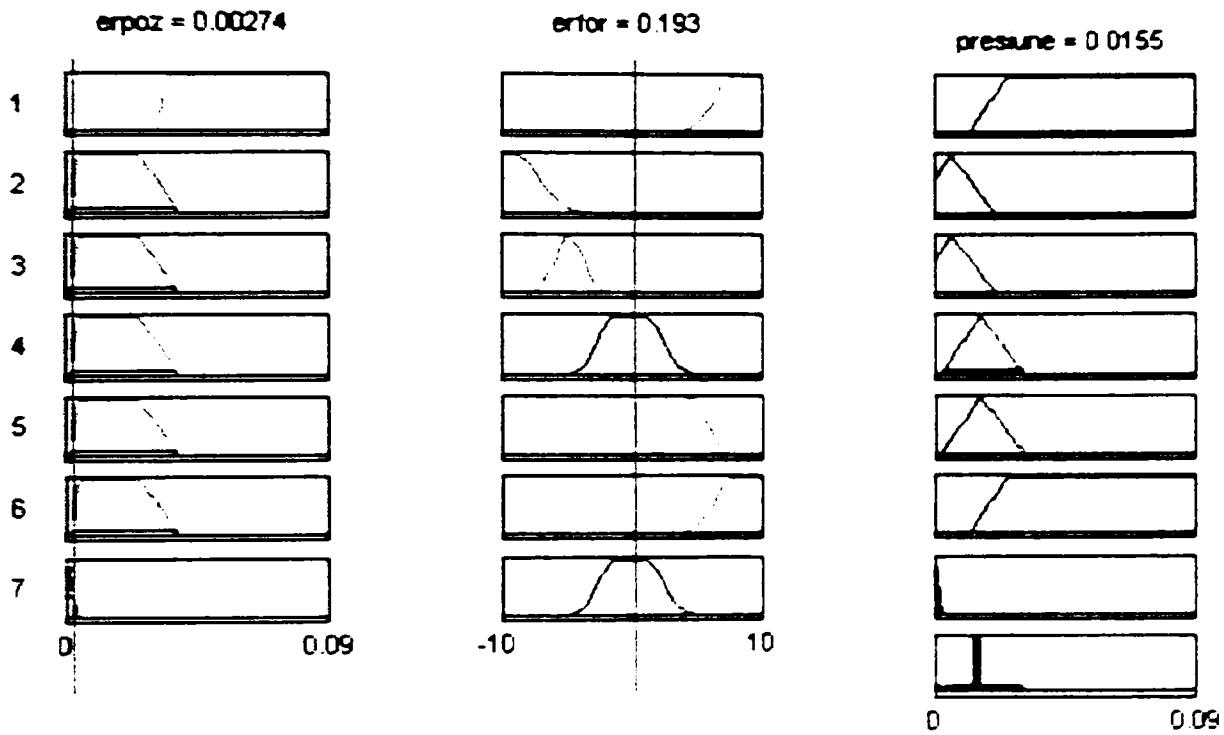


Fig.7.26. Diagrama de inferență a sistemului cu reglare fuzzy în forță- poziție.

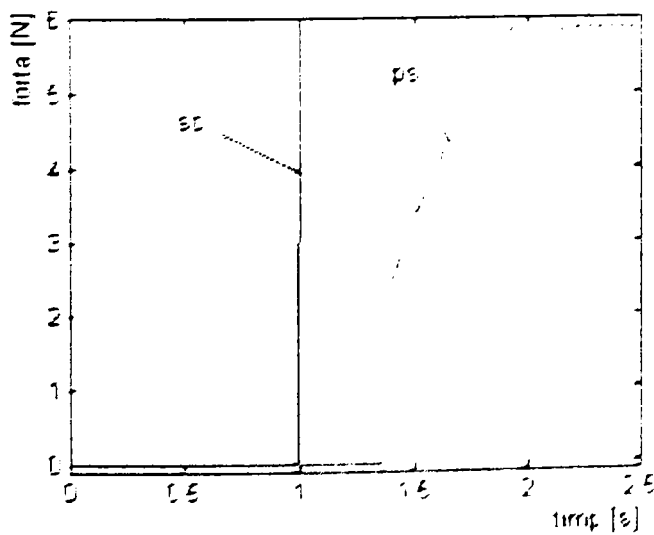


Fig.7.27. Diagrama semnalelor de referință și reacție pentru forță.

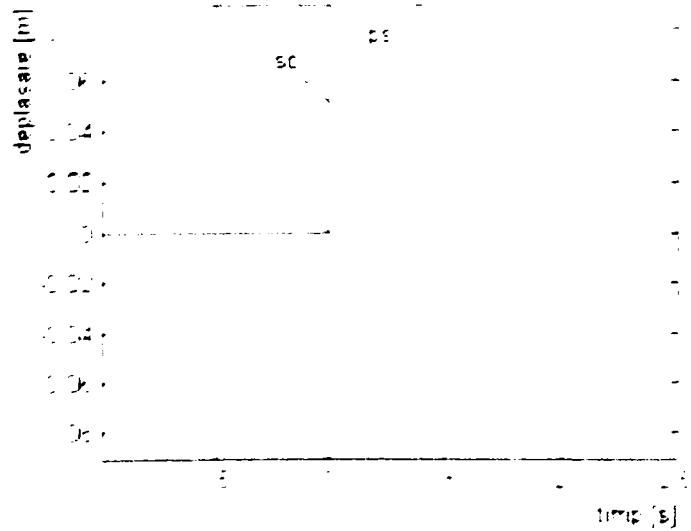


Fig.7.28. Diagrama semnalelor de referință și reacție pentru poziție.

7.4. Realizarea sistemului experimental cu regulator fuzzy.

Sistemul experimental pentru testarea funcționării reglării fuzzy în forță și poziție este arătat în figura 7.29. Ideea de bază în această concepție a fost folosirea unei plăci de achiziție de date PCI6023E (National Instruments), pentru care există funcții predefinite, în modulul „Data Acquisition Toolbox” din mediul MATLAB®. Acest lucru face posibilă folosirea sistemelor de inferență fuzzy (prin intermediul fișierelor (FIS), în programul de achiziție și comandă fără a părăsi mediul MATLAB®. O altă posibilitate ar fi folosirea blocurilor „xPCTarget”, dar aceasta ar necesita un calculator dedicat rulării codului compilat în limbaj C, care nu s-a realizat datorită unor criterii economice. Sistemul conceput, este lent, în comparație cu un sistem ce ar folosi variante soft compilate dar poate realiza testarea logică a algoritmului regulatorului fuzzy, chiar dacă nu la parametrii ceruți de aplicația industrială.

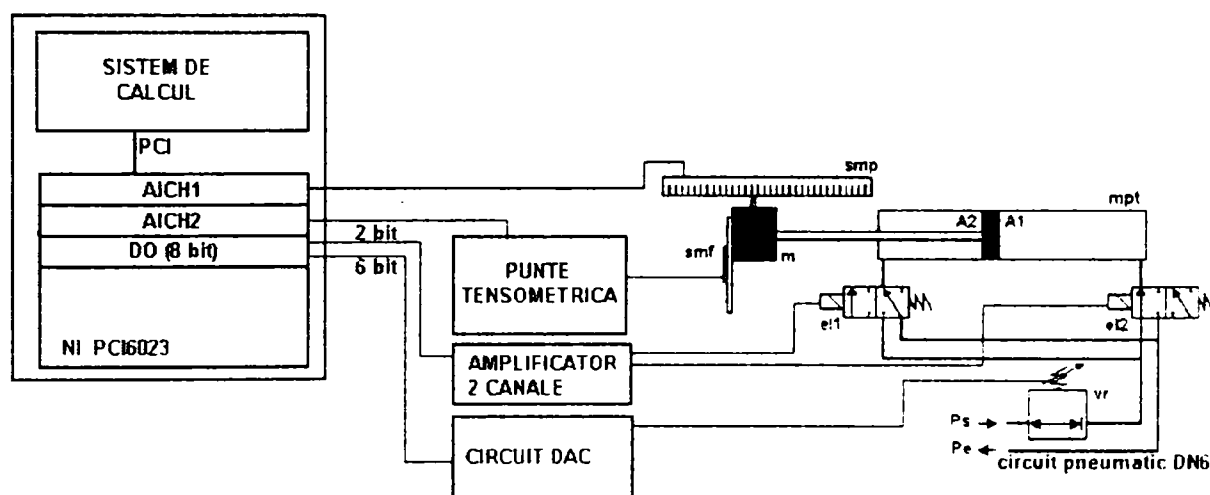


Fig.7.29. Schema sistemului experimental.

Modulul de achiziție posedă 8 canale de intrare analogică (ACH1....8) pe 12 biți, dintre care s-au folosit ACH1 și ACH2 (figura 7.29.).

Un inconvenient major al plăcii PCI6023 este lipsa unei ieșiri analogice ce ar fi putut fi folosită pentru comanda ventilului regulator de presiune (vr). Problema a fost rezolvată prin folosirea a șase din cele opt canale digitale configurabile (I/O) și a unui circuit de conversie numeric-analog (N/A). Se pot obține astfel 64 trepte de reglare a presiunii ce sunt suficiente pentru scopul experimentelor.

Nu se pot folosi toate cele 8 canale digitale (pentru o rezoluție mai bună, deoarece două dintre acestea sunt necesare pentru comanda ventilelor de cale (on-off), utilizate la schimbarea sensului de mișcare al pistonului.

Sistemul experimental a fost realizat în cadrul laboratorului de Mecatronică al Universității din Oradea, fotografiile acestuia și a unor componente fiind arătate în figurile 7.30-7.33.

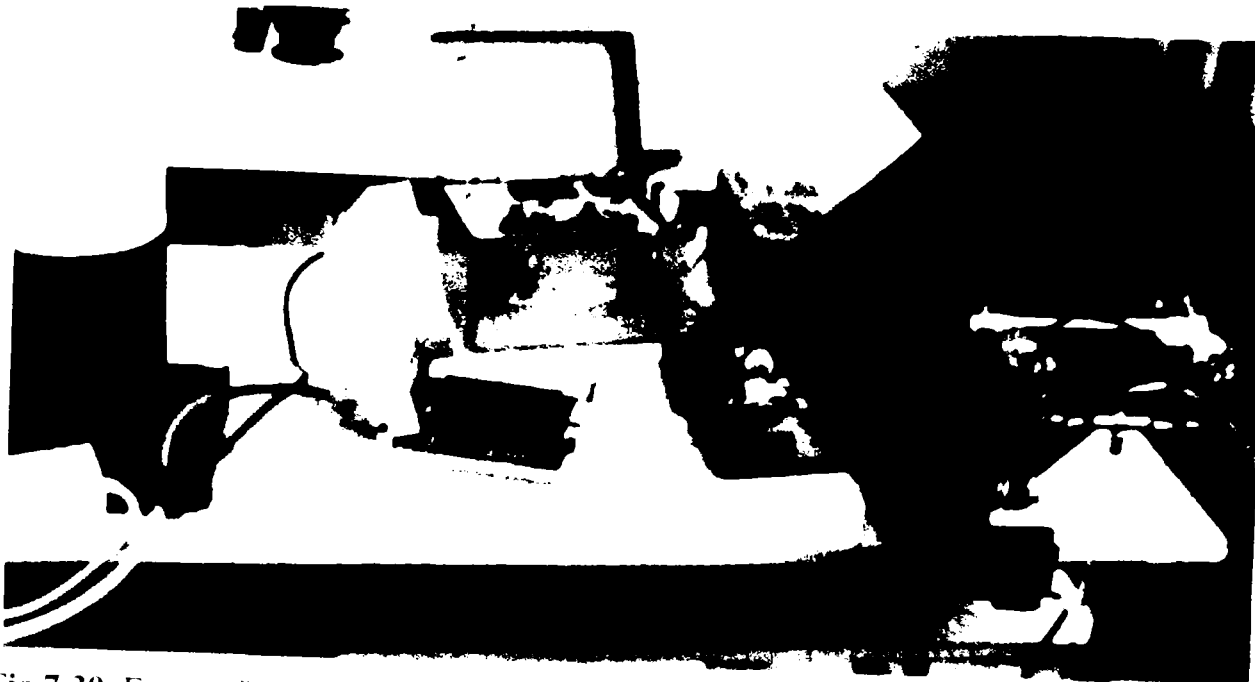


Fig.7.30. Fotografia sistemului experimental – blocul de conectare al plăcii de achiziție, punte tensometrică (centru), sistem pneumatic (dreapta).

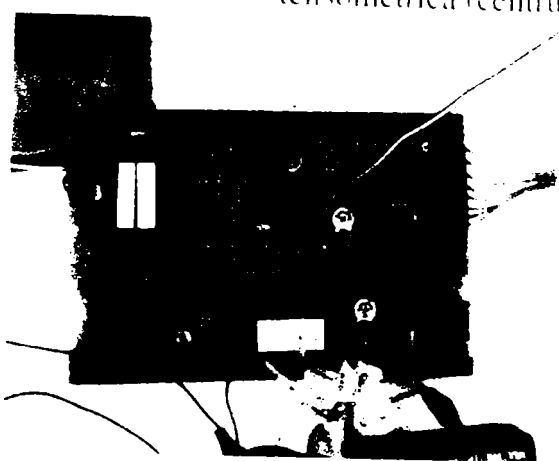


Fig.7.31. Circuitul de conversie N/A

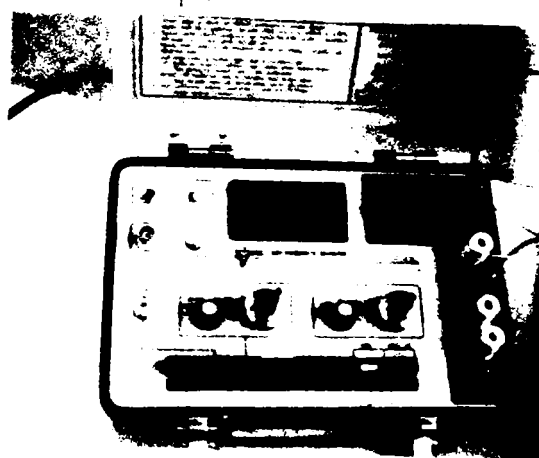


Fig.7.32. Punte tensometrică P3500 (Vishay Measurements Group), timbru tensometric MI20 (Microtehnă Prahă)



Fig.7.33. Fotografia sistemului pneumatic cu ventilul regulator de presiune proporțional de precizie MPPLS-3-1/8-10-010-B, FESTO (jos).

7.5. Concluzii privind funcționarea sistemului cu reglare fuzzy după poziție și forță.

În urma analizei rezultatelor simulării și a funcționării sistemului realizat s-a desprins concluzia că regulatorul fuzzy folosit necesită ajustări, în continuare, pentru mărirea robusteții reglajului.

În funcționarea actuală, apar efecte nedorite ca :

- eroarea de reglaj a poziției este constantă, dar reacția rămâne sub valoarea referinței cu circa 10% pentru o plajă de valori de referință de la 6 la 9;
- domeniul de incertitudine în care eșantionul poate întâlni abrazivul, și pentru care reglajul mai funcționează acceptabil, este mic (aproximativ 3mm) față de situațiile ce se pot întâlni în practică (estimativ 5-6 mm);
- eroarea de reglaj a forței crește odată cu mărirea referinței.

Deși, pe anumite domenii de lucru reglajul este acceptabil, se impune îmbunătățirea regulatorului pentru mărirea robusteții acestuia.

8. Analiza automată a stării suprafeței.

8.1. Necesitatea analizei rugozității.

În capitolul 6 s-au enunțat principalele modalități de creștere a productivității, printre care și reducerea duratei de șlefuire și a numărului de cicluri de șlefuire-verificare a rugozității suprafeței. În primă fază s-a estimat că este suficient ca analiza rugozității suprafeței să se facă doar într-o formă mai simplă și anume: estimarea duratei de șlefuire numai pe baza durtății ce se presupune a fi cunoscută la intrarea eșantionului în sistem. În acest caz este suficient să se realizeze doar o verificare a conformității parametrilor stării suprafeței cu parametri necesari unei analize metalografice corecte (figura 8.1). Aceasta presupune existența unui sistem (blocul „Verificarea rugozității suprafeței”) ce decelează doar starea suprafeței de rugozitate sub $R_a=0,2$ de orice altă stare. Numărul de cicluri șlefuire-verificare va fi dependent de rugozitatea inițială a suprafeței : la o rugozitate inițială mare numărul de cicluri va fi mare.

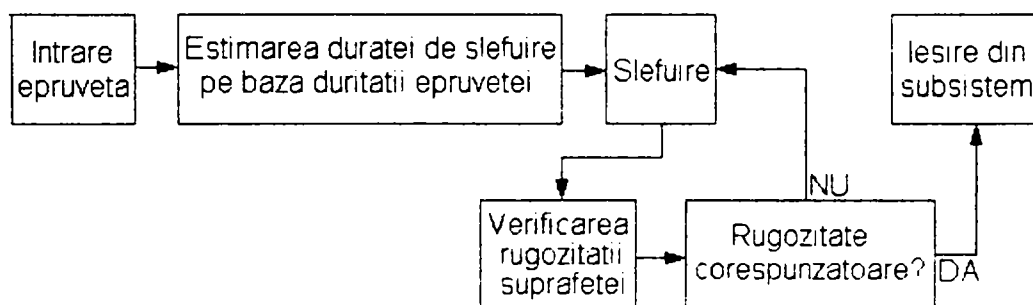


Fig.8.1. Schema explicativă a ciclului de lucru în subsistemul 1, varianta 1.

Prin reducerea numărului de cicluri se reduc timpii de transfer între postul de șlefuire și postul de analiză a suprafeței și deasemenea se reduce numărul de verificări ale stării suprafeței reducându-se astfel durata de timp pe care eșantionul o petrece în subsistemul 1. Pe baza analizei rugozității inițiale a eșantionului se poate estima mai exact durata de șlefuire reducându-se numărul de cicluri de verificare-șlefuire (figure 8.2.). Aceasta implică existența unui algoritm de clasificare complex capabil să decidă asupra apartenenței eșantionului intrate în sistem la o anumită grupă. Pe baza clasificării stării suprafețelor pe anumite grupe se poate estima mai exact durata primei șlefuirii, prin corelarea fiecărei grupe cu o anumită durată.

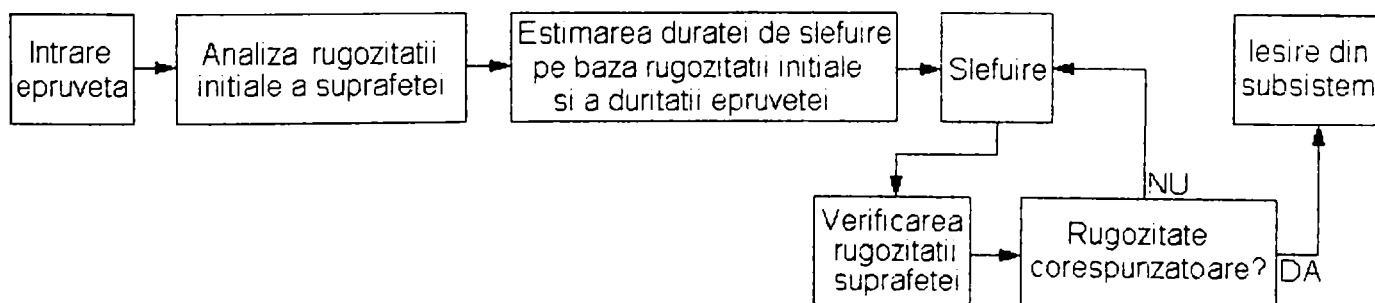


Fig.8.2. Schema explicativă a ciclului de lucru în subsistemul 1, varianta 2.

Esențial în realizarea sistemului în varianta 2 (figura 8.2.), este analiza rugozității inițiale. Pentru rezolvarea acestei probleme s-a considerat necesară o analiză a parametrilor ce caracterizează rugozitatea suprafețelor.

8.2. Caracterizarea rugozității suprafeței.

Rugozitatea suprafeței este definită în SR ISO 4287/2001 prin specificarea caracteristicilor cuantificate prin parametrii ca :

- parametrii de amplitudine;
- înălțimea maximă de proeminență a profilului;
- adâncimea maximă de gol a profilului;
- înălțimea maximă a profilului;
- înălțimea medie a parametrilor profilului;
- parametrii de amplitudine :
 - abaterea medie pătratică a profilului evaluat;
 - abaterea medie aritmetică a profilului evaluat;
 - factorul de asimetrie al profilului evaluat;
 - factorul de aplatizare a profilului evaluat;
- parametrii de pas;
- parametrii hibridi (panta medie pătratică a profilului).

Atât parametrii menționați cât și alți parametrii mai complecși sunt explicați pe larg în diferite lucrări [DRAGU, COSTA]. În aceste lucrări se consideră utilizarea profilometrelor pentru măsurarea rugozității după o traiectorie liniară. O reprezentare bidimensională a rugozității se obține atunci când sunt măsurate o serie de traiectorii liniare alăturate.

Abaterea medie aritmetică a rugozității R_a [DRAGU] este dată de relația:

$$R_a = \frac{1}{L} \int_0^L |y_R - R_p| dx_R \quad (8.1)$$

$$R_p = \frac{1}{L} \int_0^L y_R dx_R \quad (8.2)$$

unde

L – lungimea liniei de referință, y_R – înălțimea rugozității; dx_R - distanța de-a lungul dimensiunii L .

Relația (8.1) se poate scrie sub formă discretă :

$$R_a = \frac{\sum_{i=1}^n |y_i|}{n} \quad (8.3)$$

unde

y_i – înălțimea în puncte discrete; n – numărul de puncte de măsurare.
Adâncimea medie în 10 puncte a rugozității R_z :

$$R_z = \frac{(R_1 + R_3 + R_5 + R_7 + R_9) - (R_2 + R_4 + R_6 + R_8 + R_{10})}{5} \quad (8.4)$$

unde

R_1, R_2, \dots, R_{10} – este înălțimea a zece vârfuri consecutive.

La obținerea parametrilor de rugozitate se are în vedere orientarea neregularităților dată de direcția predominantă care în general este determinată de procedeul utilizat pentru prelucrarea suprafeței.

În [COSTA] se subliniază faptul că rugozitatea este o caracteristică importantă a suprafețelor obținute pe cale tehnologică. În aceeași lucrare se specifică mărimile statistice utilizate în caracterizarea rugozității. Se utilizează des diferiți parametri derivați din funcția densitate de probabilitate (FDP).

Pe lângă R_a se amintește amplitudinea (înălțimea) pătratică medie R_q (abaterea pătratică medie de la linia de referință), dată de relația :

$$R_q = \frac{1}{L} \int_0^L y_R^2 dx_R \quad (8.5)$$

În afară de parametrii de amplitudine se utilizează parametrii pentru descrierea formei distribuției rugozităților pentru exprimarea cărora se folosesc momentele statistice de ordinul 3 și 4 ale funcției densitate de probabilitate ce arată asimetria respectiv aplatizarea FDP.

Asimetria (Skew) :

$$S_k = \frac{1}{\sigma^3} \int_{-\infty}^{+\infty} (z - \bar{z})^3 p(z) dz \quad (8.6)$$

unde

σ – dispersia; z – înălțimea profilului ; $p(z)$ – probabilitatea de apariție a înălțimii z .

Aplatizarea (Kurtosis) :

$$K = \frac{1}{\sigma^4} \int_{-\infty}^{+\infty} (z - \bar{z})^4 p(z) dz - 3 \quad (8.7)$$

Parametrii de spațiere sunt caracterizați cu ajutorul funcției de autocorelație :

$$A(\tau) = \lim_{L \rightarrow \infty} \int_{-L/2}^{L/2} z(x)z(x + \tau) dx \quad (8.8)$$

unde

τ – spațierea; $z(x)$ – înălțimea profilului x – coordonata de-a lungul traiectoriei.

În practică se utilizează formele discrete ale relațiilor amintite, pentru că instrumentele măsoară valoarea rugozității în puncte discrete.

Numeroasele caracterizări ale rugozității conduc adesea la rezultate contradictorii obținute de diferiți cercetători chiar și pentru același set de date. O altă problemă ce se constată la utilizarea parametrilor statistici este aceea că este ușor să se găsească două suprafețe diferite ce au aceleași valori pentru acești parametri [COSTA]. Mulți dintre parametrii statistici sunt parțial dependenți de

definiții arbitrare. Cea mai mare dificultate în utilizarea parametrilor statistici este ca aceștia să fie invariabili la condițiile de măsurare. Eliminarea erorilor de formă și a undulațiilor este de asemenea dificilă.

Sunt definiți de asemenea o serie de parametri statistici bidimensionali. Calculul acestora se realizează utilizând înălțimea rămasă $w(x,y)$ după extragerea planului de aproximare mediu al suprafeței respective.

Media pătratică a înălțimilor, în formă discretă este dată de relația :

$$S_q = \sqrt{\frac{1}{MN} \sum_{j=1}^N \sum_{i=1}^M w^2(x_i, y_j)} \quad (8.9)$$

unde

M, N – dimensiunile suprafeței; x_i și y_j – coordonatele punctului curent.

Asimetria curbei FDP:

$$S_{sk} = \frac{1}{MNS_q^3} \sum_{j=1}^N \sum_{i=1}^M w^3(x_i, y_j) \quad (8.10)$$

Aplatizarea curbei FDP:

$$S_k = \frac{1}{MNS_q^4} \sum_{j=1}^N \sum_{i=1}^M w^4(x_i, y_j) \quad (8.11)$$

Densitatea vârfurilor pe unitatea de suprafață :

$$S_{\phi} = \frac{n_v}{(M-1)(N-1)\Delta x \Delta y} \quad (8.12)$$

unde

n_v – numărul de vârfuri pe profil; $\Delta x, \Delta y$ – pasul pe direcția x respectiv y .

Funcția de autocorelație pe suprafață se calculează utilizând spațieri diferite τ_x pe direcția x și τ_y pe direcția y , $z(x,y)$ fiind înălțimea profilului. Prima dată se calculează funcția de auto-covarianță, pe lungimile de referință L_x și L_y :

$$A_c(\tau) = \frac{1}{(L_x - \tau_x)(L_y - \tau_y)} \int_0^{(L_x - \tau_x)} \int_0^{(L_y - \tau_y)} z(x,y)z(x + \tau_x, y + \tau_y) dx dy \quad (8.13)$$

și apoi funcția de auto-corelație pe suprafață este dată de raportul :

$$A_s = \frac{A_c(\tau_x, \tau_y)}{A_c(0,0)} \quad (8.14)$$

Funcția de autocorelație ar fi util să se folosească în caracterizarea rugozității pentru analiza rugozității inițiale a eșantioanelor metalografice dar selectarea parametrului τ este dificilă.

O abordare similară funcției de autocorelație este utilizarea parametrilor frecvențiali obținuți cu ajutorul transformatei Fourier bidimensionale (transformarea Fourier rapidă – FFT2) descrisă în capitolul 3.

Măsurarea rugozității cu ajutorul aparatelor ce folosesc palpatoare (profilometre), a fost dezvoltată și îmbunătățită mai ales în scopul măririi preciziei [WIECZROWSKI1, LEACH1, BRINKMANN1]. Cu toate acestea mai ales datorită duratei de măsurare îndelungate și a necesității asigurării contactului dintre piesă și palpator, metoda este inefficientă în cazul automatizării. Din această cauză se utilizează din ce în ce mai mult analiza imaginii suprafeței, ce poate să redea aspecte relevante ale rugozității acesteia [LEE1, JOSSO1, YANAGI1]. În acest caz imaginea suprafeței este achiziționată cu ajutorul unei camere CCD, cuplată la un microscop de mărire mici (maxim x100), iar pentru analiza rugozității se utilizează tehnici de procesare a imaginilor, o deosebită importanță având metodele de analiză a texturilor (capitolul 3).

8.3. Stabilirea metodelor de lucru.

În cadrul tezei de doctorat s-a utilizat analiza imaginii suprafeței pentru identificarea rugozității inițiale a eșantionului cât și pentru verificarea rugozității obținute prin șlefuire. Sistemul de achiziție a imaginilor suprafeței eșantioanelor a fost inclus în cadrul CFLM1 și CFLM2 și a fost realizat practic (capitolul 5.8).

Metoda de lucru utilizată constă în stabilirea unor domenii de variație a caracteristicilor suprafeței pentru etaloane (C_{et}) de rugozitate pentru diferite procedee tehnologice și diferite rugozități pentru fiecare procedeu în parte (figura 8.3). Pe baza domeniilor de variație stabilite pot fi constituite grupe (G_{et}), în care se încadrează caracteristicile determinate. Odată stabilite aceste domenii, caracteristicile suprafeței unui eșantion oarecare (C_{ep}) se pot calcula în același mod ca și caracteristicile suprafețelor etaloanelor și apoi, prin stabilirea apartenenței C_{ep} la o anumită grupă G_{et} , se poate determina rugozitatea suprafeței eșantionului (figura 8.4).

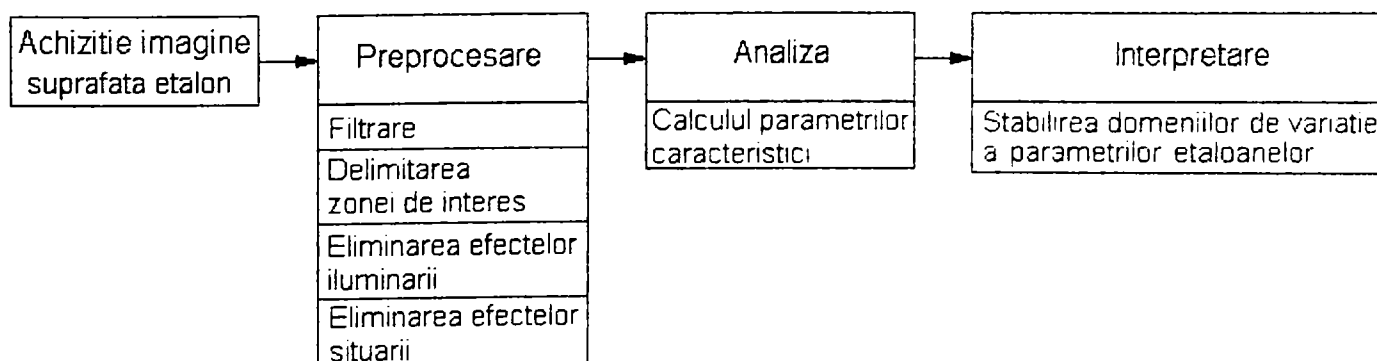


Fig.8.3. Schema algoritmului de determinare grupelor de rugozitate (G_{et}).

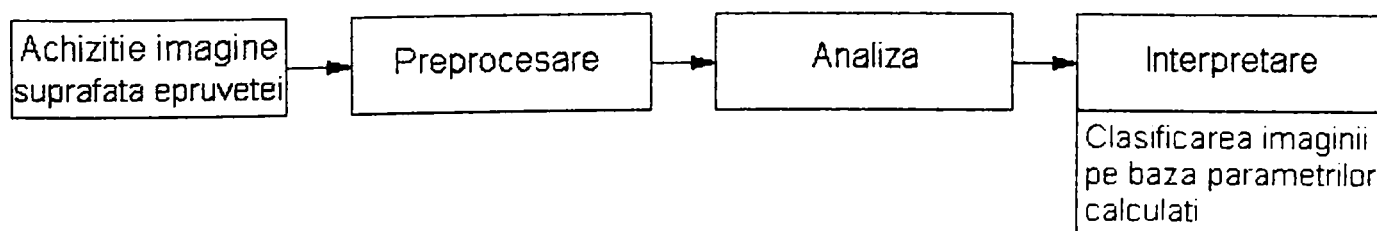


Fig.8.4. Schema algoritmului de clasificare a unui eșantion pe baza C_{ep} .

În ambele situații prezentate în schemele din figurile 8.3. și 8.4. primele trei faze sunt identice.

Etalonele utilizate (Trusă de etaloane de rugozitate STALÍ DOVODCA – GOST 9378-80 E15718) sunt codificate în funcție de rugozitate și procedeul tehnologic de prelucrare. S-au luat în considerare doar rugozitățile obținute prin procedeele tehnologice specifice realizării unor suprafețe plane, acestea fiind majoritare în analiza metalografică. Procedeele tehnologice alese sunt :

- frezare cilindrică (cod : fc5, fc6 fc7) ;
- frezare frontală (cod : ff5, ff6, ff7);
- rabotare plană (cod : rab4, rab5, rab6, rab7);
- rectificare frontală (cod : rf6, rf7, rf8);
- rectificare plană (cod : rp6, rp7, rp8, rp9);
- netezire plană (cod : np10, np11, np12, np13);

Codul utilizat se compune dintr-un grup de litere ce specifică tipul prelucrării (de exemplu frezare frontală : „ff” urmat de o cifră ce codifică rugozitatea etalonului dată în tabelul 1 din ANEXA 8.1. Pentru urmărirea dependenței de scară a mărimilor caracteristice, s-au achiziționat imagini ale etaloanelor la mărimi de x10 respectiv x25 (mărimi existente la microscopul CITIVAL). Rezoluția asigurată de camera CCD utilizată (Trust Space C@M) a fost de 640x480 pixeli.

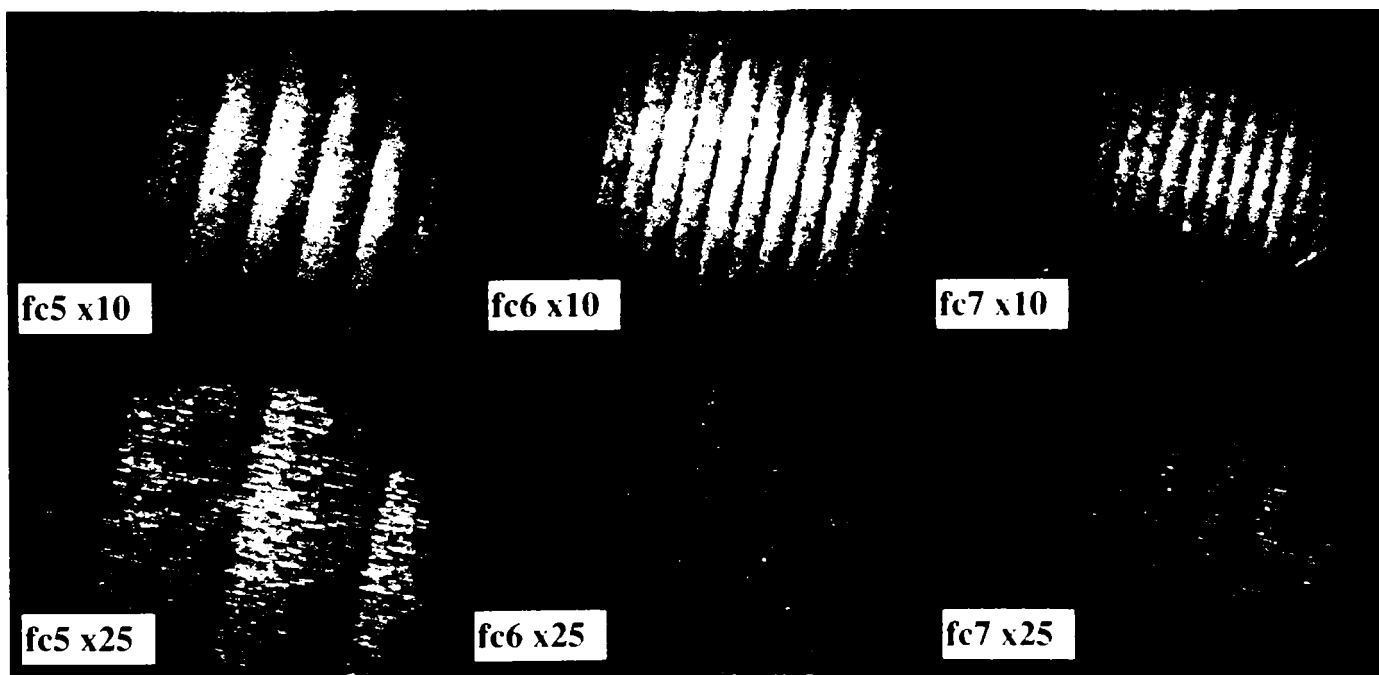


Fig.8.5. Imagini achiziționate ale etaloanelor : frezare cilindrică (fc) pentru trei rugozități (5.6.7) și două mărimi x10 respectiv x25 la rezoluție de 640x480 pixeli.

8.4. Considerații asupra relației dintre rugozitatea suprafeței și imaginea acesteia.

În [LEE1, JOSSO1, YANAGI1] se arată că interdependența dintre rugozitatea suprafeței și imaginea acesteia se bazează, în principiu, pe dependența funcțională dintre înălțimea rugozității și nivelul de gri al imaginii suprafeței, în

sensul că părțile mai înalte ale asperităților corespund pixelilor de intensitate mai mare. Studiul experimental al imaginilor achiziționate arată că relația dintre rugozitate și imagine este mai complexă fiind puternic dependentă de modul de iluminare. Iluminarea eșantionului se realizează cu surse de lumină externe, microscopul pentru rugozitate (cu mărimi mici de până la x100) neavând iluminare prin obiectiv. În figura 8.6. s-a reprezentat schematic iluminarea unui eșantion cu o singură sursă de lumină. Se observă că intensitatea iluminării este mai mare pe zona „A” decât pe zona „B”, delimitarea acestor zone nefiind în concordanță cu înălțimea rugozității. În acest caz este importantă poziția sursei de iluminare față de orientarea microneregularităților suprafeței. Pentru a elimina efectele poziției relative dintre eșantion și sursa de lumină este de preferat să se utilizeze surse de iluminare de tipul tuburilor fluorescente de formă toroidală ce asigură o iluminare uniformă a eșantionului.

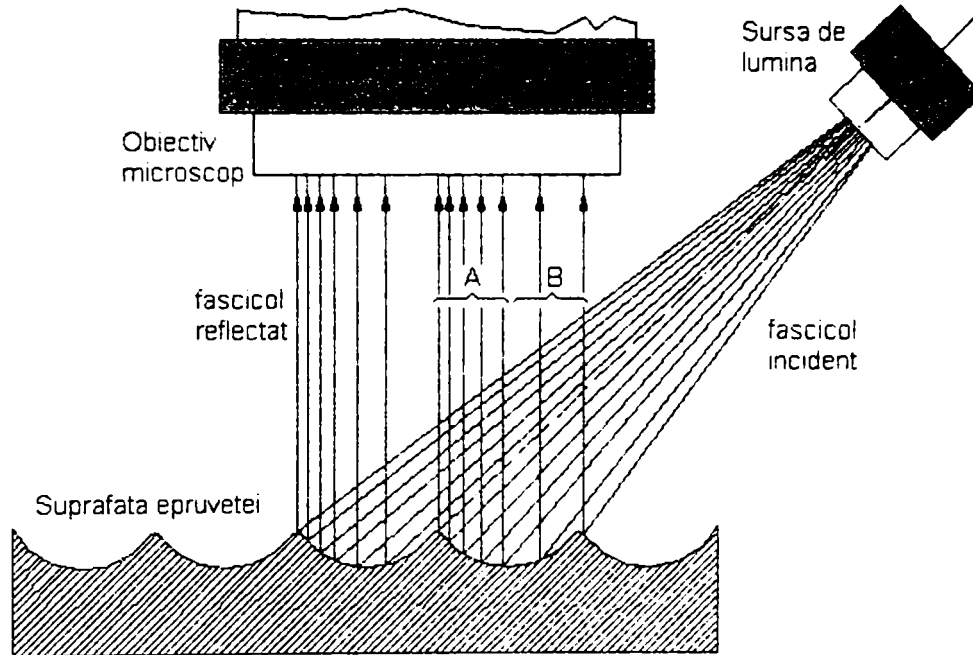


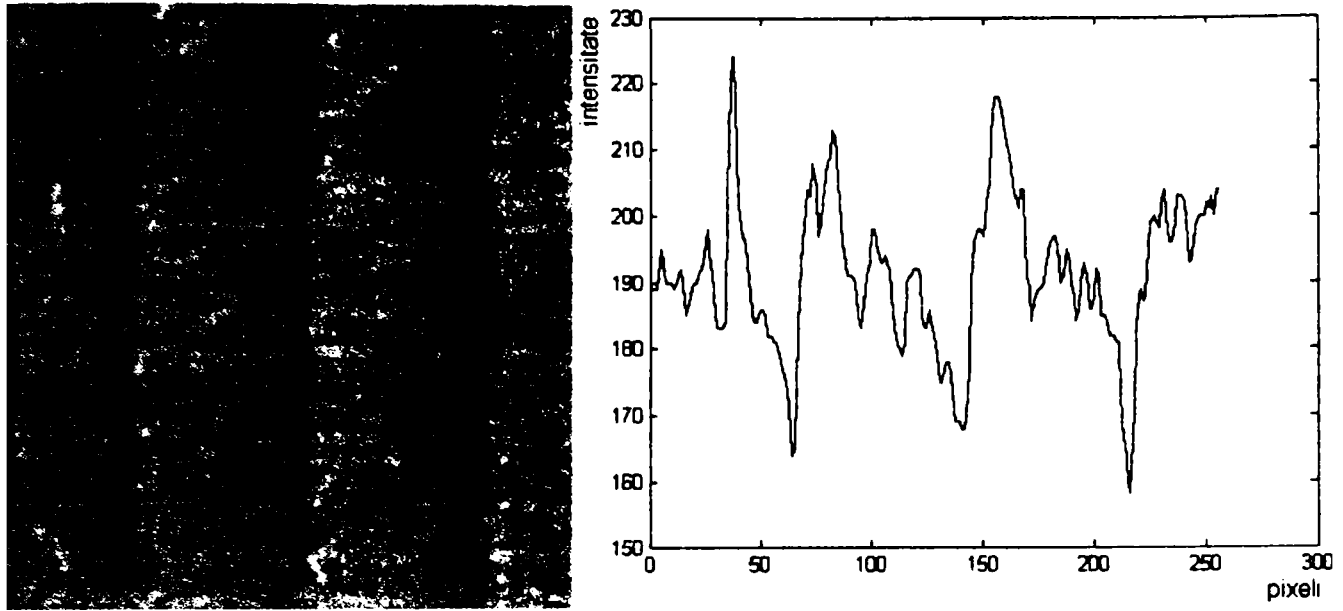
Fig.8.6. Schema de iluminare cu o singură sursă de lumină.

Imaginea achiziționată a unei suprafețe etalon este arătată în figura 8.7.a. iar diagrama intensității pe o linie a matricii imagine este arătată în 8.7.b. Pentru o ilustrare mai clară în figura 8.8. este redată matricea imagine în reprezentare 3D.

Din figura 8.7 (a și b) rezultă în mod clar fenomenul descris anterior : vârfurile înălțimilor nu corespund univoc zonelor de intensitate maximă din imagine. Relația dintre rugozitatea și imaginea suprafeței este deosebit de complexă și nu se propune tratarea amănunțită a acesteia. Se consideră că există o corelație bine definită: o anumită rugozitate la o anumită mărire va produce o imagine distinctă în raport cu imaginile altor suprafețe cu alte rugozități, la aceeași mărire. Deasemenea se consideră importantă realizarea unui algoritm ce asigură în prima fază (faza de cercetare) formarea grupelor G_{et} , iar în a doua (faza de lucru) încadrarea unui eșantion (pe baza C_{ep}) în una din grupele G_{et} . Algoritmul se concepe pe două etape distincte : etapa de determinare a procedurii tehnologice utilizat și etapa de determinare a rugozității, așa cum se arată în figura 8.9.

Conform acestei concepții imaginea unui eșantion va fi încadrată într-una din grupele de prelucrare iar după determinarea grupei de prelucrare va fi încadrată

una din grupele de rugozitate corespunzătoare acelei prelucrări.



a. b.
Fig.8.7. a. Imaginea achiziționată a suprafeței etalon fc5 (x10)
 b. Diagrama de intensitate pe o linie a matricii imagine.

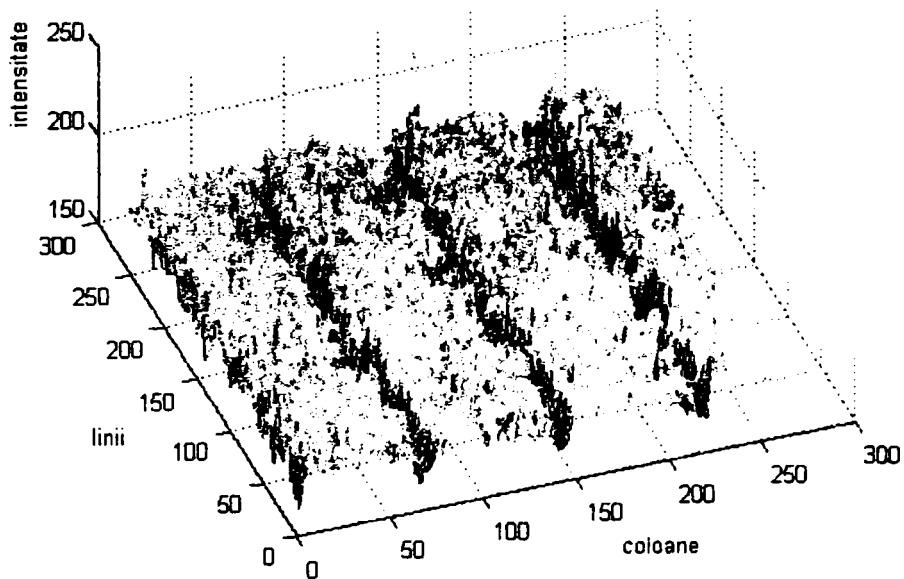


Fig.8.8. Reprezentarea 3D a matricii imagine pentru suprafața etalon fc5x10.

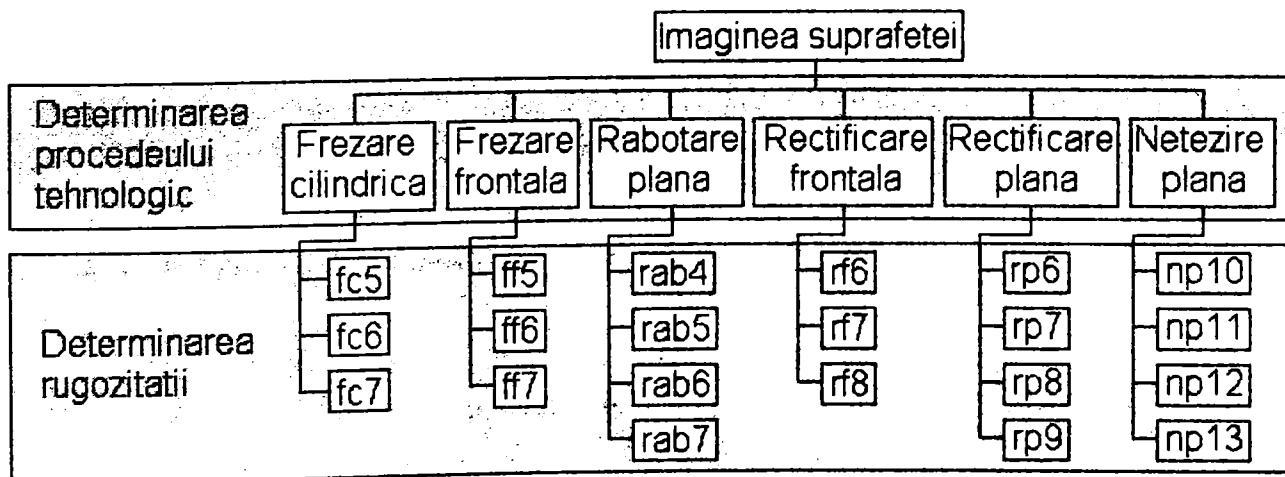


Fig.8.9. Schema partajării formării grupelor G_{et} .

8.5. Preprocesare.

După cum se prezintă în figura 8.3. preprocesarea imaginilor presupune la rândul ei parcurgerea anumitor faze, fiecare dintre acestea având un anumit scop :

- *filtrarea* se realizează pentru eliminarea zgomotelor din imaginea achiziționată;
- *delimitarea domeniului de interes* se realizează pentru eliminarea zonelor ce nu conțin informații utile și ar putea denatura rezultatele analizei;
- *eliminarea efectelor iluminării neuniforme* ce apare mai ales în cazul utilizării surselor de lumină separate de microscop (componente aleatoare ale iluminării), aceste efecte observându-se în figura 8.5.;
- *eliminarea efectelor situării suprafeței de analizat față de microscop* se presupune că axa optică a microscopului este perpendiculară pe suprafața de analizat dar aceasta poate fi rotită și traslatată în plan.

Programul de preprocesare realizat în limbajul MATLAB este prezentat în ANEXA 8.2. Se descriu în continuare etapele de realizare a preprocesării.

După achiziția imaginii, aceasta este disponibilă în memoria de lucru sub forma unei matrici (RGB) ce este transformată în format intensitate (capitolul 3.3.1). Pentru sesizarea efectului zgomotelor datorate achiziției, în figura 8.10. a și b se prezintă diagramele de intensitate după o linie a două imagini. Scopul filtrării este pregătirea delimitării zonei de interes. Se observă că pentru diagrama din figura 8.10.a. zona de interes este ușor separabilă existând un salt semnificativ între zona întunecoasă și zona de intensitate ridicată, dar pentru diagrama 8.10.b. această departajare este mai puțin evidentă.

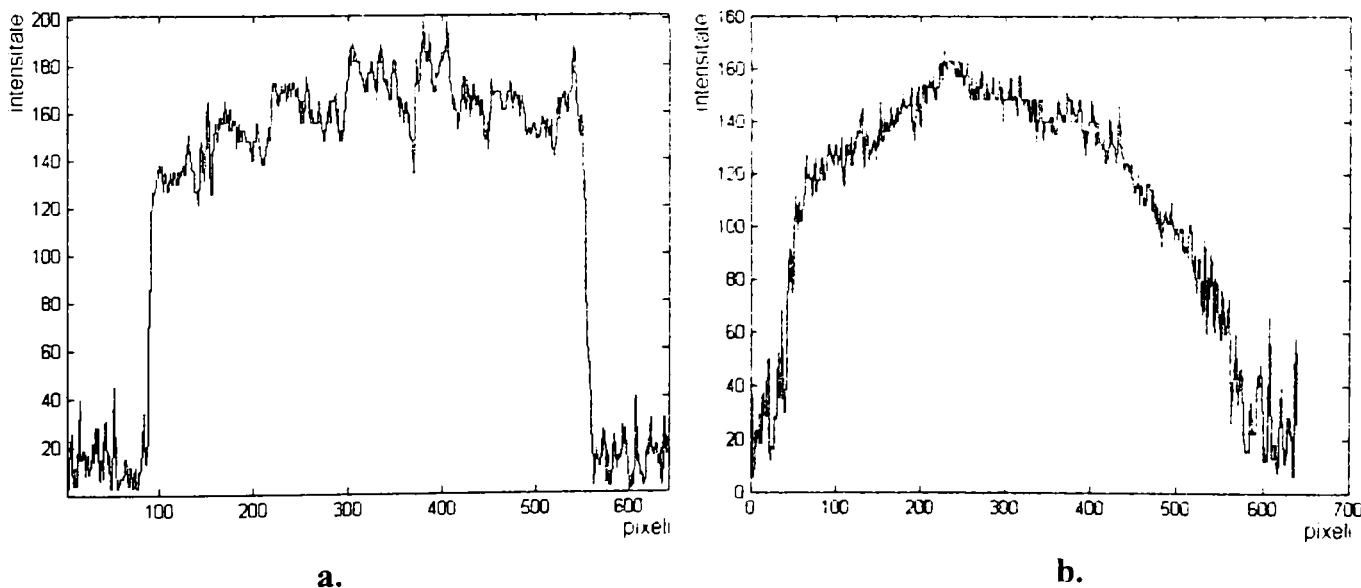


Fig.8.10. Diagrame de intensitate pe o linie pentru fc5 x10 (a) și pentru np12x10 (b).

La alegerea filtrului se are în vedere că zgomotele induse de camera CCD (black current) sunt de natură aleatoare. Pentru acest tip de zgomote în literatura de specialitate [SMITH] se recomandă utilizarea filtrelor Gauss bidimensionale (figura 8.11). S-a constatat experimental că cele mai bune rezultate se obțin cu filtrul Gaussian cu dimensiunea a 1/5 din imaginea originală. Filtrarea imaginii se

realizează prin convoluție spațială.

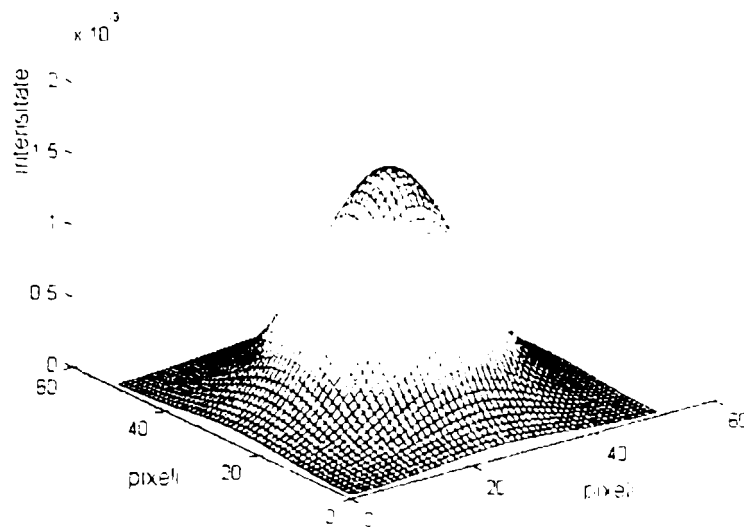


Fig.8.11. Filtru Gaussian bidimensional.

Filtrarea aplicată elimină zgomotele din imagine dar odată cu acestea sunt eliminate și unele detalii ale microneregularităților. Se reamintește că în acest stadiu se dorește selectarea zonei de interes. Din această cauză cu ajutorul filtrului se realizează o mască ce în urma binarizării asigură decelarea zonei de interes de fundalul imaginii. Pe imaginea filtrată este mult mai simplu să se aplice un algoritm de binarizare cu stabilirea automată a pragului de binarizare ca de exemplu metoda Otsu [OTSU]. Astfel din imaginea inițială (figura 8.5., $fc5 \times 10$) se obține masca reprezentată în figura 8.12.a. Se aplică apoi două prelucrări morfologice : prima de eliminarea a zonelor negre de mici dimensiuni ce are ca rezultat obținerea măștii reprezentate în figura 8.12.b. iar a doua de închidere morfologică ce are ca rezultat obținerea măștii finale (figura 8.12.c).

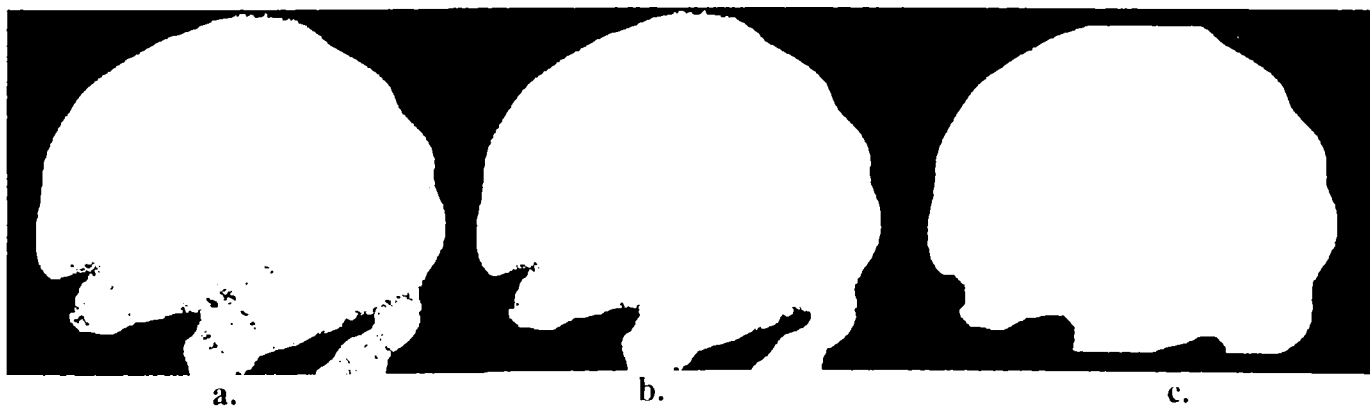


Fig.8.12. Masca obținută după filtrare și binarizare (a): după eliminarea zonelor negre de mici dimensiuni (b): după curățarea conturului (c).

Datorită faptului că analiza imaginii se poate realiza pe matrici dreptunghiulare, zona de interes va trebui aleasă în așa fel încât să aibă formă dreptunghiulară și în același timp să conțină maxim de informații posibile. Pentru aceasta s-a realizat un algoritm de selectare automată a dreptunghiului de arie maximă ce să conțină numai pixelii de valoare „1” (culoare albă) (figura 8.13.).

Pentru aceasta se parcurge imaginea pe coloane ($j = 1 \dots N$, unde N este

lățimea imaginii) căutându-se punctele de pe contur ce intersectează coloana

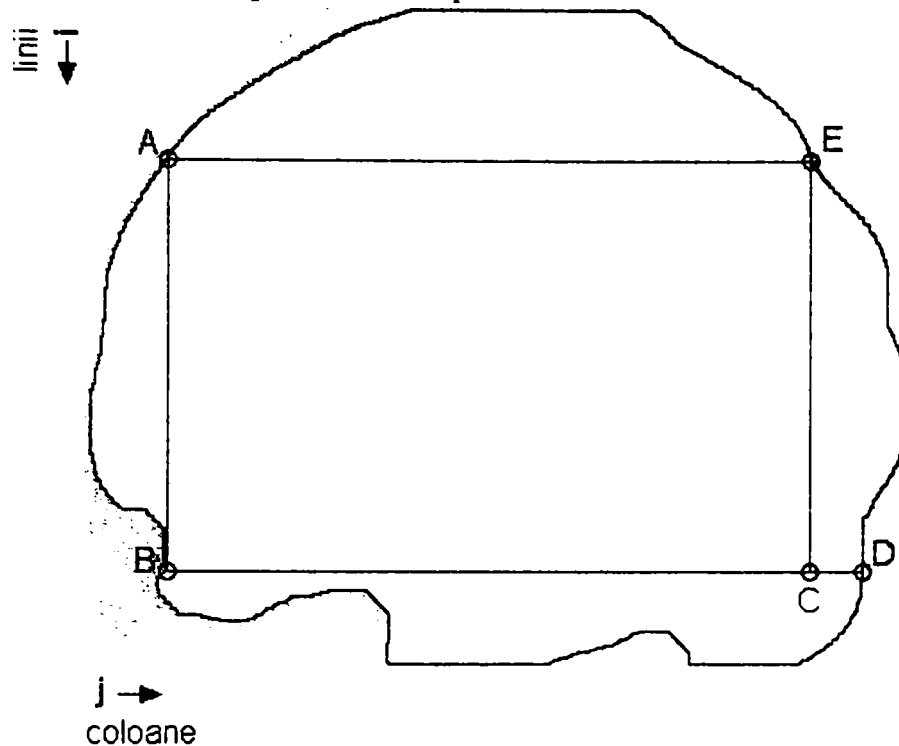


Fig.8.13. Selectarea zonei dreptunghiulare.

curentă ($A_{i,j}$ și $B_{i+h,j}$). Pornind de la aceste puncte se parcurg liniile corespunzătoare punctelor obținute anterior (i_A și i_B) și se găsesc alte două puncte de pe contur ($E_{i+h,j+m}$ și $D_{i+h,j+n}$). Se selectează lungimea minimă dintre cele două segmente (minimul dintre $\overline{AB} = m$ și $\overline{BD} = n$) și se găsește punctul $C_{i+h,j+m}$. Punctele A, B, C, E reprezintă colțurile dreptunghiului selectat. Se calculează aria corespunzătoare a dreptunghiului selectat și se continuă algoritmul pentru următoarele puncte inițiale ($A_{i,j+1}$ și $B_{i+h,j+1}$). În final se va selecta dreptunghiul de arie maximă.

Masca finală se aplică asupra imaginii inițiale reținându-se astfel doar zona de interes (zona ce conține informații utile). Imaginea astfel obținută este arătată în figura 8.14.a. iar reprezentarea 3D a matricii imagine în figura 8.14.b.

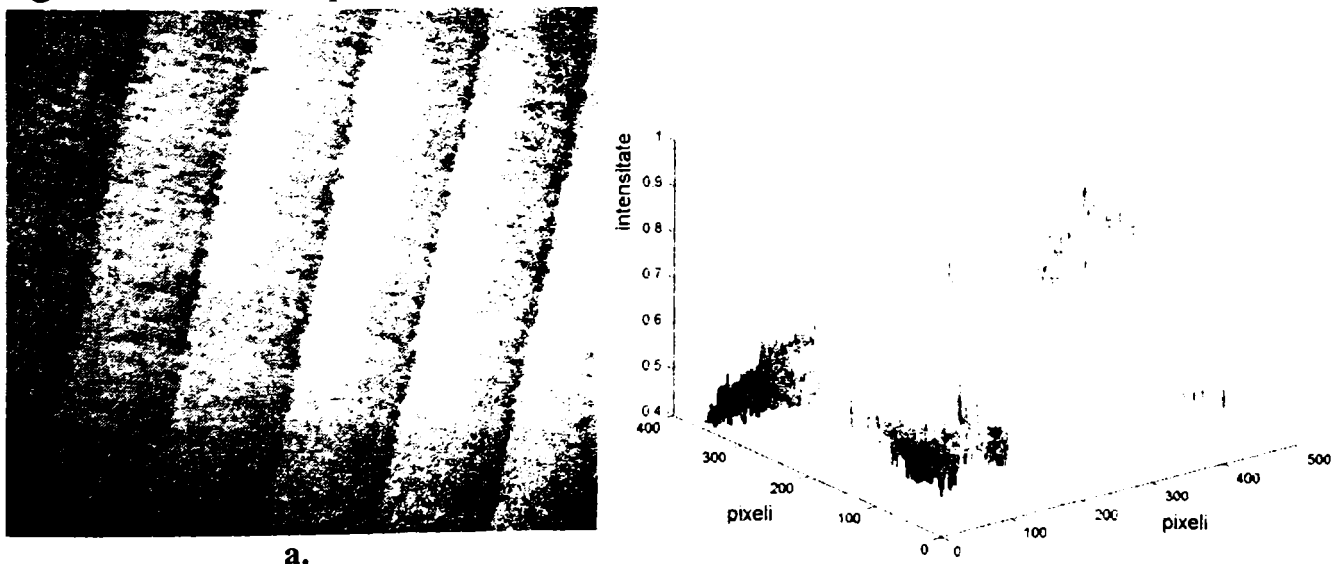


Fig.8.14. Imaginea suprafeței afectată de efectele iluminării neuniforme : reprezentare 2D (a) și 3D (b) (fc5x10).

În aceste figuri apare pregnant efectul iluminării neuniforme ce poate afecta în mod negativ, rezultatul analizei caracteristicilor.

Mărimea intensității este rezultanta compunerii iluminării cu reflectanța suprafeței, legea de compunere fiind neliniară. Ceea ce interesează în cazul analizei este componenta ce caracterizează suprafața (componenta de reflexie) și nu iluminarea. Eliminarea componentei de iluminare se realizează prin aplicarea unui filtru Gauss similar cu cel prezentat anterior dar de această dată elementele matricii imagine se împart la elementele matricii imagine filtrate. Se aplică deasemenea o corecție a marginilor imaginii rezultate ce este afectată de filtrare [SMITH].

În figura 8.15 este arătată reprezentarea 3D a imaginii filtrate ce este de fapt componenta de iluminare. Imaginea finală reprezintă componenta ce conține numai informația caracteristică suprafeței (figura 8.16.a. și b.).

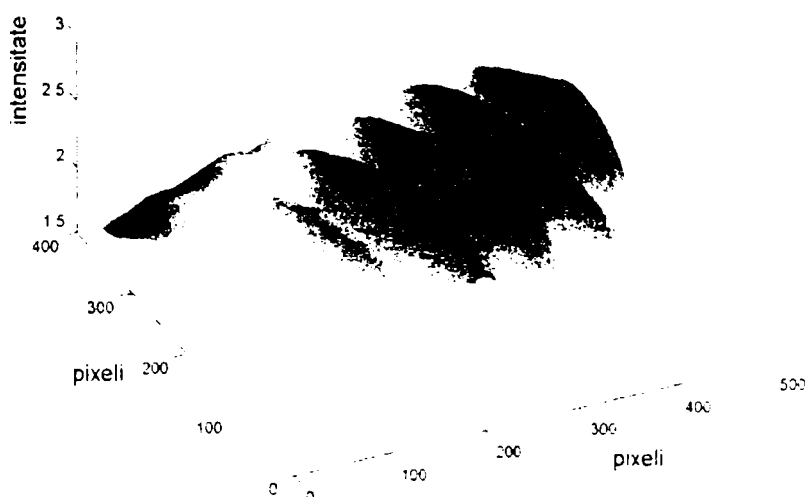


Fig.8.15. Reprezentarea 3D a componentei de iluminare (fc5x10).

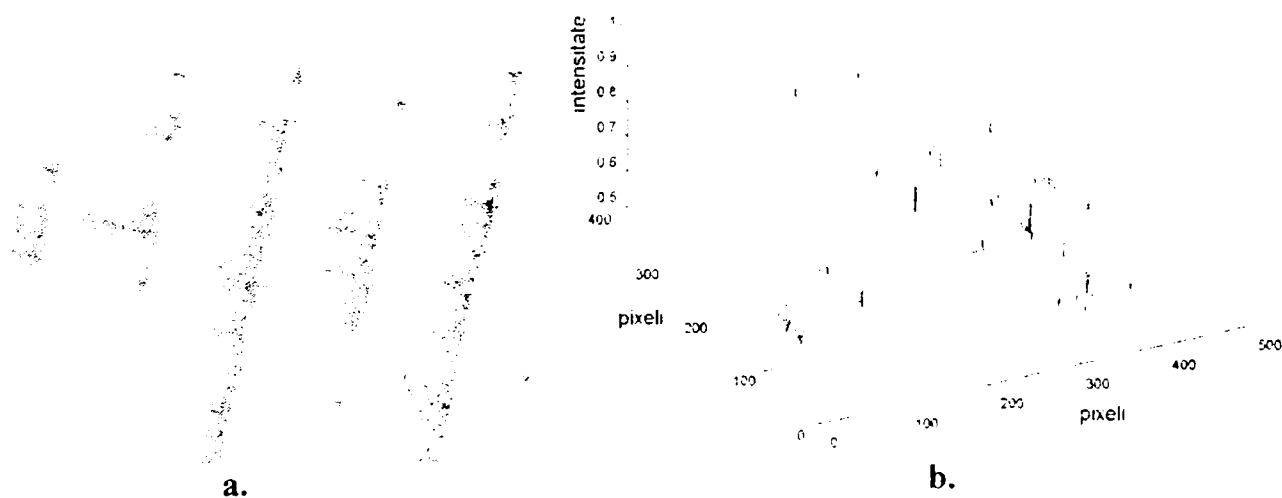


Fig.8.16. Reprezentarea 2D (a) și 3D (b) a componentei de reflectanță a imaginii (fc5x10).

Microneregularitățile suprafețelor obținute prin diverse procedee tehnologice sunt anizotropice ceea ce ridică o nouă problemă pentru algoritmul de analiză, deoarece în momentul achiziției imaginii unui eșantion, direcția de orientare a microneregularităților este necunoscută. În consecință se pot adopta două strategii :

- se realizează un algoritm invariant la rotația imaginii (IRI);
- se realizează un algoritm care poate fi afectat de rotația imaginii dar se realizează în cadrul preprocesării rotirea automată într-o poziție bine stabilită (non-IRI).

În cel de-al doilea caz rotirea imaginii se realizează ținând cont de direcția orientării microneregularităților. Se consideră poziție de zero a orientării poziția imaginii în care energia maximă a intensității se obține pentru direcția orizontală.

Rotirea în poziție se poate realiza prin mai multe metode, dintre care, s-a ales determinarea orientării prin valoarea maximă a DSP (densitatea spectrală de putere [SMITH]) calculată din media pe o direcție a valorilor de intensitate, această metodă prescurtându-se în continuare DSPM.

În cadrul acestei metode se rotește succesiv imaginea cu un anumit increment unghiular în intervalul 0° - 180° , pentru fiecare orientare calculându-se parametrul de orientare. Pentru fiecare pas de orientare se calculează media intensității pe coloane (utilizând notațiile ca în figura 8.13). Pentru coloana j se va calcula valoarea medie a intensității :

$$\mu_j = \frac{1}{N} \sum_{i=1}^N g_{i,j} \quad (8.15)$$

unde : $g_{i,j}$ este valoarea intensității în punctul i,j ; N este numărul de linii al imaginii de dimensiuni $N \times M$.

Pentru șirul valorilor medii în funcție de j , se calculează DSP utilizând FFT (descriș în capitolul 3) și se reține valoarea maximă pentru fiecare unghi de rotație (θ) :

$$DSP_{\max}(\theta) = \max(\text{FFT}(\mu(j, \theta))) \quad (8.16)$$

În ANEXA 8.3. (figurile A.8.3.1., A.8.3.2., A.8.3.3., A.8.3.4.) sunt date imaginile suprafețelor, diagramele mediei respectiv DSP pentru 4 poziții rotite cu un increment $\Delta\theta = 10^\circ$. Se obține astfel șirul valorilor $DSP_{\max}(\theta)$. Unghiul de rotație corespunzător orientării căutate va fi :

$$\theta_{\text{rot}} = \max(DSP_{\max}(\theta)) \quad (8.17)$$

În cadrul preprocesării imaginea va fi rotită cu unghiul θ_{rot} astfel încât orientarea inițială a imaginii nu va influența rezultatul algoritmului de calcul a parametrilor rugozității, indiferent dacă acest algoritm este sau nu invariant la orientare. Precizia de orientare realizată de algoritm (constatată experimental) a fost de $\pm 3^\circ$, în consecință algoritmi ce nu sunt invariante la rotație au fost testați și cu imagini rotite la 3° respectiv la -3° față de orientarea corectă.

Determinarea orientării imaginii se poate realiza și prin alte metode frecvențiale cum ar fi analiza FFT2, descrișă în [JIANG1] sau convoluția cu filtre Gabor orientate [TSAI1]. În urma încercărilor s-a stabilit că metoda ce utilizează

FFT2 este aproximativ la fel de rapidă ca și metoda descrisă anterior dar precizia de orientare este mai mică. Metoda ce utilizează convoluția cu filtre Gabor este mai precisă dar există o serie de parametri a căror determinare este dificilă și este mai lentă decât celelalte două metode.

După realizarea rotirii imaginii aceasta este gata pentru calculul parametrilor rugozității. În ANEXA 8.3., figurile A.8.3.5 – A.8.3. sunt arătate imaginile preprocesate utilizând algoritmi descriși în acest paragraf pentru etaloanele de rugozitate luate în considerare. Programul de preprocesare generează pentru fiecare dintre imaginile inițiale (de tipul celor arătate în figura 8.5), câte 4 imagini diferite, decupate din zona de interes dreptunghiulară (figura 8.13) cu o dimensiune de 256x256 pixeli, pentru ca în cadrul dezvoltării algoritmului de calcul a parametrilor rugozității să se poată testa și invarianța la translație a acestuia.

8.6. Algoritmi de analiză.

8.6.1. Considerații generale

Studiind cercetările recente în domeniul analizei texturilor și a prelucrării de imagini (capitolul 3) într-o primă fază s-au încercat mai multe metode ce fac parte din grupa metodelor statistice (co-ocurență, momente statistice de diferite ordine) și frecvențiale (filtre Gabor, wavelet) care însă, pentru setul de imagini etalon selectate au produs rezultate nesatisfăcătoare. În a doua fază s-au încercat metode fractale ce au produs rezultate mai bune, din care cauză s-a optat pentru utilizarea acestora pentru experimentări mai aprofundate. Utilizând metode fractale s-a reușit elaborarea unor algoritmi ce pot să realizeze discriminarea diverselor tipuri de imagini.

8.6.2. Metode fractale.

În domeniul analizei imaginilor fractalii oferă o alternativă la descrierea euclidiană a seturilor de obiecte sau a structurilor cuprinse în imagine. Noțiuni de bază privind definiția fractalilor cât și câteva dintre metodele fractale au fost descrise în capitolul 3.9. Calculul dimensiunii fractale absolute (în sens matematic) este mai puțin important din punctul de vedere al aplicațiilor practice. Din punct de vedere aplicativ este important să se realizeze algoritmi prin care, pe baza unor caracteristici fractale, să se poată realiza discriminarea imaginilor suprafețelor obținute prin diferite prelucrări și în cadrul acestora a diferitelor rugozități.

Dimensiunea fractală în sensul pur matematic (D) poate fi folosită pentru testarea algoritmilor pe imagini artificiale la care se compară valoarea D obținută prin calcul pe baza formulei (8.20), cu valoarea obținută cu ajutorul algoritmului.

Dacă $Z(x)$ este înălțimea neregularităților în funcție de lungimea profilului x (figura 8.17), relația dintre $Z(x)$ și dimensiunea fractală se definește prin funcția Weirstrass-Mandelbrot (W-M), [JIANG1]:

$$Z(x) = A^{(D-1)} \sum_{n=n_0}^{\infty} \frac{\cos 2\pi\gamma^n \cdot x}{\gamma^{(2-D)n}} \quad (8.18)$$

unde

$Z(x)$ – înălțimea neregularităților în funcție de lungimea de măsurare;

D – dimensiunea fractală a profilului;

A – constantă de scalare;

γ^n – modurile de frecvență corespunzătoare lungimii de undă a rugozității,
 $\gamma^n = 1/\lambda^n$, în [XIONG] se adoptă $\gamma = 1.5$;

n_0 – parametru corespunzător frecvențelor joase de tăiere pentru profilul măsurat, fiind practic o limitare a frecvențelor pentru separarea rugozităților de undulații.

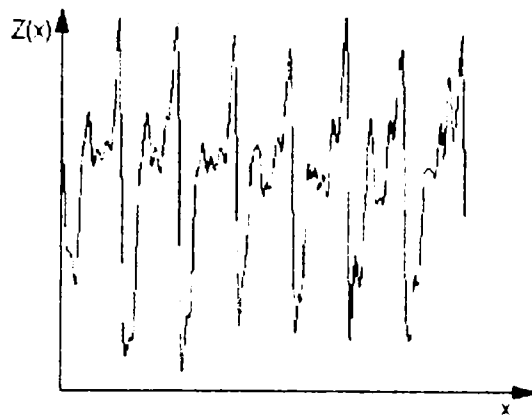


Fig.8.17. Diagrama înălțimii neregularităților în funcție de lungimea de măsurare (etalon ff5x10).

Când funcția W-M este corelată cu densitatea spectrală de putere a semnalului $Z(x)$, dimensiunea D este corelată cu panta drepte de aproximare a spectrului în coordonate dublu logaritmice:

$$\text{DSP}(\omega) = \frac{A^{2(D-1)}}{2 \ln \gamma} \cdot \frac{1}{\omega^{(5-2D)}} \quad (8.19)$$

unde ω este pulsația spațială ($\omega = 2\pi f$, f – frecvența spațială).

Desigur în cazul imaginilor $Z(x)$ va reprezenta intensitate de culoare și nu înălțimea neregularităților.

Funcția W-M este dificil de utilizat în practică. De aceea se vor utiliza metode a căror implementare în cadrul algoritmilor numerici este mai ușoară. Relația (8.19) subliniază legătura dintre DSP și dimensiunea fractală ceea ce permite elaborarea unor algoritmi de calcul ai parametrilor fractali bazați pe DSP.

În cadrul tezei de doctorat s-au elaborat mai mulți algoritmi de calcul a unor parametrii fractali (figura 8.18.), capabili să caracterizeze rugozitatea suprafeței.

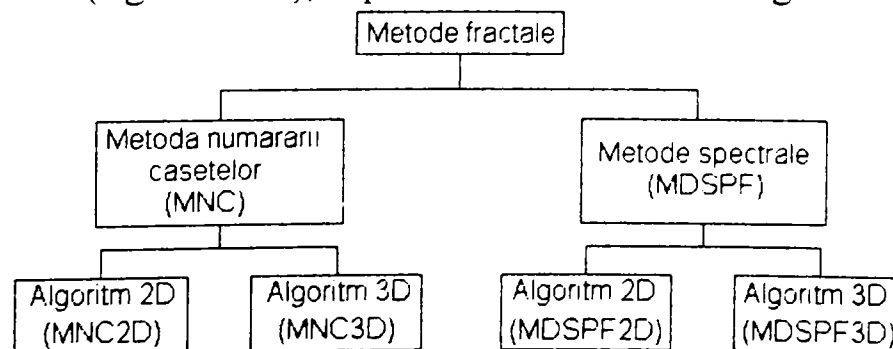


Fig.8.18. Schema relației dintre algoritmii utilizați.

MNC și MDSPF sunt metode de calcul a unor parametrii fractali a imaginilor digitale a suprafețelor etaloanelor de rugozitate. Ambele metode s-au aplicat în mai multe variante pe imaginile etalon, studiindu-se apoi parametrii fractali rezultați. Dacă prin $I(i,j)$ cu $i = 1..N, j=1..M$, se notează intensitatea funcție de poziție în matricea imagine (pentru o imagine de dimensiuni $N \times M$), prin „Algoritm 2D” se înțeleg acei algoritmi ce analizează variația intensității $I(i,j)$ când i este fixat și j variază de la 1 la M , iar prin „Algoritm 3D” se înțeleg acei algoritmi ce analizează variația intensității $I(i,j)$ când i variază de la 1 la N și j variază de la 1 la M . S-a studiat deasemenea și aplicarea succesivă pentru mai multe valori ale lui i a unui „Algoritm 2D”, dar acest caz nu este identic cu un „Algoritm 3D”.

Metoda numărării casetelor

Această metodă este strâns legată de calculul dimensiunii fractale așa cum a fost ea definită de Mandelbrot, cu ajutorul relației :

$$D = \frac{\log N}{\log(1/r)} \quad (8.20)$$

Relația (8.20) este aplicabilă în cazul algoritmilor 2D (în contextul în care sunt analizate curbe asemănătoare celei arătate în figura 8.17.), dintre care este descrisă metoda „dimensiunii compas” (compas dimension [COSTA]).

Se selectează o mulțime ordonată descrescătoare de măsuri ($r_i, i=1..k$) și se reține numărul de măsuri N_i ce acoperă curba pentru fiecare măsură r_i în parte. Se trasează apoi graficul în scară dublu logaritmică a numărului de măsuri funcție de inversul valorii măsurii obținând diagrama Richardson (figura 8.19).

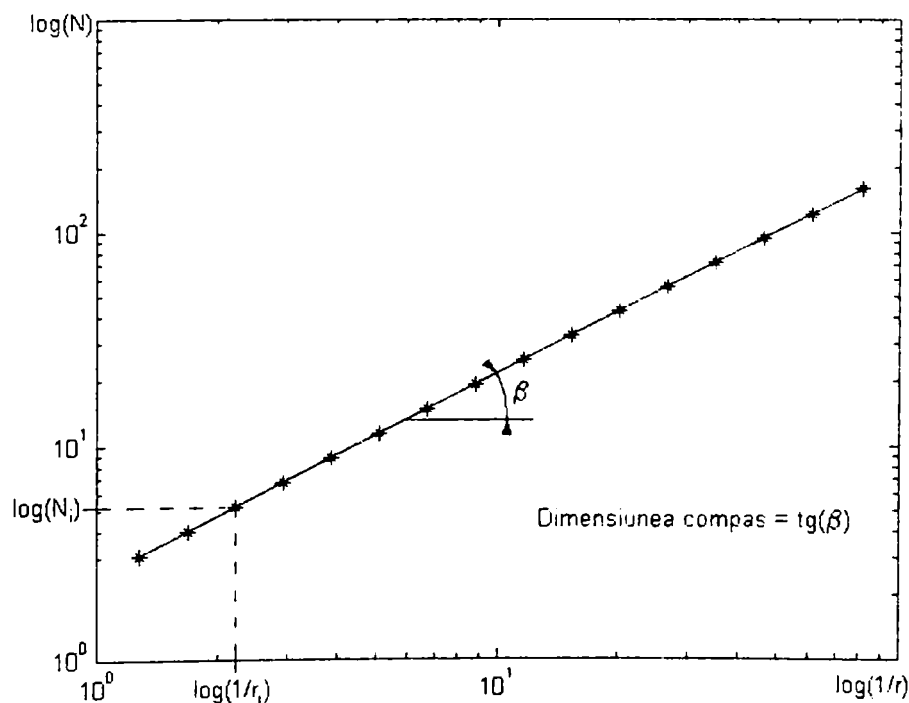


Fig. 8.19. Diagrama Richardson, după [COSTA].

Dacă setul de valori obținute este aranjat după o dreaptă atunci se consideră că structura analizată este fractală. Caracteristica fractală este denumită

„dimensiunea compas” și reprezintă de fapt panta dreptei obținute.

Metoda dimensiunii compas se realizează dificil în practică deoarece curbele experimentale sunt discrete și trebuie realizate interpolări între două puncte de pe curbă pentru a stabili poziția coardei de măsură.

Metoda numărării casetelor (MNC) simplifică algoritmul prin suprapunerea peste curba analizată a unor casete (zonă de formă pătrată) în locul unor segmente de măsură.

Pentru testarea metodei s-a realizat programul „dim_fractal”, în limbaj MATLAB, prezentat în ANEXA 8.4. Schema logică a programului este dată în figura 8.20. Programul a fost realizat după un cod similar dat în [OLIVER].

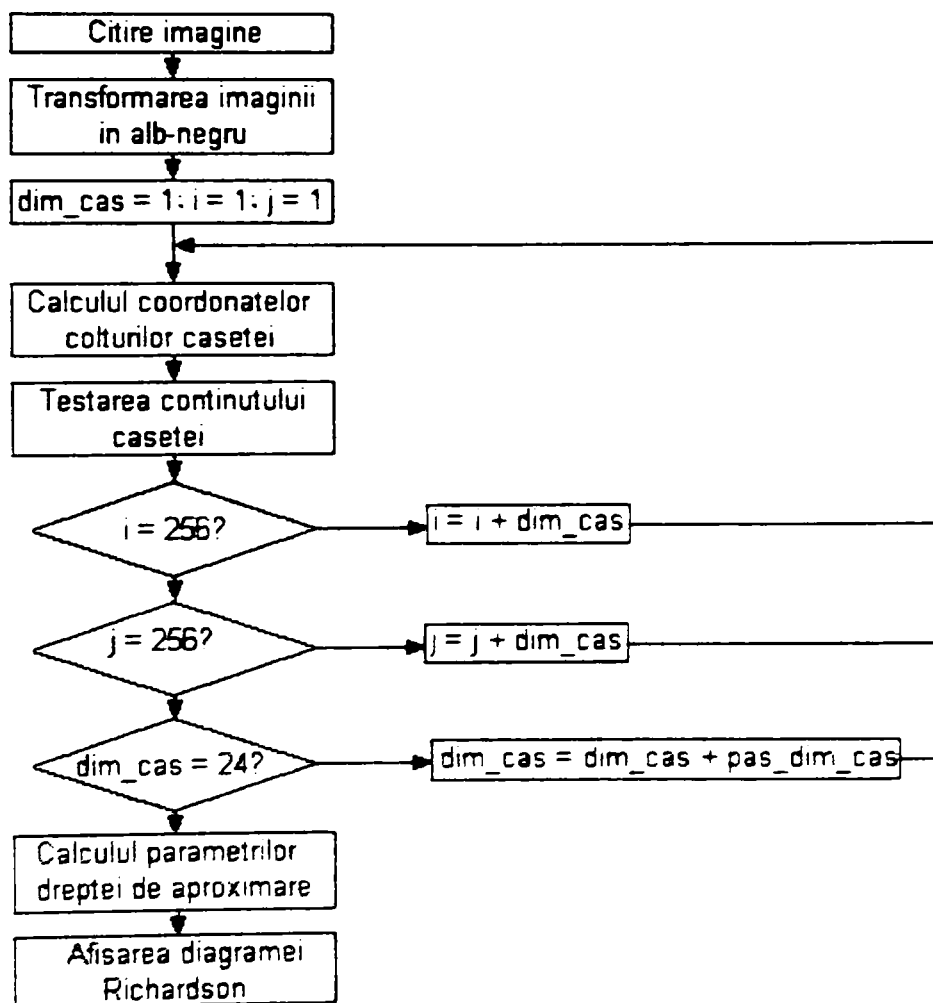


Fig.8.20. Schema logică a programului de numărare a casetelor.

Notățiile utilizate în figura 8.20, sunt:

dim_cas – dimensiunea casetei în pixeli;

i – coordonata liniei matricii imagine;

j – coordonata coloanei matricii imagine;

pas_dim_cas – pasul de modificare a dimensiunii casetei;

Prin citirea imaginii se realizează transformarea imaginii cu dimensiunea de 256x256 pixeli din fișierul tip *BMP* într-o matrice în zona de memorie gestionată de MATLAB. Matricea rezultată are dimensiunea 256x256x3 corespunzătoare formatului *RGB*. Prin transformarea imaginii în alb-negru (binarizare) se va obține o matrice de dimensiunea 256x256 în care fiecare element are valoarea 0 sau 1.

Algoritmul de calcul al parametrului fractal p_1 (dimensiunea compas) se realizează prin testarea existenței unuia sau mai multor pixeli cu valoarea 1 în interiorul casetei pătrate cu latura dim_cas , ce parcurge matricea pe linii i și pe coloane j , (figura 8.21).

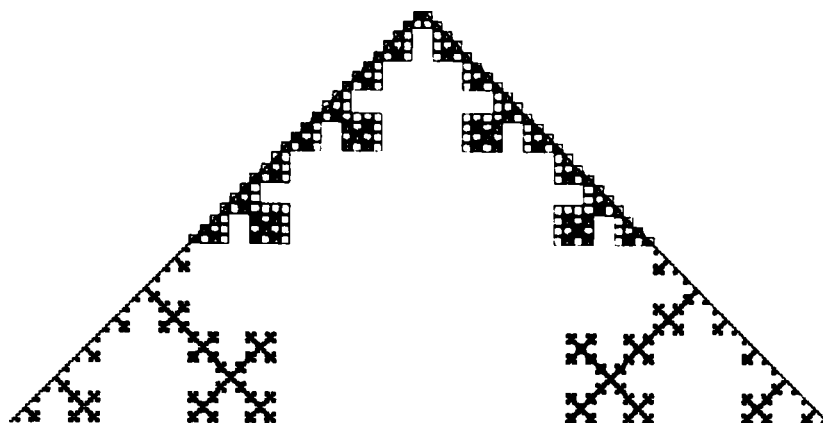


Fig.8.21. Acoperirea curbei lui von Koch cu casete de dimensiune 6 pixeli.

La fiecare parcurgere a matricii se reține numărul casetelor ce au conținut cel puțin un pixel cu valoarea 1 ($cont_pix_on$) și dimensiunea casetei (dim_cas). Se iterează apoi dimensiunea casetei cu un anumit pas (pas_dim_cas) și se repetă parcurgerea matricii. Se obține în final șirul valorilor $cont_pix_on$ și dim_cas ce se reprezintă în coordonate dublu logaritmice și se aproximează cu o dreaptă (prin metoda celor mai mici pătrate) de ecuație :

$$\log(cont_pix_on) = p_1 \log(dim_cas) + p_2 \quad (8.21)$$

în care p_1 reprezintă panta dreptei de aproximare iar p_2 ordonata la origine.

Algoritmul prezentat a fost testat pe doi fractali bine cunoscuți: curba lui von Koch și triunghiul lui Sierpinsky. Pentru ambele cazuri s-au desenat faze succesive ale construirii fractalilor (ANEXA 8.5). Dacă algoritmul funcționează corect, pe măsură ce figura este mai complexă, valoarea caracteristicii fractale calculate cu ajutorul algoritmului ar trebui să tindă către valoarea dimensiunii fractale calculate în mod direct (cu ajutorul formulei 8.20).

Pentru curba lui von Koch s-au construit 4 faze arătate în figurile A.8.5.1 – A.8.5.4. Dimensiunea fractală se calculează în mod direct cu relația :

$$D = \frac{\log(5)}{\log(4)} = 1.161 \quad (8.22)$$

la fiecare fază a construcției, fiecare segment este împărțit în 5 segmente iar lungimea fiecărui segment este o pătrime din lungimea anterioară.

Valoarea calculată de către programul „dim_fract” este $D_{ex}=1,216082$ iar eroarea $D - Dex = 0,055082$ (aproximativ 4,6%).

În figura 8.22. se arată evoluția valorii dimensiunii fractale calculate pentru diferitele faze ale construcției curbei lui von Koch. Se observă că pe măsură ce complexitatea curbei crește D_{ex} se apropie de D . Construcția figurii a fost limitată de către rezoluția grafică și dimensiunile imaginii folosite, dar se consideră că eroarea de calcul a programului este acceptabilă.

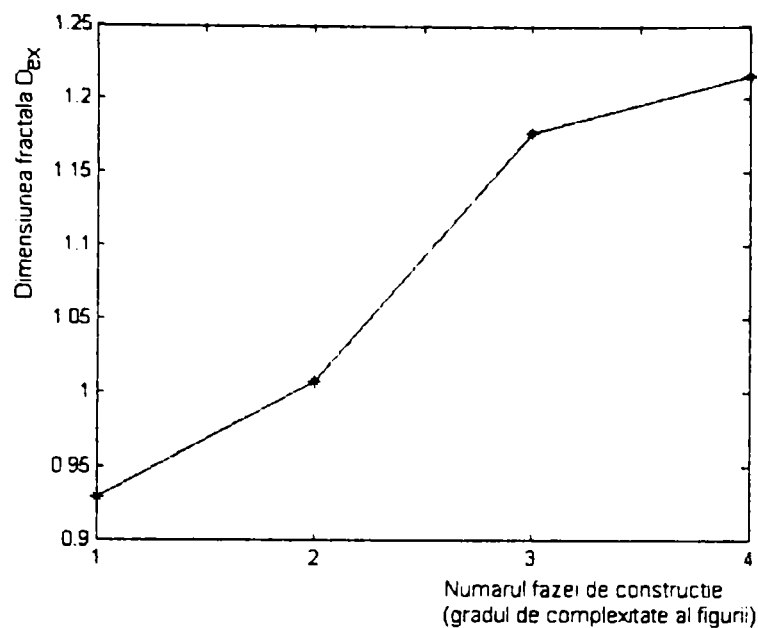


Fig.8.22. Diagrama dimensiunii fractale în funcție de faza de construcție pentru curba lui von Koch.

Programul a fost rulat și pentru 6 faze ale construcției triunghiului lui Sierpinsky (figurile A8.5.5.–A8.5.10, ANEXA 8.5). Dimensiunea fractală calculată în acest caz este :

$$D = \frac{\log(4)}{\log(3)} = 1.2619 \quad (8.23)$$

la fiecare fază a construcției, fiecare triunghi a fost împărțit în patru triunghiuri cu aria de o treime din cel de la pasul anterior. În acest caz valoarea calculată de programul „dim_fract” este $D_{ex} = 1,320178$, iar eroarea $D - D_{ex} = 0.058278$, (aproximativ 4,5%). În figura 8.23. se prezintă evoluția valorilor dimensiunii fractale în funcție de faza de construcție. Și în acest caz se observă că odată cu creșterea complexității figurii analizate D_{ex} tinde către D .

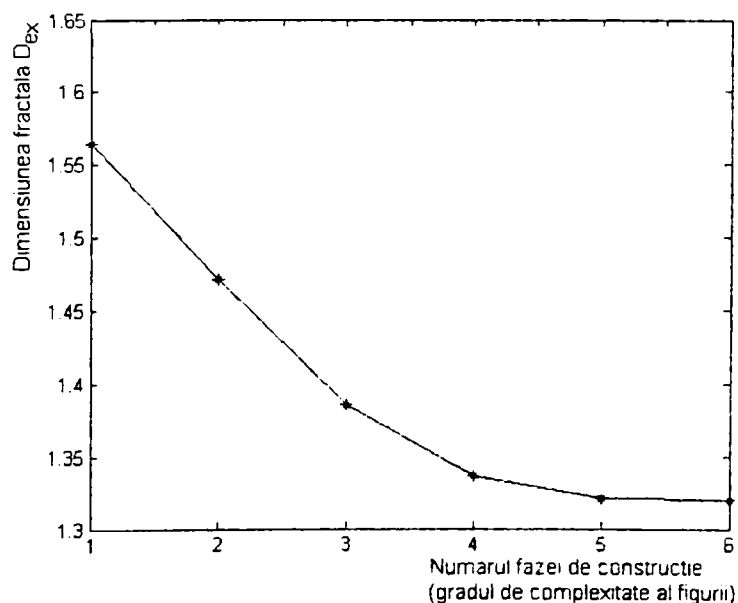


Fig.8.23. Diagrama dimensiunii fractale în funcție de faza de construcție pentru triunghiul lui Sierpinsky.

În acest caz se constată o evoluție mai bună a dimensiunii fractale decât în cazul curbei lui von Koch.

Pentru imagini pe nivele de intensitate metoda numărării casetelor nu se poate aplica în mod direct. Dacă se consideră imaginea ca o suprafață reprezentată în 3D, se pot stabili plane de nivel ce secționează suprafața. Curbele rezultate se pot apoi analiza utilizând numărarea casetelor. Mandelbrot a sugerat că dimensiunea fractală a suprafeței va fi egală cu dimensiunea fractală a curbei de nivel plus 1. Acest fapt a fost demonstrat experimental doar pentru anumite suprafețe izotropice. Unii cercetători propun acoperirea suprafeței cu seturi de triunghiuri, utilizate ca măsură și trasarea diagramei Richardson pe această bază. În loc de triunghiuri se pot utiliza acoperiri sferice ale suprafeței (Minkowsky Comforter). Diferențe de înălțimi la diferite spațieri se utilizează la trasarea diagramei Hurst [COSTA], ce poate reprezenta deasemenea o caracterizare fractală a suprafeței. Russ a modificat metoda Hurst (care inițial este valabilă pentru suprafețe izotrope) în „Transformata de orientare Hurst” pentru suprafețe anizotrope.

Metode spectrale

Dimensiunea fractală poate fi calculată și pe baza curbei densității spectrale de putere (DSP), legătura dintre DSP și dimensiunea fractală fiind definită de relația (8.19). În [COSTA] este dată o relație mai simplă ce arată că pentru majoritatea suprafețelor prelucrate prin diverse procedee tehnologice, amplitudinea DSP este invers proporțională cu frecvența ridicată la o anumită putere:

$$DSP(f) = \frac{1}{f^{p_1}} \quad (8.24)$$

Relația 8.24. sugerează și modul în care se poate calcula parametrul fractal p_1' prin trasarea diagramei DSP în funcție de frecvență în scară dublu logaritmică și aproximarea punctelor diagramei cu o dreaptă a cărei pantă (p_1') este corelată cu dimensiunea fractală prin relația :

$$D = \frac{(5 - p_1')}{2} \quad (8.25)$$

Deși în [COSTA] se arată că metoda spectrală este mai puțin precisă decât metoda numărării casetelor, cea din urmă are avantajul că este mai rapidă deoarece pentru calculul DSP se poate utiliza FFT.

Pentru ilustrarea modului de calcul a parametrului p_1' se prezintă în figura 8.24. diagrama DSP în funcție de frecvența spațială, obținută prin aplicarea FFT pe valorile de intensitate prezentate în figura 8.7.b. ce reprezintă o linie a matricii imagine pentru etalonul fc5x10. Calculând apoi $\log(DSP)$ și reprezentându-le în funcție de $\log(f)$ (logaritmul frecvenței spațiale), se obțin punctele din diagrama din figura 8.25. Se calculează apoi, utilizând metoda celor mai mici pătrate, dreapta de aproximare dată de relația:

$$\log(DSP) = p_1' \log(f) + p_2' \quad (8.26)$$

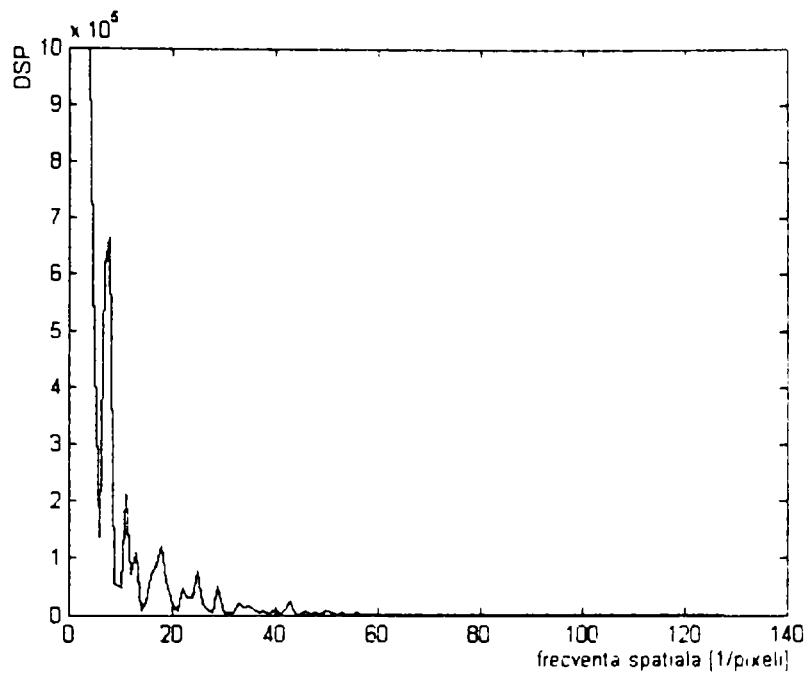


Fig.8.24. DSP pentru diagrama din figura 8.7.b.

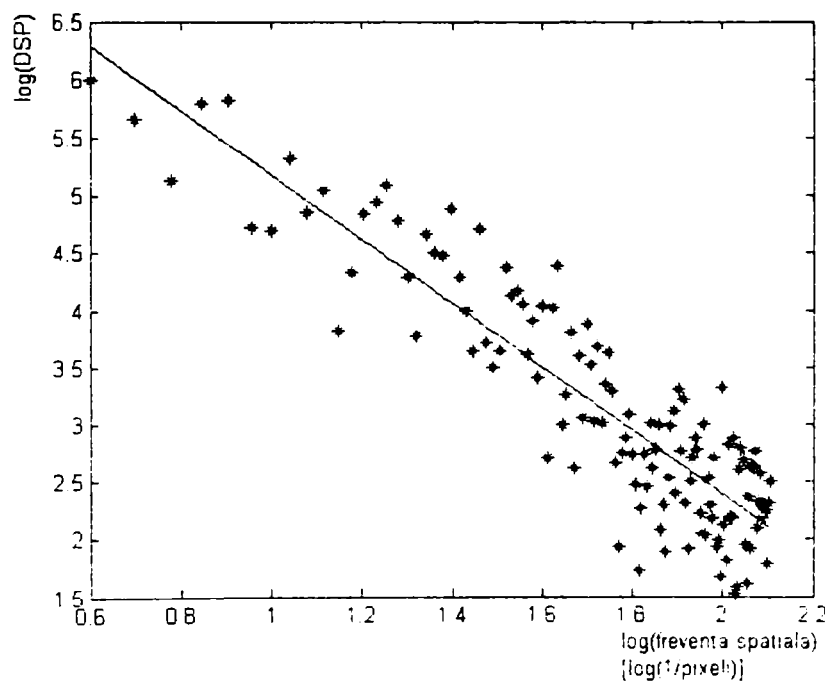


Fig.8.25. Diagrama $\log(\text{DSP})$ în funcție de $\log(f)$, și dreapta de aproximare, (dispersia $\sigma = 0.4539$).

8.6.3. Descrierea algoritmilor realizați și rezultate experimentale

8.6.3.1. Tipuri de algoritmi

În cadrul tezei de doctorat s-au elaborat programe ce utilizează metoda numărării casetelor (MNC) și metoda spectrală (MDSPF) pentru calculul parametrilor fractali ai imaginilor suprafeței etaloanelor de rugozitate. Scopul acestor algoritmi este realizarea unei discriminări între imaginile etaloanelor studiate, ce face posibilă stabilirea grupelor G_{et} . Metodele ce s-au experimentat sunt arătate în figura 8.26. Nu s-a utilizat metoda spectrală 3D deoarece această

metodă se pretează mai bine pentru analiza imaginilor suprafețelor izotrope.

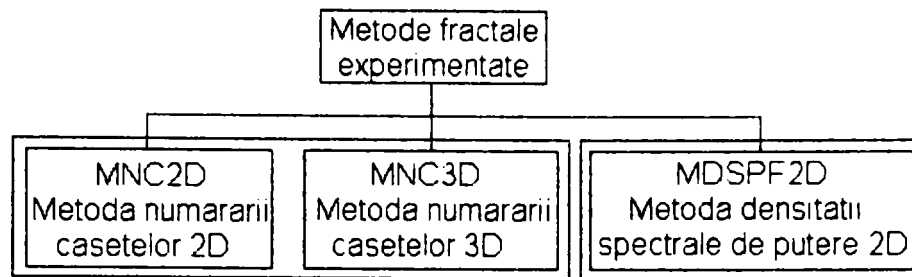


Fig.8.26. Metode fractale utilizate în cadrul experimentelor.

Înainte de descrierea algoritmilor și prezentarea rezultatelor obținute se impune o scurtă analiză a imaginilor etaloanelor utilizate (ANEXA 8.3 figurile A.8.3.5.- 8.3.16), prin specificarea următoarelor observații :

- la etaloanele obținute prin frezare cilindrică (figura A.8.3.5.), frezare frontală (figura A.8.3.6.), și rabotare rab4 și rab5 (figura A.8.3.7.a și b) se observă pregnant caracterul periodic al structurilor.
- pentru etaloanele de rabotare rab6 și rab7 (figura A.8.3.7.c și d), caracterul periodic este mai puțin semnificativ acestea fiind mai curând similare cu etaloanele de rectificare plană sau netezire decât cu celelalte (rab 4 și rab5) de rabotare.
- etalonul de rectificare frontală rf6 (figura A.8.3.8.a) este deasemenea mai apropiat din punctul de vedere al aspectului vizual de rectificarea plană decât de celelalte două etaloane din grupa rectificării frontale rf7 și rf8 (figura A.8.3.8.b și c). Aceasta se datorează faptului că rf6 a fost obținut cu trecerea pe suprafață, a pietrei oală de rectificare, numai până la centrul acesteia.
- etaloanele de rectificare plană și netezire plană nu prezintă o periodicitate clară, microneregularitățile având un aspect aleator.

Pe baza acestor observații se estimează că realizarea unor grupe G_{et} pe două nivele (primul nivel pentru prelucrări și al doilea nivel pentru rugozități), așa cum se arată în figura 8.9., este dificilă și va trebui realizată formarea unor grupe mixte.

8.6.3.2. Metoda numărării casetelor 2D.

Programul „MNC2D” calculează parametrii p_1 și p_2 ai drepte de aproximare aferente diagramei Richardson, pentru toate etaloanele de rugozitate considerate. Codul sursă al programului este dat în ANEXA 8.6.

S-a specificat anterior că algoritmul de numărarea a casetelor funcționează doar pentru imagini binare, iar pentru imaginile de intensitate vor trebui efectuate prelucrări suplimentare. După cum s-a arătat imaginile de intensitate sunt reprezentate în memorie sub forma unor matrici imagine (I_m) (de dimensiune 256x256 pixeli), fiecare element al matricii având o valoare de intensitate de la 1 la 256. O linie a matricii imagine poate fi privită ca un vector a căror elemente sunt, în mod similar, valori de intensitate ($val(I_m)$) în același domeniu. Pentru a utiliza algoritmul numărării casetelor, se transformă fiecare linie a matricii imagine (I_m)

într-o imagine binară (I_b). Pentru linia i a matricii I_m în prima fază se declară matricea (I_{bi}) cu elemente nule iar apoi se setează:

$$I_{bi}(j,k) = 1; k = \text{val}(I_m(i,j)) \quad (8.27)$$

Datorită faptului că valorile de intensitate sunt valori discrete, între două valori consecutive j și $j+1$, ordonata k va apare un gol de lungime m pixeli (figura 8.27).

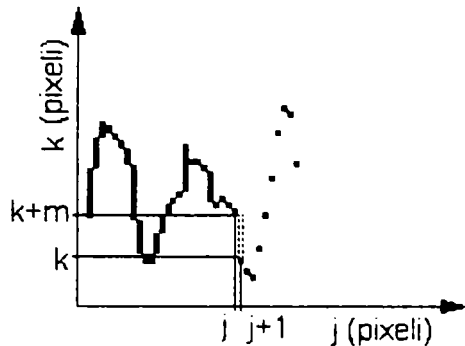


Fig.8.27. Schema completării imaginii binare, segmentul $k...k+m$ pe abscisa $j+1$.

Algoritmul completează aceste valori nule ajungându-se de la forma prezentată în diagrama din figura 8.28.a la forma prezentată în figura 8.28.b. Pentru o mai bună documentare în reprezentările grafice s-a utilizat complementarea matricilor binare.

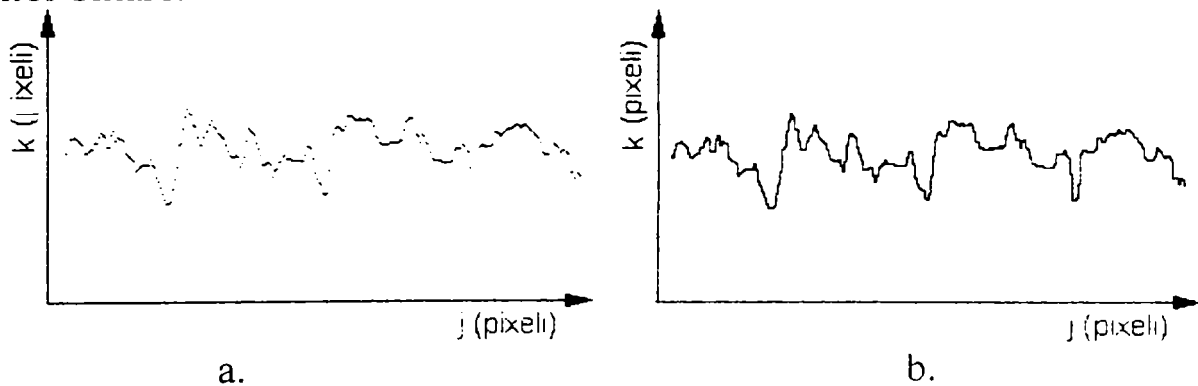


Fig.8.28. Imagine I_{bi} obținută cu relația (8.28.) (a) și completată (b).

Matricea I_{bi} va fi reprezentarea binară a vectorului $I_m(i,j)$ cu i fixat și $j = 1...256$. Pe matricea I_{bi} se aplică acum algoritmul de numărare a casetelor similar cu cel prezentat în programul „dim_cas”. În figura 8.29. sunt redată 3 faze de numărare cu casete de diferite dimensiuni.

Pentru fiecare matrice binară I_{bi} se obțin parametrii fractali p_{1i} (panta dreptei de aproximare) și p_{2i} (ordonata la origine a dreptei de aproximare). Iterând dimensiunea i a matricii I_m se obține setul parametrilor fractali, în funcție de care se calculează apoi parametrii fractali globali ai imaginii :

$$p_{1m} = \frac{1}{256} \sum_{i=1}^{256} p_{1i} \quad (8.28)$$

$$p_{2m} = \frac{1}{256} \sum_{i=1}^{256} p_{2i} \quad (8.29)$$

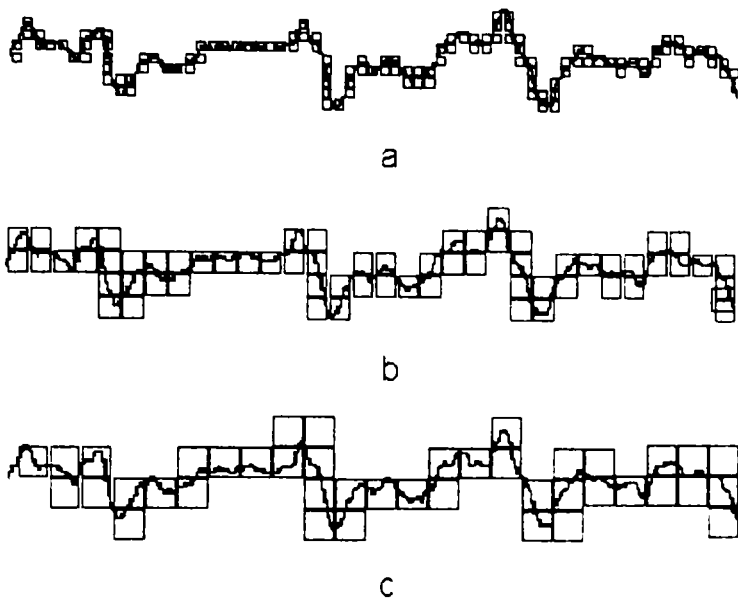


Fig.8.29. Diferite țaze ale numărării casetelor cu dimensiunea de : a. 7 pixeli, b. 11 pixeli, c. 15 pixeli.

În ce privește domeniul de variație a dimensiunii casetelor s-au realizat încercări pe diferite valori pornind de la un domeniu de 1...24 pixeli. Rezultatele încercărilor se arată în diagramele din ANEXA 8.7 (figurile A.8.7.1. –A.8.7.6.). Se observă că dispersia erorilor față de dreapta de aproximare nu scade odată cu mărirea domeniului de variație a dimensiunii casetelor utilizate, din care cauză s-a decis în continuare utilizarea unui domeniu de 1...24 pixeli. Din aceste diagrame se mai observă că punctele calculate de algoritm nu se așează după o dreaptă ceea ce poate fi un indiciu al multi-fractalității sau pseudo-fractalității imaginilor testate. Oricum importanța acestui fapt este pur teoretică și nu afectează în mod esențial rezultatele obținute.

Programul „MNC2D” rulează algoritmul descris pentru toate etaloanele de rugozitate afișând caracteristica fractală a fiecăreia reprezentată în planul p_1-p_2 , prin simbolul aferent, conform simbolizării date în tabelul 2 ANEXA 8.1. Această diagramă se va numi în continuare „diagramă P_f ” și va fi utilizată și pentru algoritmul „MNC3D”. Diagramele P_f se realizează separat pentru imagini cu mărimi de $\times 10$ și mărimi de $\times 25$. Pentru micșorarea duratei de rulare a programului în condiții de lucru, s-a încercat calculul caracteristicilor fractale pentru fiecare a n -a linie a imaginii astfel încât în loc de 256 de imagini I_{hi} se vor analiza doar $256/n$ linii, considerându-se că diferența dintre liniile apropiate ale imaginii nu este foarte mare. În ANEXA 8.7. figurile A.8.7.7. – A.8.7.10. sunt arătate diagrame P_f pentru diferite spațieri $n = 2,4,,8,16$. Diferențele constatate nu sunt semnificative, decât în cazul $n = 16$, arătând că această metodă poate da rezultate în micșorarea duratei de analiză a imaginii. Totuși în cadrul experimentelor s-a utilizat $n = 2$ pentru a avea certitudinea unei analize corecte.

Diagramele P_f sunt realizate în continuare pentru mai multe seturi de imagini etalon (3 sau 4 imagini ale aceluiași etalon dar din zone diferite ale suprafeței) verificându-se în acest fel robustețea algoritmului.

În figura 8.30. este prezentată diagrama P_f pentru trei seturi de etaloane cu mărime de $\times 10$, iar în figura 8.31. cu mărime de $\times 25$.

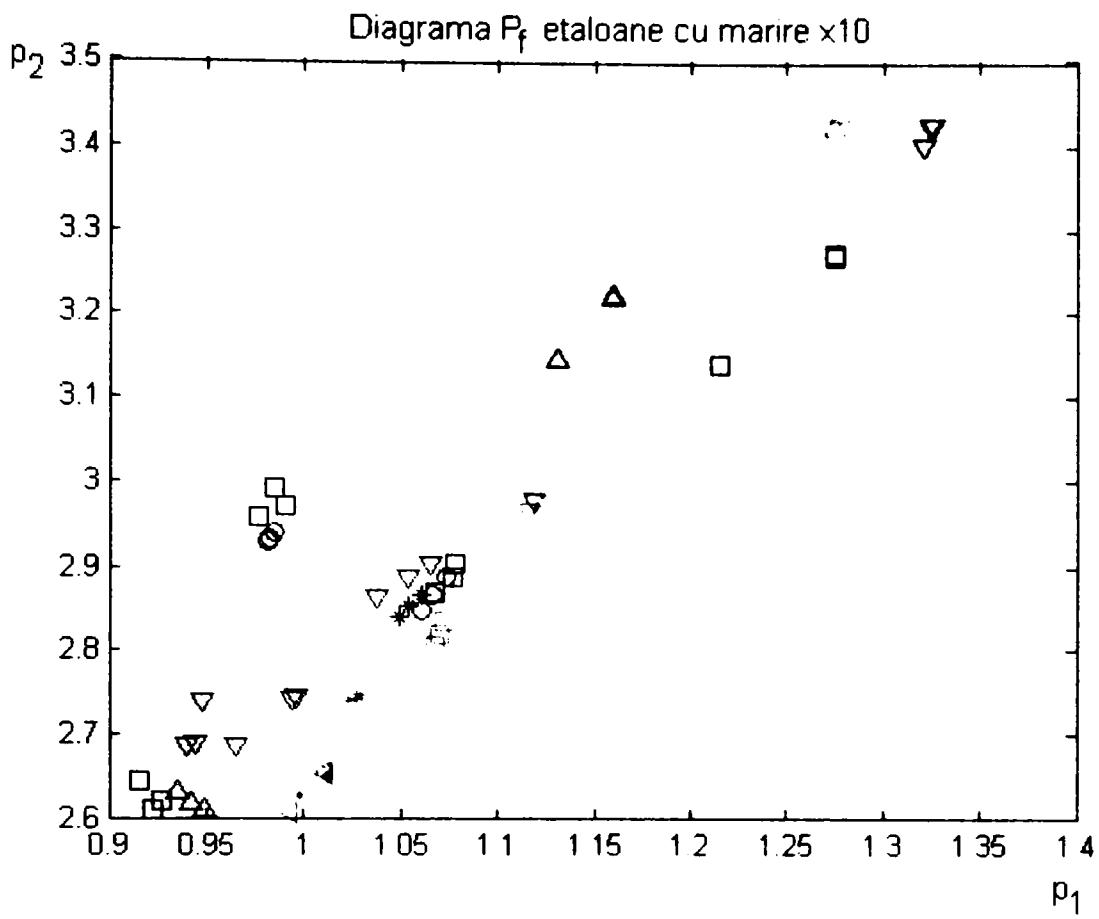


Fig.8.30. Diagrama P_f pentru etaloane cu mărire de $\times 10$.

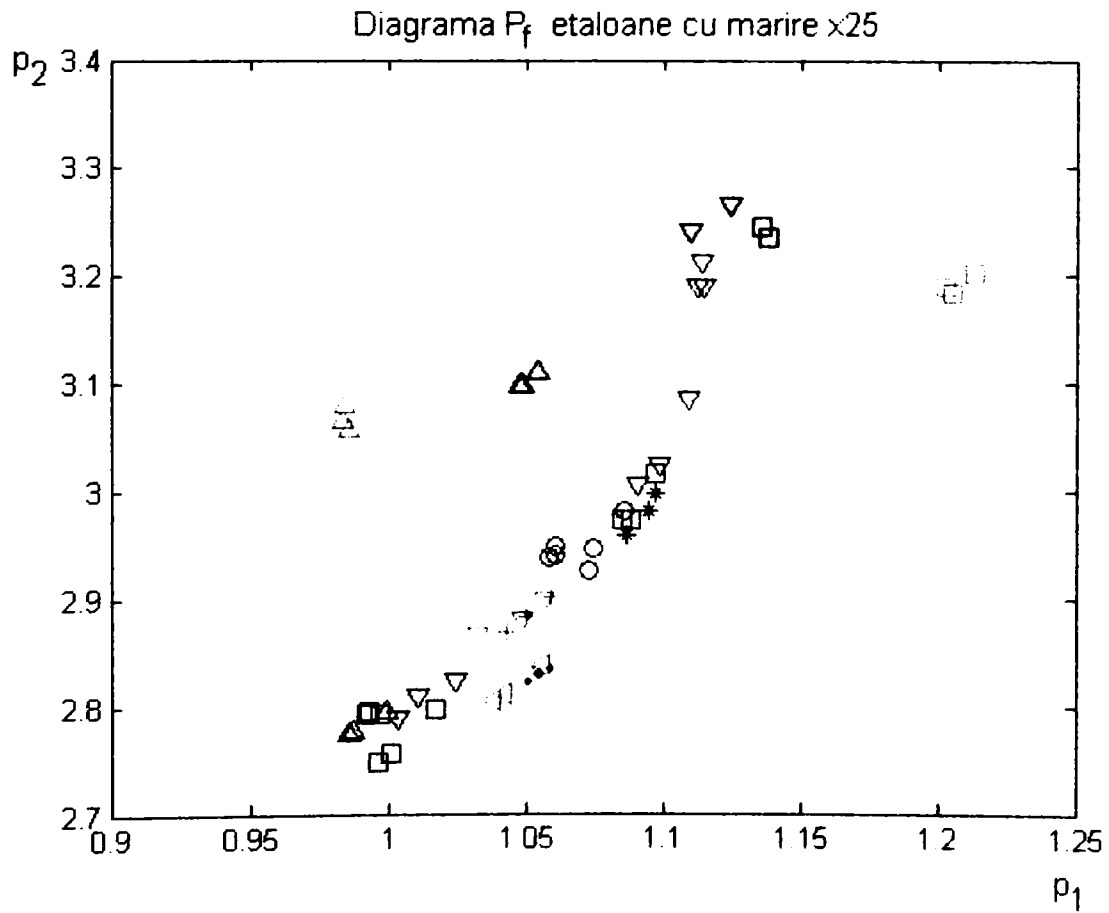


Fig.8.31. Diagrama P_f pentru etaloane cu mărire de $\times 10$.

Pe aceste diagrame se urmărește modul în care se grupează diferitele etaloane : cu cât sunt mai apropiate punctele ce reprezintă același etalon (aceeași prelucrare și aceeași rugozitate) cu atât este mai reușită discriminarea etaloanelor.

Din analiza diagramelor P_f prezentate se constată:

- a. în general gruparea punctelor pe diagramă respectă observațiile cu privire la aspectul vizual al imaginilor aferente:
 - rab4 și rab5 sunt relativ grupate dar mai depărtate de rab6 și rab7 care sunt mai apropiate de grupa prelucrărilor prin rectificare plană (rp6-rp9);
 - deasemene rf6 este mai depărtată de celelalte două rectificări frontale și mai apropiată de rectificările plane și netezire pe diagrama cu mărire x10 iar pe cea cu mărire x25, rf6 este mai aproape de frezările frontale;
- b. etaloanele ce prezintă o periodicitate clară nu se grupează bine în funcție de procedeul tehnologic ci mai mult în funcție de rugozități (ff, rab), iar etaloanele ce au un caracter aleator sunt bine grupate în funcție de procedeu dar nu se pot decela ca rugozități (fc, rp, np). Trebuie observat aici că frezările cilindrice prezintă o anumită periodicitate a structurilor vizuale dar aceasta nu este sesizată de către algoritm datorită lungimii mari de undă sau a redundanței mici (număr mic de perioade conținute în imagine);
- c. din punctul de vedere al măririi se poate observa că unele etaloane au valori ale parametrilor fractali asemănătoare atât la mărire x10 cât și la mărire x25 (fc, rp, np) iar alte etaloane, la mărire x25 au valori ale parametrilor fractali mult diferite de cele constatate la mărire x10 (ff, rab, rf), acest aspect fiind în concordanță cu natura aleatoare a structurilor vizuale; tot în acest sens se mai constată că etaloanele ce au caracteristici diferite la diferite mărimi își modifică în mod esențial poziția în raport cu celelalte grupe;
- d. la ambele mărimi se observă tendința alinierii punctelor după o dreaptă ceea ce arată că parametrii p_1 și p_2 utilizați sunt dependenți într-o anumită măsură; acest lucru se datorește faptului că p_1 este panta iar p_2 este ordonata la origine a dreptei de aproximare din diagrama Richardson.

Din constatările arătate la punctul c. se poate trage concluzia că diferențele de grupare între măririle x10 și x25 se pot utiliza pentru discriminarea etaloanelor.

Pentru eliminarea corelației dintre cei doi parametri, discutată la punctul d., în loc de parametrul p_2 se poate utiliza parametrul p_3 definit ca:

$$p_3 = \frac{p_2}{p_1} \quad (8.30)$$

Diagramele P_f (pentru ambele mărimi) stau la baza alcătuirii unui algoritm euristic pe baza căruia să se poată decide apartenența unui eșantion cu rugozitate necunoscută la o anumită grupă (G_{el}) corelată cu un anumit domeniu al planelor p_1 - p_2 stabilite atât pentru x10 cât și pentru x25.

În tabelul 8.1. s-au sintetizat domeniile de grupare ale diferitelor etaloane.

Tabel 8.1.

Cod etalon	Grupa etalon G_{et}	Domeniu $P_f \times 10$		Domeniu $P_f \times 25$		Rugozitate R_a
		$P_{1 \times 10}$	$P_{2 \times 10}$	$P_{1 \times 25}$	$P_{2 \times 25}$	
fc5	G1	0,9-0,96	2,6-2,8	0,96-1,025	2,7-2,92	6,3
fc6		0,9-0,96	2,6-2,8	0,96-1,025	2,7-2,92	3,2
fc7		0,9-0,96	2,6-2,8	0,96-1,025	2,7-2,92	1,6
ff5	G2	1,1-1,2	3,05-3,35	1,03-1,175	2,92-3,15	6,3
ff6	G3	1,2-1,3	3,05-3,35	1,03-1,175	3,15-3,4	3,2
ff7	G4	1,3-1,4	3,35-3,5	1,03-1,175	3,15-3,4	1,6
rab4	G5	1,2-1,25	3,35-3,5	0,9-0,96	2,92-3,15	12,5
rab5	G6	1,25-1,3	3,35-3,5	0,96-1,025	2,92-3,15	6,3
rab6	G7	1,1-1,2	2,6-3,5	1,03-1,175	2,7-2,92	3,2
rab7	G8	1,035-1,1	2,6-3,5	1,175-1,25	3,15-3,4	1,6
rf6	G9	0,96-1,035	2,6-2,8	1,03-1,175	3,15-3,4	3,2
rf7	G10	0,9-1,035	2,8-3,5	0,96-1,025	2,7-2,92	1,6
rf8	G11	0,9-1,035	2,8-3,5	1,03-1,175	2,92-3,15	0,8
rp6	G12	1,035-1,1	2,6-3,5	1,03-1,175	2,92-3,15	3,2
rp7		1,035-1,1	2,6-3,5	1,03-1,175	2,92-3,15	1,6
rp8		1,035-1,1	2,6-3,5	1,03-1,175	2,92-3,15	0,8
rp9		1,035-1,1	2,6-3,5	1,03-1,175	2,92-3,15	0,4
np10	G13	0,95-1,035	2,6-2,8	1,03-1,175	2,7-2,92	0,2
np11		0,95-1,035	2,6-2,8	1,03-1,175	2,7-2,92	0,1
np12		0,95-1,035	2,6-2,8	1,03-1,175	2,7-2,92	0,05
np13		0,95-1,035	2,6-2,8	1,03-1,175	2,7-2,92	0,025

Din valorile domeniilor date în tabelul 8.1. se constată că grupele G1, G12, G13 nu corespund scopului de discriminare a rugozităților. Din această cauză s-au realizat experimentări și cu alte metode (MNC3D și MDSPF2D).

O observație importantă este aceea că pe baza tabelului 8.1 se poate dezvolta un algoritm fuzzy (în mod similar cu cele prezentate în capitolul 5.6.3.2.), care să realizeze decizia de stabilire a rugozității. În cadrul acestui algoritm mulțimea criteriilor este:

$$C = \{ P_{1 \times 10}, P_{2 \times 10}, P_{1 \times 25}, P_{2 \times 25} \} \quad (8.31)$$

Pentru criteriile stabilite domeniile de valori sunt următoarele:

$$\begin{aligned} P_{1 \times 10} : D_{P_{1 \times 10}} &= [0,9 - 1,3] \\ P_{2 \times 10} : D_{P_{2 \times 10}} &= [2,6 - 3,5] \\ P_{1 \times 25} : D_{P_{1 \times 25}} &= [0,9 - 1,25] \\ P_{2 \times 25} : D_{P_{2 \times 25}} &= [2,7 - 3,4] \end{aligned} \quad (8.32)$$

Variabilele lingvistice sunt $p_{1 \times 10}$, $p_{2 \times 10}$, $p_{1 \times 25}$ și $p_{2 \times 25}$ iar gradele lingvistice se stabilesc după domeniul de variație al variabilelor lingvistice pentru fiecare grupă

în parte (conform tabel 8.1.). Astfel de exemplu gradul lingvistic corespunzător grupei $G2$ al variabilei lingvistice $p_{1 \times 10}$ va fi $p_{1 \times 10 G2}$. Mărimea de ieșire utilizată este rugozitatea (R) corespunzătoare grupei aferente de exemplu pentru $G2$: $R_{aG2}=6,3$.

Regulile de inferență se stabilesc pe baza celor descrise anterior, de exemplu, dacă valorile de intrare pentru un eșantion analizat sunt $p_{1 \times 10 ep}$, $p_{2 \times 10 ep}$, $p_{1 \times 25 ep}$ și $p_{2 \times 25 ep}$, prima regulă de inferență va fi:

$$\text{Dacă } (p_{1 \times 10 ep} \text{ este } p_{1 \times 10 G1}) \text{ și } (p_{2 \times 10 ep} \text{ este } p_{2 \times 10 G1}) \text{ și} \\ (p_{1 \times 25 ep} \text{ este } p_{1 \times 25 G1}) \text{ și } (p_{2 \times 25 ep} \text{ este } p_{2 \times 25 G1}) \text{ atunci } R \text{ este } R_{aG1} \quad (8.33)$$

Acest tip de algoritm fuzzy poate fi utilizat pentru oricare din metodele fractale experimentate.

8.6.3.3. Metoda numărării casetelor 3D.

Algoritmul MNC3D este o extindere în trei dimensiuni a algoritmului prezentat anterior.

În cadrul acestui algoritm în loc de casete ce reprezintă domenii de formă pătrată a imaginii se vor utiliza casete de formă cubică. În locul matricilor I_{bi} se va utiliza un singur masiv (matrice tridimensională) M_b . Inițial, pentru o matrice imagine de dimensiuni 256x256 pixeli și 256 de nivele de intensitate, se declară un masiv de dimensiuni 256x256x256 cu toate elementele nule. Apoi masivul este încărcat cu valori de 1 pe baza informațiilor de intensitate a matricii imagine I_m cu ajutorul relației :

$$M_b(i, j, k) = 1; k = \text{val}(I_m(i, j)) \quad (8.34)$$

scanând I_m pentru indicii $i = 1 \dots 256$; $j = 1 \dots 256$.

Masivul astfel obținut (ANEXA 8.8 figurile A.8.8.1.–A.8.8.3.), este acoperit cu casete cubice de diferite dimensiuni iar metoda numărării casetelor se aplică în mod similar cu cel utilizat în cadrul algoritmului anterior. Codul sursă al programului MNC3D este prezentat în ANEXA 8.9, iar în ANEXA 8.8. figura 8.8.4. se prezintă diagrama Richardson pentru etalonul $fc5 \times 10$ obținută cu ajutorul acestui program.

Diagramele P_f obținute cu algoritmul MNC3D sunt prezentate în figura 8.32. pentru mărire de $\times 10$ și figura 8.33 pentru mărire de $\times 25$. În aceste diagrame s-a utilizat pe ordonată parametrul p_3 definit prin relația (8.30.).

Și în acest caz se poate realiza un tabel cu valorile concrete ale domeniilor de discriminare pentru stabilirea grupelor G_{et} . Pentru simplitatea expunerii domeniile s-au delimitat în cadrul diagramelor.

Din corespondența unui etalon la un anumit domeniu pe diagrama cu mărire de $\times 10$ și la un alt domeniu pe diagrama cu mărire de $\times 25$ se poate realiza discriminarea etaloanelor (caracter multi-fractal).

În tabelul 8.2. sunt date relațiile de apartenență a diferitelor etaloane la domeniile stabilite pe diagramele P_f .

Tabel 8.2.

Cod etalon	Grupa etalon G_{et}	Domeniu $P_r \times 10$	Domeniu $P_r \times 25$	Rugozitate R_a
fc5	G1	D2	D2	6,3
fc6	G2	D2	D3	3,2
fc7		D2	D3	1,6
ff5	G3	D5	D4	6,3
ff6	G4	D9	D5	3,2
ff7	G5	D8	D8	1,6
rab4	G6	D3	D1	12,5
rab5	G7	D6	D2	6,3
rab6	G8	D5	D7	3,2
rab7	G9	D7	D9	1,6
rf6	G10	D2	D5	3,2
rf7	G11	D1	D6	1,6
rf8	G12	D1	D7	0,8
rp6	G13	D4	D7	3,2
rp7		D4	D7	1,6
rp8		D4	D7	0,8
rp9		D4	D7	0,4
np10	G14	D7	D7	0,2
np11		D7	D7	0,1
np12		D7	D7	0,05
np13		D7	D7	0,025

Și în acest caz etaloanele din grupele G2 G13 și G14 sunt nediscriminabile, dacă se consideră doar domeniile arătate în tabelul 8.2.

Diagrama P_f marire x10

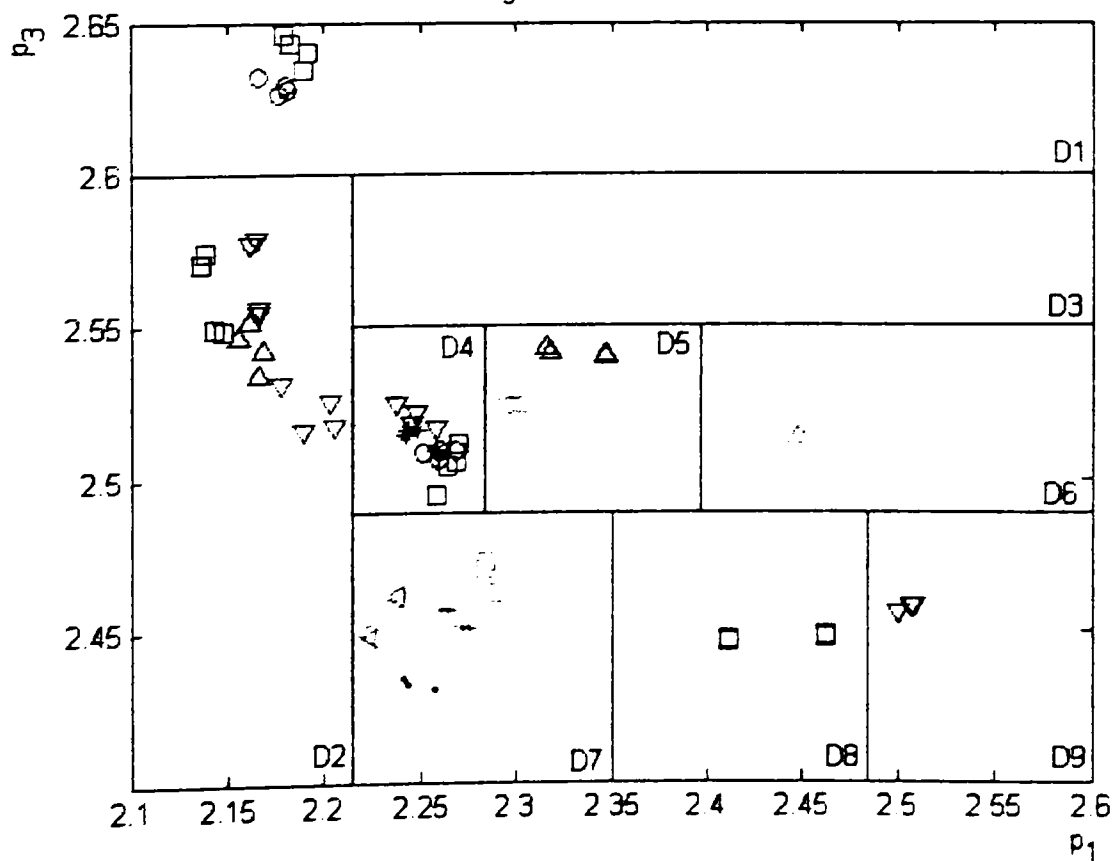


Fig.8.32. Diagrama P_f realizată cu algoritmul MNC3D. etaloane cu mărire x10.

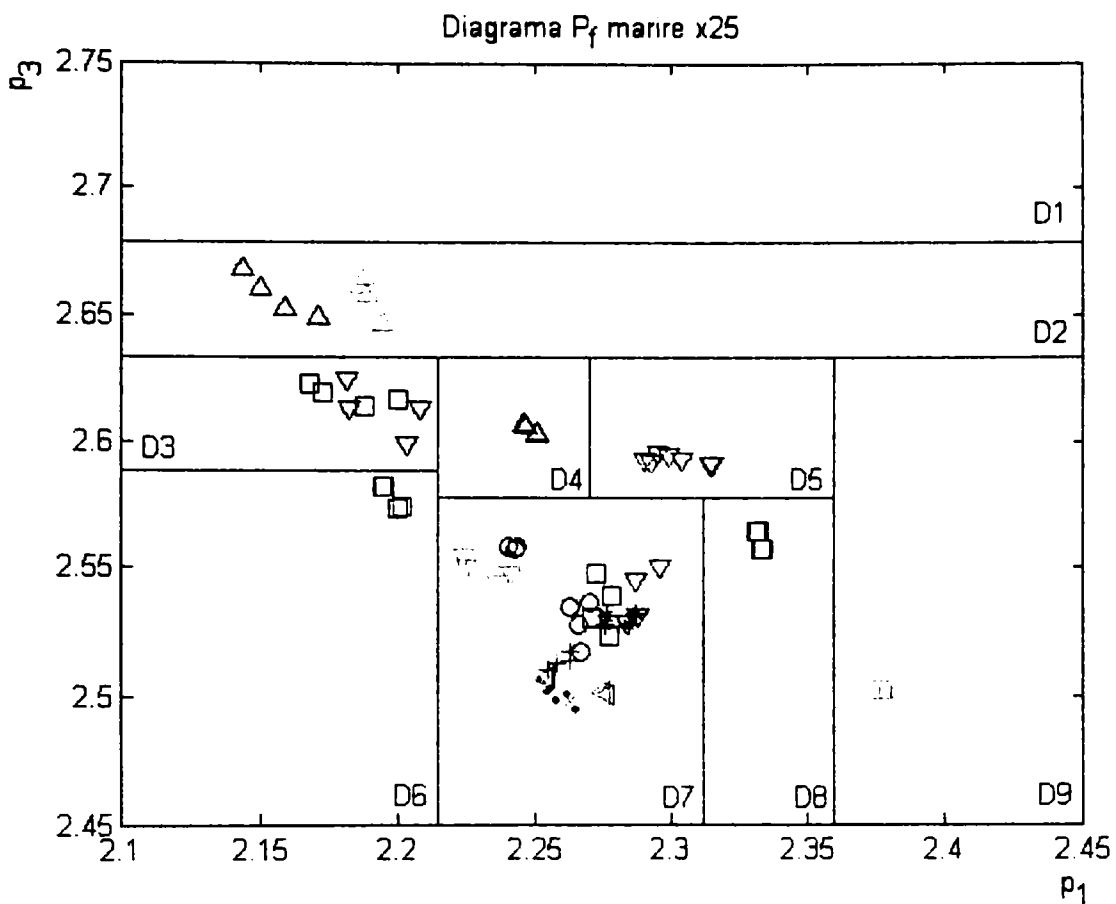


Fig.8.33. Diagrama P_f realizată cu algoritmul MNC3D, etaloane cu mărire x25.

Dacă se generează diagrame P_f numai pentru zona acestor etaloane se obțin diagramele arătate în figurile 8.34, 8.35 pentru etaloane fc, în figura 8.36 și 8.37 pentru etaloane np. Aceste diagrame conțin numai etaloanele ce nu au putut fi dicriminate pe baza tabelului 8.2.

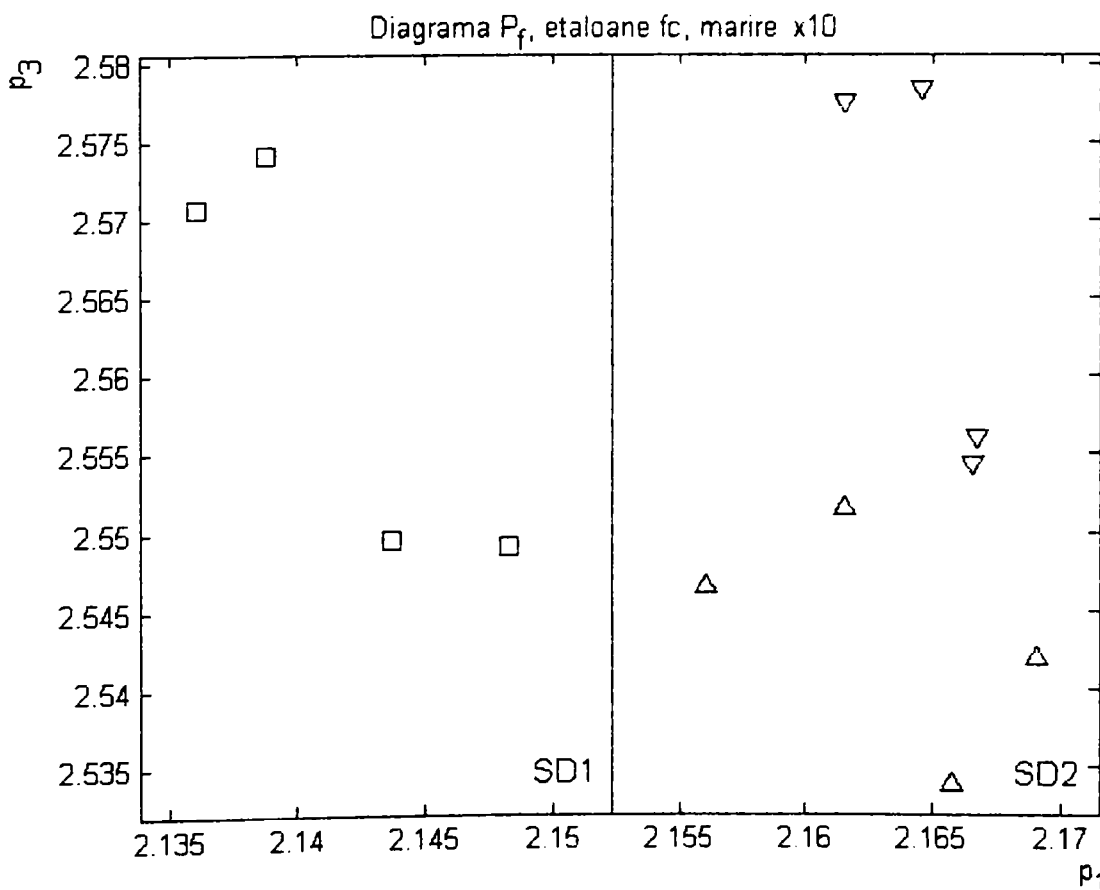


Fig.8.34. Diagrama P_f realizată cu algoritmul MNC3D, etaloane fc cu mărire x10.

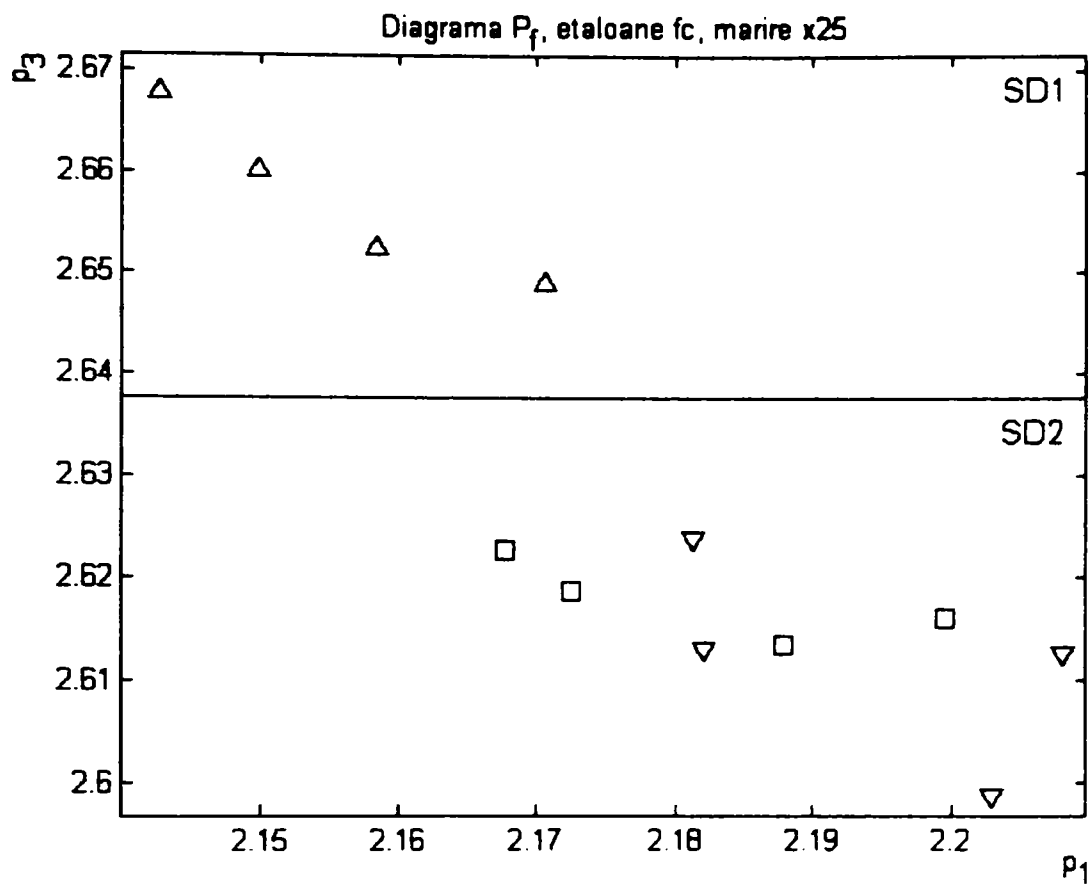


Fig.8.35. Diagrama P_f realizată cu algoritmul MNC3D, etaloane fc cu mărire x25.

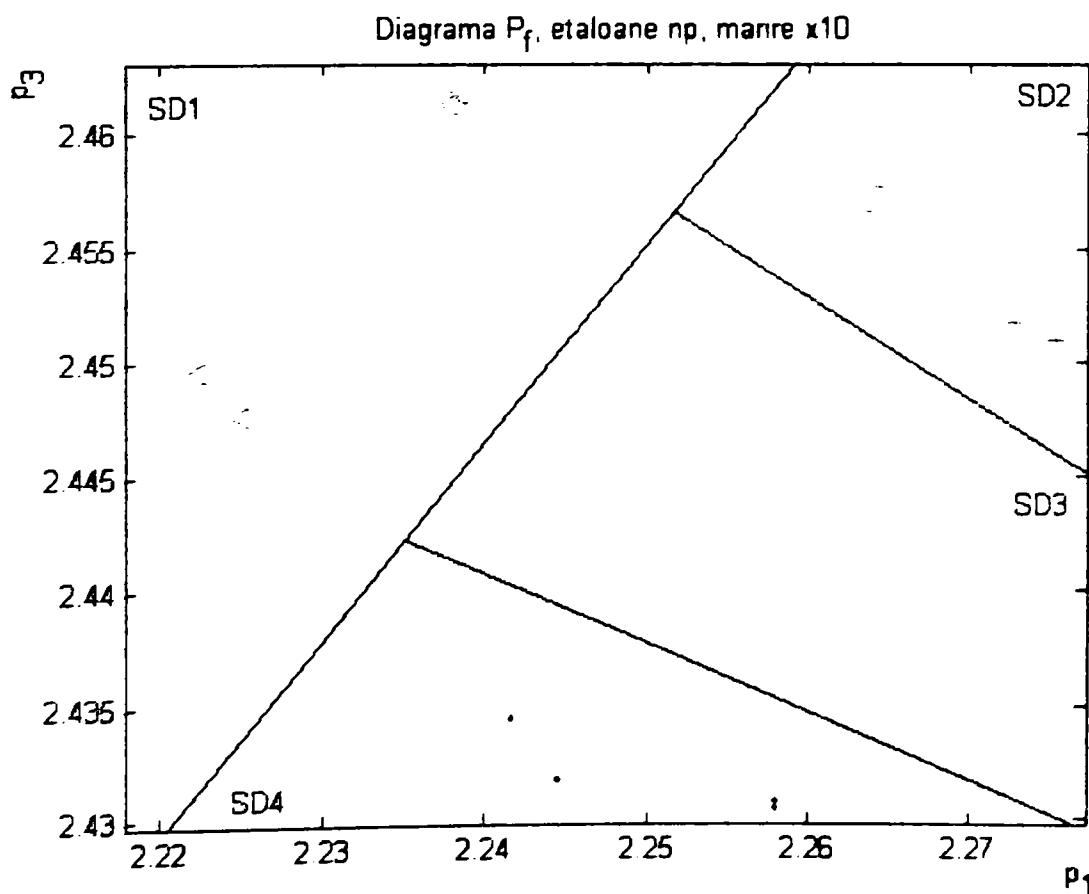


Fig.8.36. Diagrama P_f realizată cu algoritmul MNC3D, etaloane np cu mărire x10.

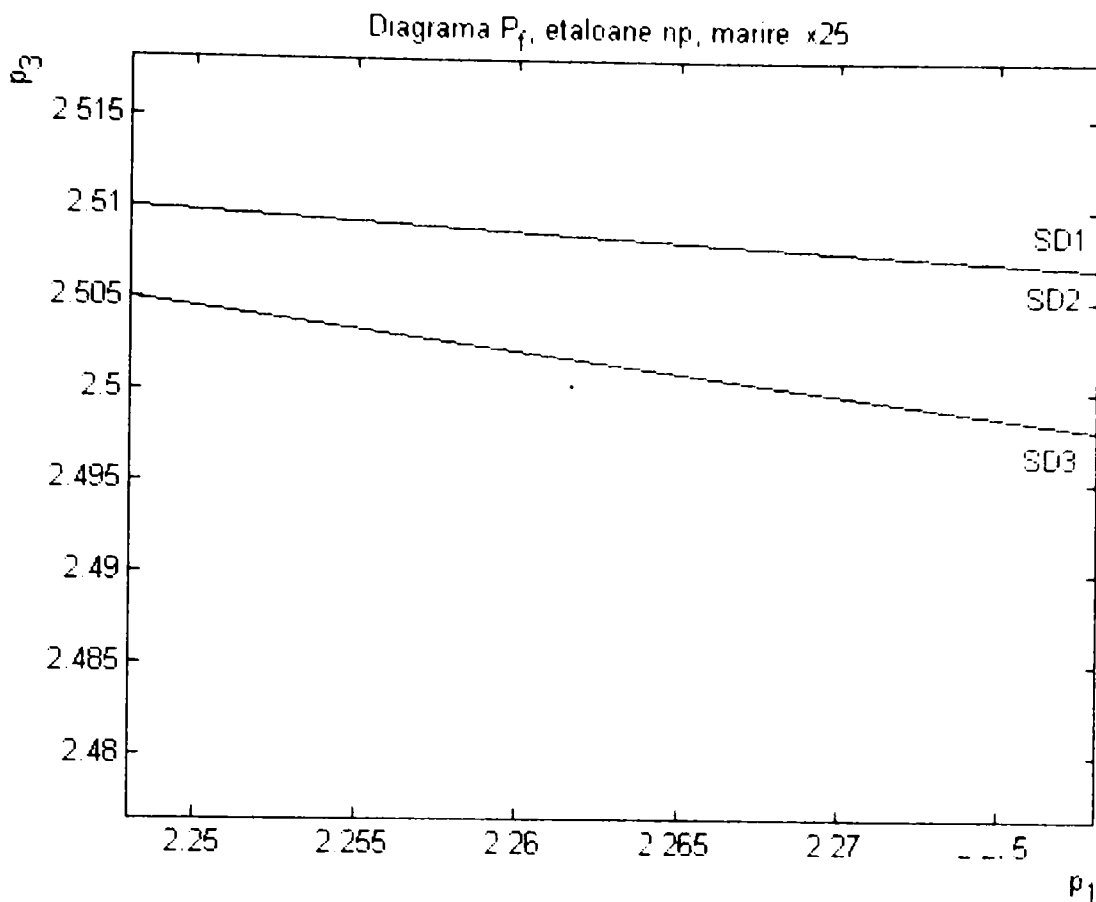


Fig.8.37. Diagrama P_f realizată cu algoritmul MNC3D, etaloane np cu mărire x25.

Se observă că etaloanele fc și np sunt discriminabile dacă se ia în considerare conceptul de „liniar separabilitate”, binecunoscut în teoria rețelelor neuronale artificiale. Acest concept se referă la separabilitatea unor seturi de puncte din plan utilizând o dreaptă definită în planul respectiv. Definind astfel de drepte se pot stabili subdomenii care să realizeze a discriminare acceptabilă.

Pentru etaloanele fc discriminarea se poate realiza considerând apartenența la domenii arătată în tabelul 8.3.

Tabel 8.3.

Cod etalon	Subgrupa etalon G_{et}	Subdomeniu $P_f \times 10$	Subdomeniu $P_f \times 25$	Rugozitate R_a
fc5	G11	SD2	SD1	6,3
fc6	G21	SD2	SD2	3,2
fc7	G22	SD1	SD2	1,6

Pentru etaloanele np subgrupa de discriminare se poate stabili din diagrama cu mărire x10 (figura 8.37), neluând în considerare diagrama cu mărire x25 în care etaloanele sunt mai puțin discriminabile.

Toate diagramele prezentate mai sus, au fost realizate pentru un domeniu de variație al dimensiunii casetei de 1...24 pixeli.

Pentru etaloanele rp stabilirea unor subdomenii este dificilă datorită faptului că seturile de puncte nu sunt liniar separabile. Pentru etaloanele rp s-au încercat și alte domenii de variație a dimensiunii casetei diagramele P_f fiind arătate în

ANEXA 8.10, figurile A.8.10.1. – A.8.10.8. Din aceste diagrame se observă că seturile de etaloane corespunzătoare unei anumite prelucrări și rugozități sunt distribuite diferit în funcție de domeniul de variație al dimensiunii casetei.

Din diagramele prezentate în ANEXA 8.10. se aleg două diagrame în care există o bună discriminare a etaloanelor ce reprezintă rugozități diferite și anume diagramele din figura A.8.10.3 (mărire x10, dimensiunea casetei = 1...16) și figura A.8.10.8. (mărire x25, dimensiunea casetei = 16...32). Stabilirea subdomeniilor de partiționare pentru aceste diagrame este arătată în figurile 8.38 și 8.39.

Se stabilește pentru acest caz următoarea metodă de discriminare :

- dacă parametrii fractali ai eșantionului cu rugozitate necunoscută (C_{ep}) generează un punct în subdomeniul SD2 al diagramei P_f cu mărire x10 atunci eșantionul are o rugozitate corespunzătoare etalonului rp7;
- dacă C_{ep} aparține SD1 al diagramei P_f x10 atunci se analizează P_f x25 :
 - dacă C_{ep} aparține SD1 (P_f x25) atunci rugozitatea eșantionului corespunde etalonului rp8;
 - dacă C_{ep} aparține SD2 (P_f x25) atunci rugozitatea eșantionului corespunde etalonului rp6;
 - dacă C_{ep} aparține SD3 (P_f x25) atunci rugozitatea eșantionului corespunde etalonului rp9;

Analize de acest tip se pot utiliza și în cazul altor etaloane la care se constată o discriminare mai puțin clară.

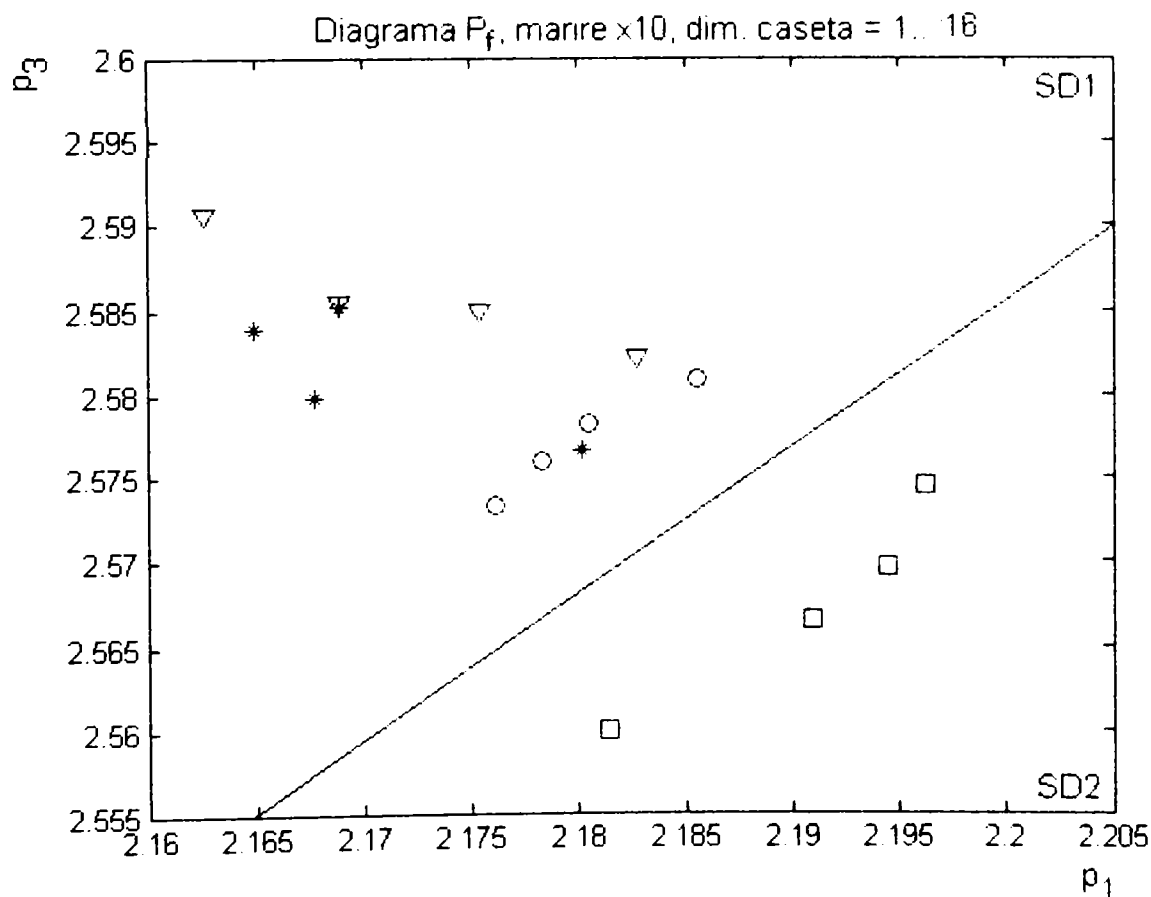


Fig.8.38. Diagrama P_f ,mărire x10, dimensiunea casetei = 1...16.

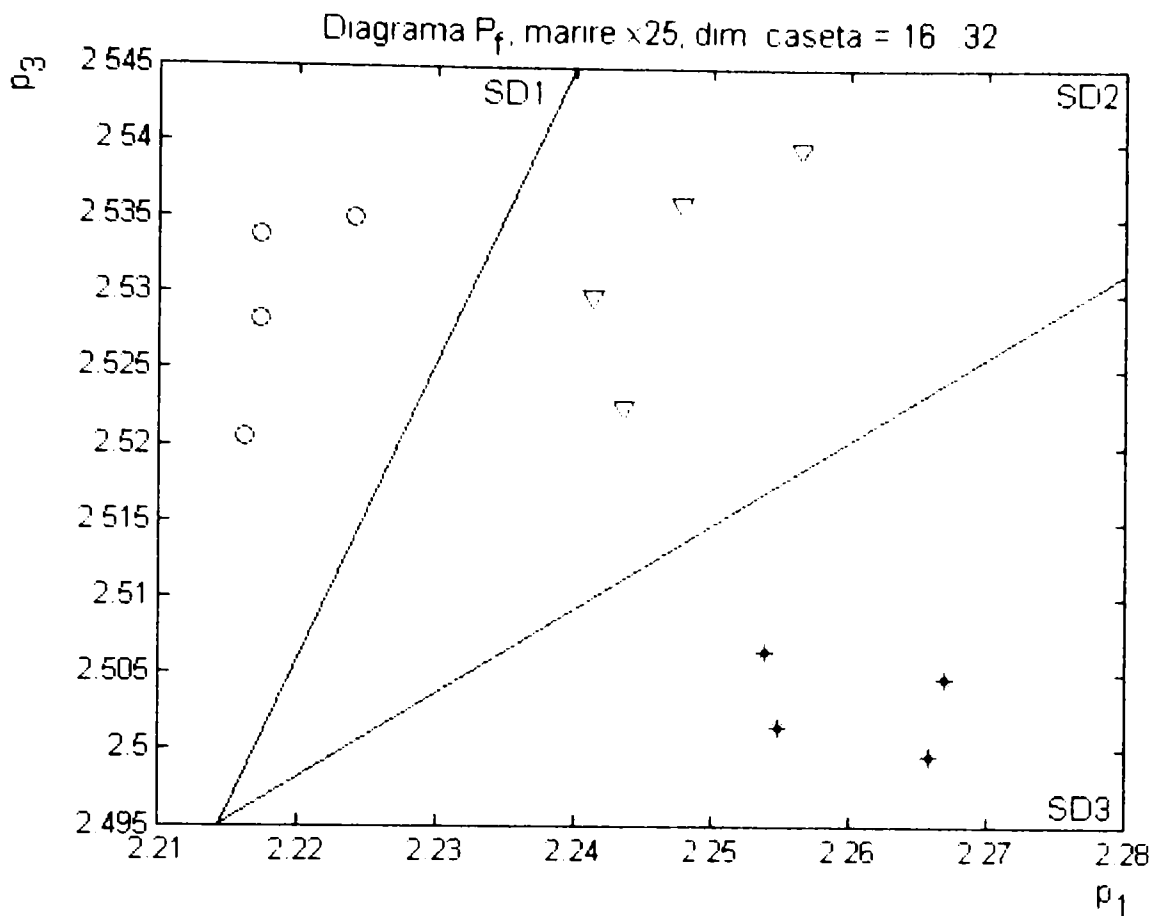


Fig.8.39. Diagrama P_f ,mărire $\times 25$, dimensiunea casetei = 16...32.

Stabilirea domeniilor și subdomeniilor de discriminare se poate realiza în mod automat prin diferite metode numite generic metode „clustering” - partiționare (ca de exemplu metoda „fuzzy c-means”) sau utilizând rețele neuronale ce se pot aplica cu ușurință deoarece seturile de puncte sunt liniar separabile.

O problemă mai delicată este stabilirea automată a domeniului de variație optim al dimensiunii casetei. O soluție ar fi rularea iterativă a programului cu diferite domenii de variație a dimensiunii casetei și interpretarea rezultatelor pe baza unui criteriu de optimizare al grupării punctelor din diagrama P_f .

Studiul acestor probleme rămâne deschis pentru cercetări ulterioare.

8.6.3.3. Metoda spectrală 2D.

Legătura dintre dimensiunea fractală și densitatea spectrală de putere este dată de relațiile (8.19) [JIANG1], (8.24), (8.26) [COSTA]. Similitudinea dintre metoda numărării casetelor și metoda dimensiunii compas pe de o parte și metodele spectrale pe de altă parte are la bază similitudinea dintre inversul dimensiunii casetei și frecvență, ambele mărimi oglindind dependența de scară a caracteristicilor imaginilor analizate.

Pentru analiza caracteristicilor fractale prin metoda spectrală s-a realizat programul MDSPF2D prezentat în ANEXA 8.11. Se amintește că imaginile sunt preprocesate și rotite conform celor expuse anterior. Algoritmul aplicat în cadrul programului se bazează pe caracterul anizotropic al imaginilor etaloanelor de

rugozitate. Se vor calcula parametrii fractali pe linii notate cu indicele x (p'_{1xi} , $i=1...256$) și pe coloane notate cu indicele y (p'_{1yj} , $j = 1...256$), după metoda descrisă în paragraful 8.6.2. (relația (8.26)). Pentru a arăta dispersia parametrilor astfel calculați se prezintă în figura 8.40. diagrama P_f , sub forma unor cercuri de rază σ_{\min} calculată cu ajutorul relațiilor :

$$\sigma_{\min} = \min(\sigma_x, \sigma_y) \quad (8.35)$$

$$\sigma_x = \frac{1}{N-1} \sum_{i=1}^{N-1} (p'_{1xi} - \mu_x)^2 \quad (8.36)$$

$$\sigma_y = \frac{1}{N-1} \sum_{j=1}^{N-1} (p'_{1yj} - \mu_y)^2 \quad (8.37)$$

$$\mu_x = \frac{1}{N} \sum_{i=1}^N p'_{1xi} \quad (8.38)$$

$$\mu_y = \frac{1}{N} \sum_{j=1}^N p'_{1yj} \quad (8.39)$$

unde $N = 256$.

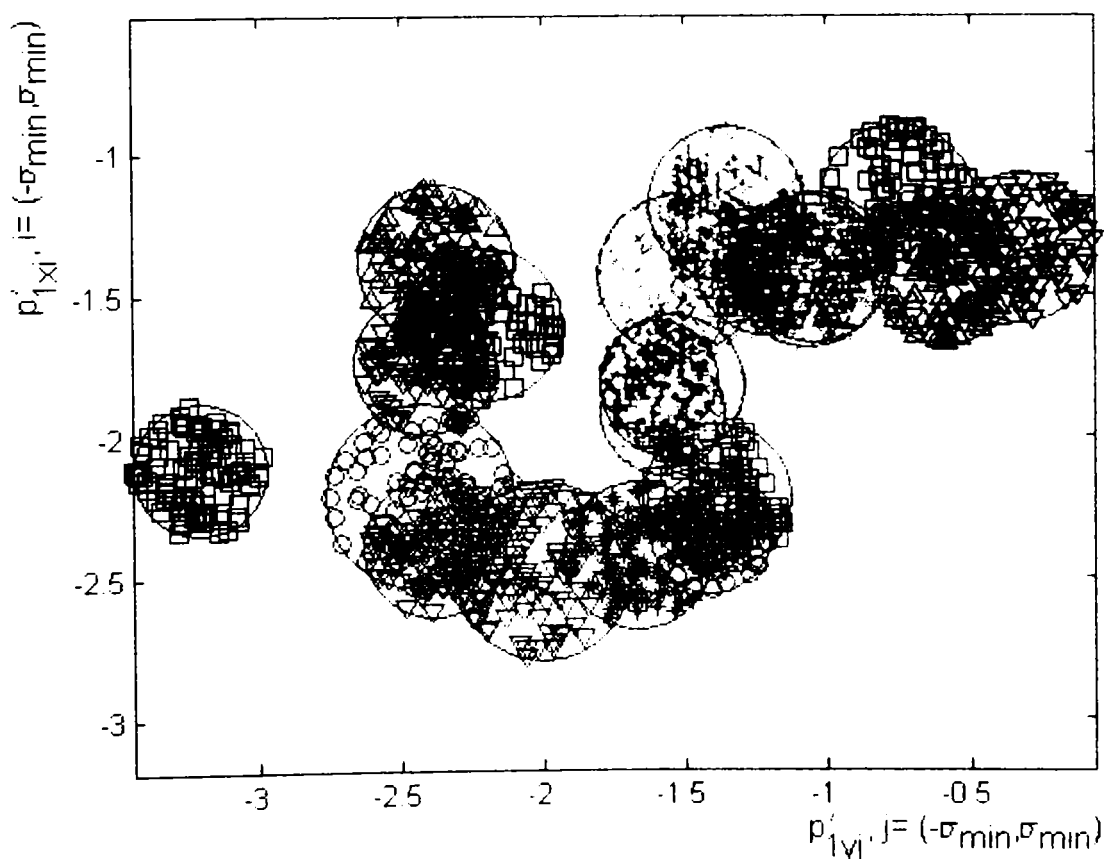


Fig.8.40. Diagrama P_f realizată cu ajutorul algoritmului MDSPF2D, reprezentarea dispersiei valorilor parametrilor fractali.

Diagrama din figura 8.40. a fost realizată pentru un domeniu de frecvențe

$D_f=1...120$ [1/pixeli]. se observă că gruparea diferitelor etaloane respectă, în principiu criteriul procedurii tehnologice utilizat la obținerea etalonului. În cursul încercărilor s-a constatat că diferite etaloane pot fi discriminate pe diferite domenii de frecvență. În consecință s-a elaborat un algoritm euristic de discriminare care poate sta la baza unei metode automate fuzzy sau neuronale (utilizând conceptul de liniar separabilitate). În continuare pentru diagramele P_f se utilizează parametrii μ_x și μ_y dați de relațiile (8.38),(8.39).

În figura 8.41 se prezintă schema discriminării etaloanelor pe baza algoritmului MDSPF2D. În cadrul schemei se utilizează imagini ale etaloanelor pentru două mărimi x10 și x25, și diferite domenii de frecvență D_f .

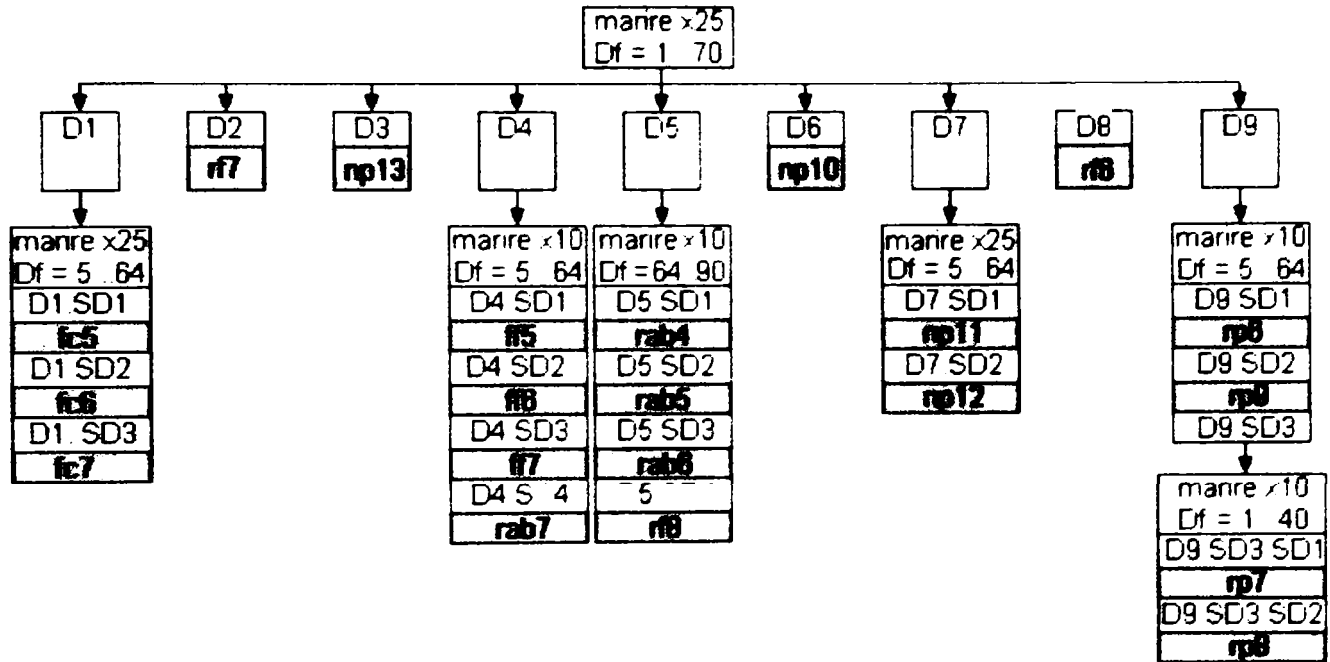


Fig.8.41. Schema discriminării etaloanelor pe baza algoritmului MDSPF2D.

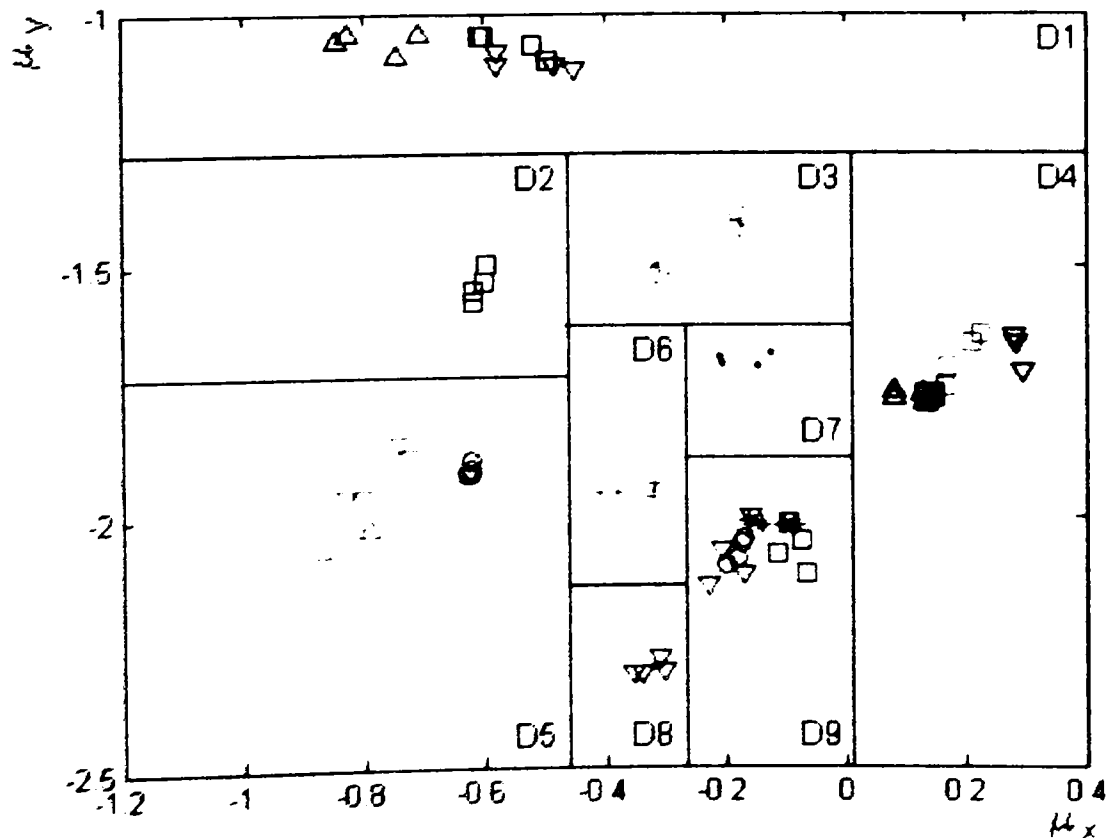


Fig.8.42. Diagrama P_f . mărime x25. Domeniu de frecvență $D_f = 1...70$.

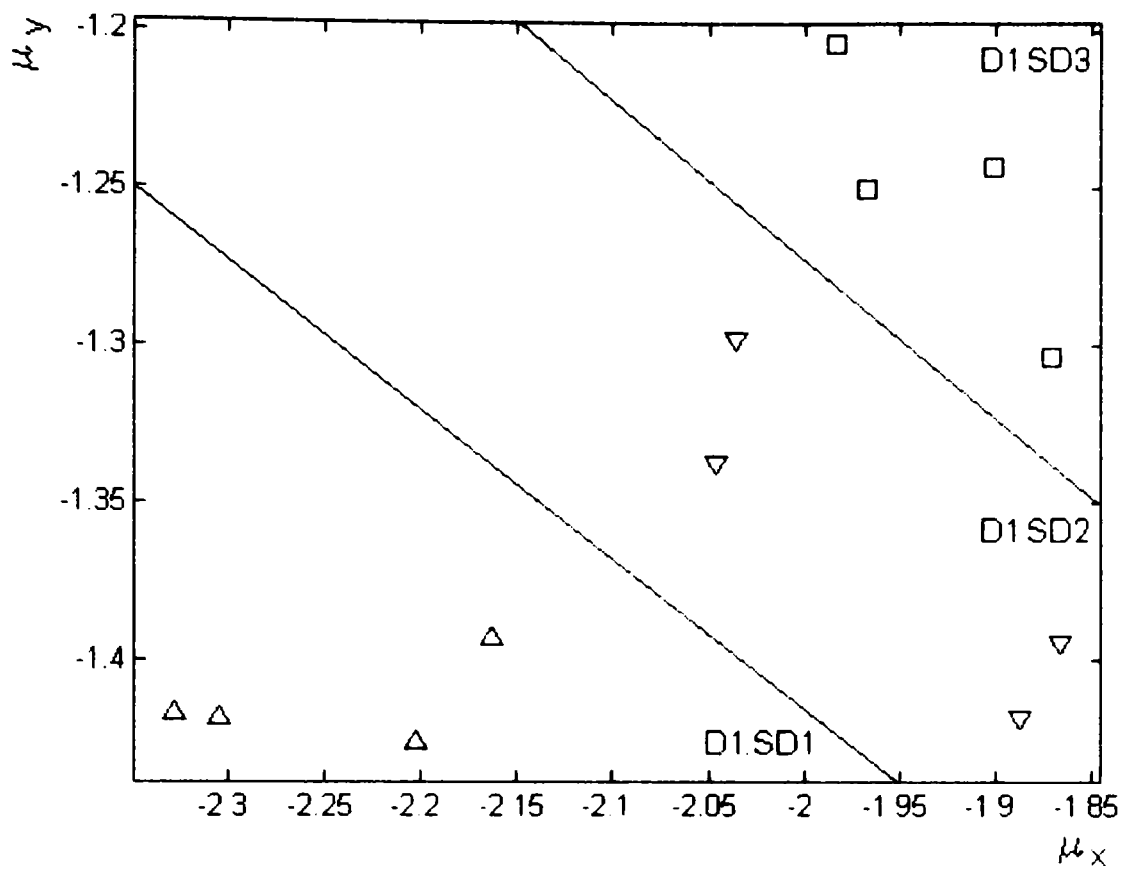


Fig.8.43. Diagrama Pf pentru D1, mărire x25, Domeniu de frecvențe Df = 5...64.

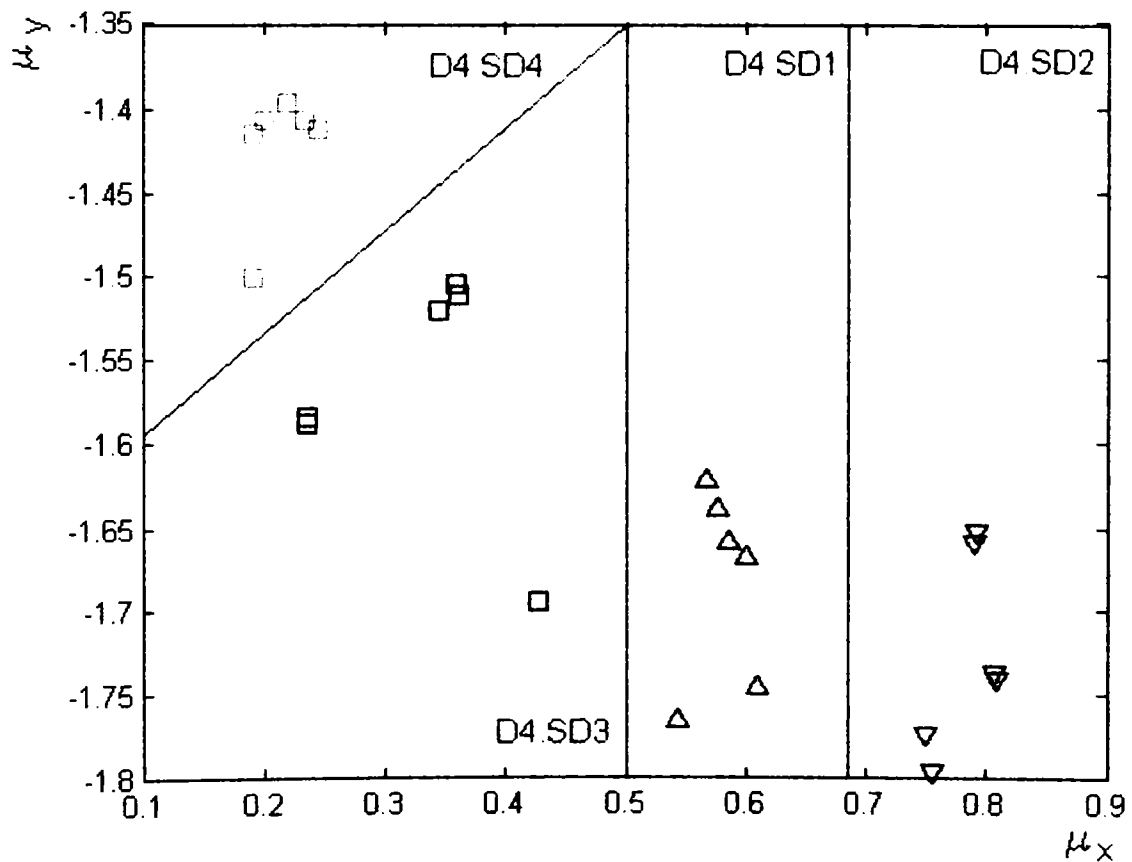


Fig.8.44. Diagrama Pf pentru D4, mărire x10, Domeniu de frecvențe Df = 5...64.

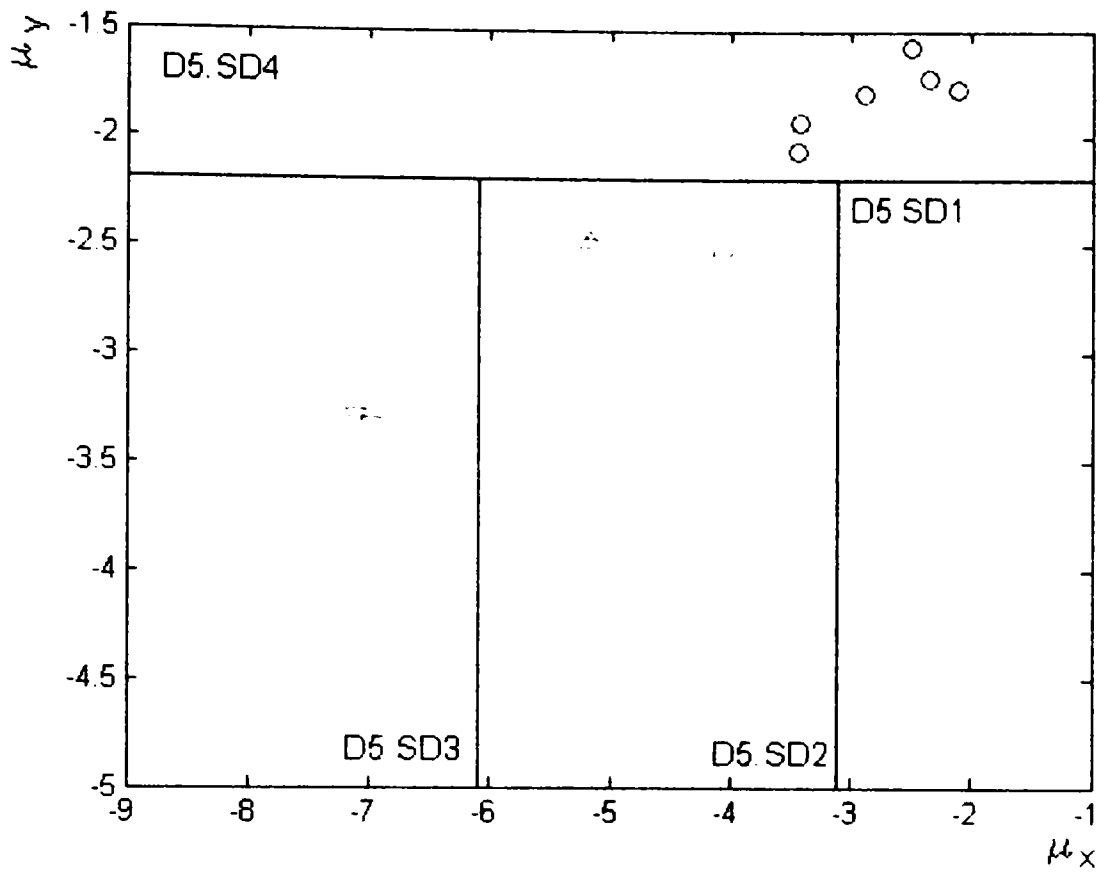


Fig.8.45. Diagrama Pf pentru D5, mărime x10, Domeniu de frecvențe Df = 64...90.

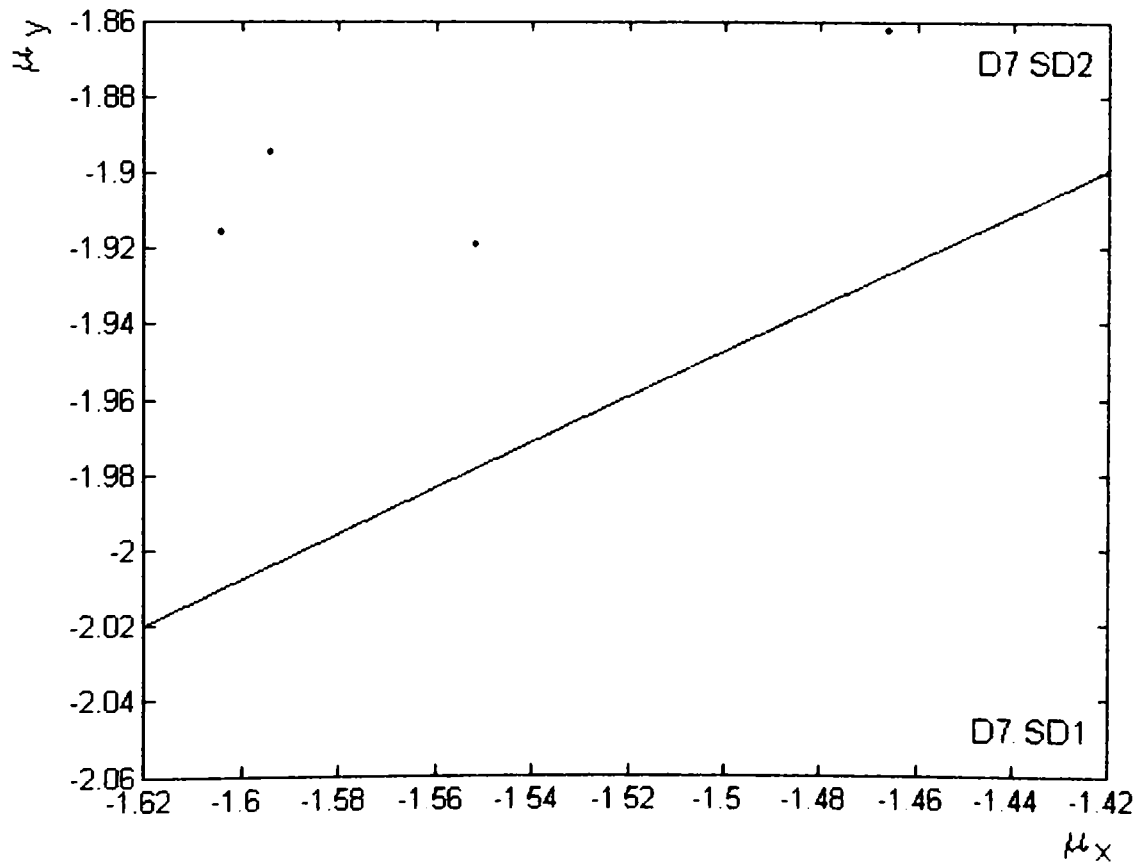


Fig.8.46. Diagrama Pf pentru D7, mărime x25, Domeniu de frecvențe Df = 5...64.

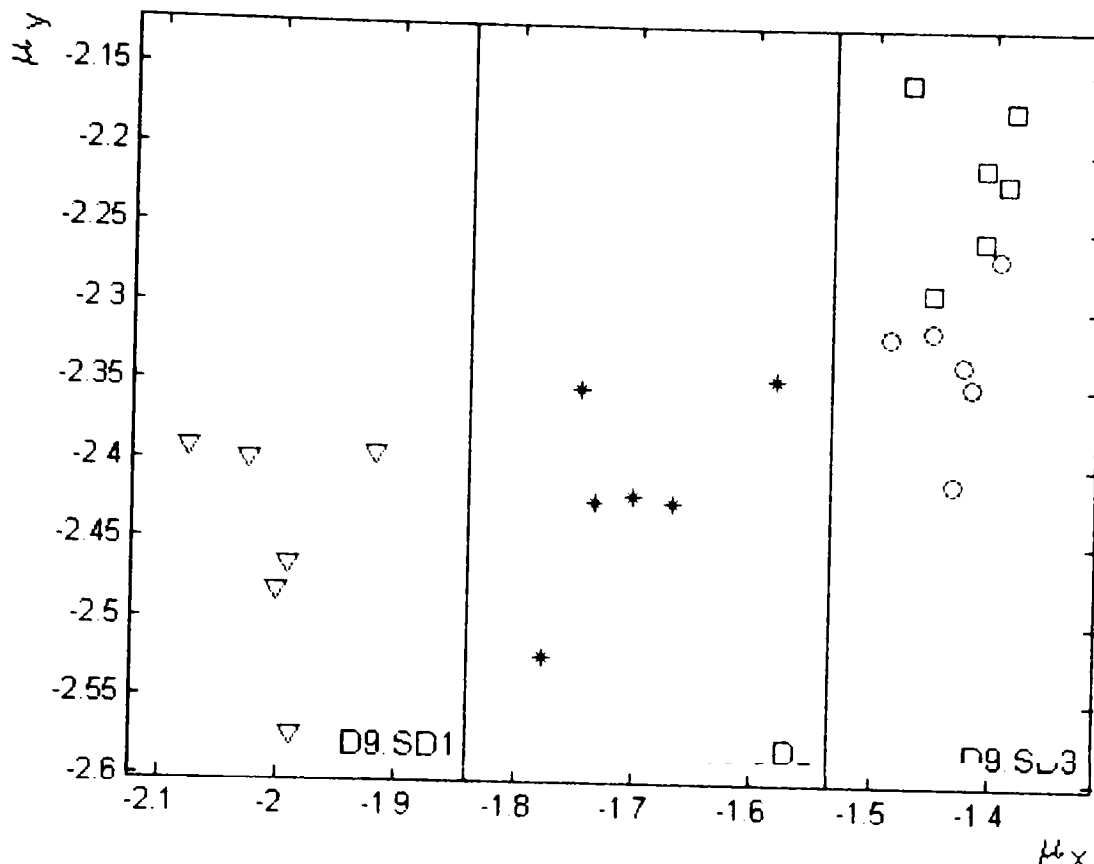


Fig.8.47. Diagrama Pf pentru D9, mărire x10, Domeniu de frecvențe Df = 5...64.

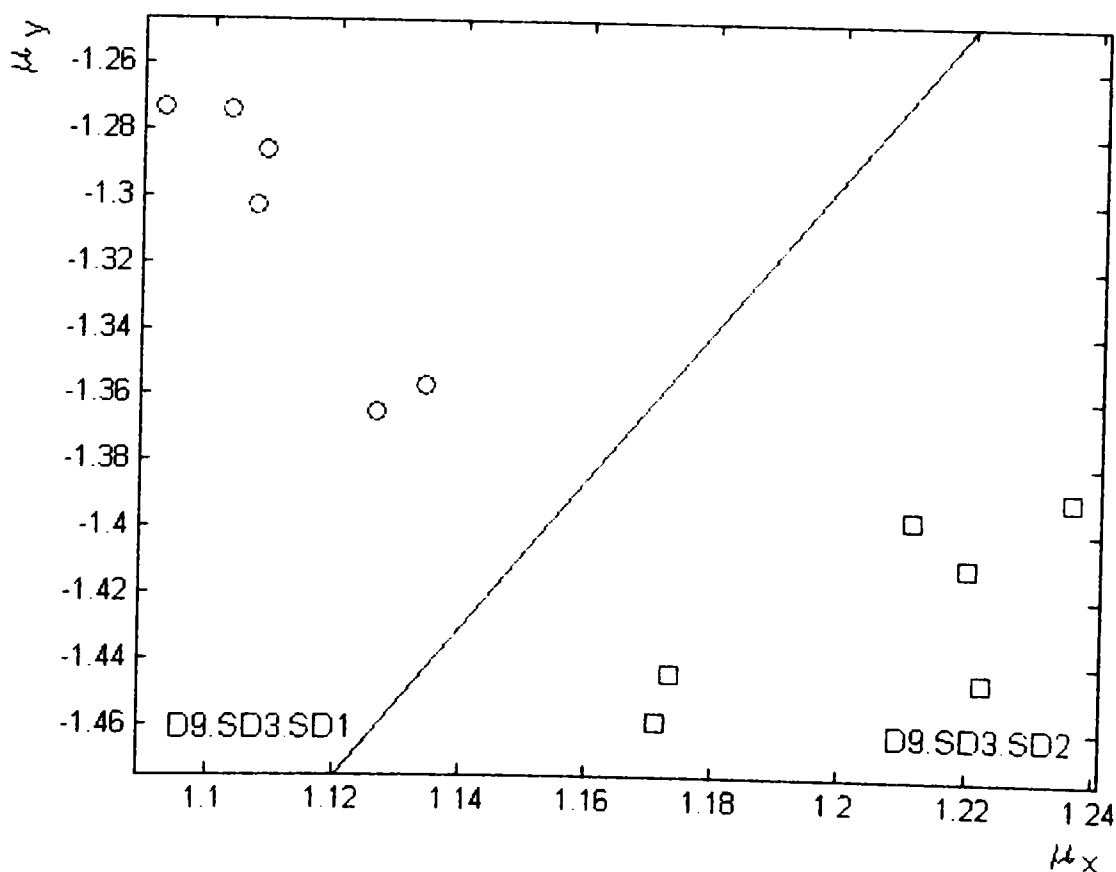


Fig.4.48. Diagrama Pf pentru D9.SD3, mărire x10, Domeniu de frecvențe Df = 1...40.

Domeniile de frecvență utilizate au fost stabilite experimental și pot fi optimizate în continuare, în cadrul unor studii ulterioare, pentru obținerea unei grupări mai bune a imaginilor ce corespund aceluiași etalon și pentru o mai bună

discriminare a etaloanelor. Pentru fiecare domeniu sau subdomeniu al diagramelor P_f se poate stabili o valoare centrală a domeniului iar aceasta se poate utiliza pentru definirea unui domeniu probabilistic gaussian. Cu ajutorul metodelor fractale utilizate s-a reușit caracterizarea imaginilor etaloanelor prin doi parametrii. Pentru o mai bună discriminare se pot utiliza pe lângă aceștia și parametrii statistici sau obținuți prin alte metode.

Faptul că valorile parametrilor fractali variază cu mărirea utilizată cât și cu dimensiunea casetei (în cazul algoritmilor MNC) sau cu frecvența (în cazul algoritmului MDSP2D) arată că imaginile analizate au un caracter pseudo-fractal sau multi-fractal.

Din punctul de vedere al timpilor de rulare a programelor realizate se evidențiază clar că metoda MDSPF2D este superioară metodelor MNC2D și MNC3D. Programele au fost rulate pe un calculator Pentium IV cu viteza procesorului de 1,5 GHz, și o memorie RAM de 256 Mb. Pentru stabilirea caracteristicilor fractale a unei singure imagini s-a înregistrat pentru algoritmul MDSPF2D 1,2 secunde iar pentru MNC2D și MNC3D timpii de rulare au fost de ordinul minutelor.

Se consideră că algoritmul MDSPF2D corespunde scopului propus de a stabili rugozitatea eșantionului atât la intrarea acestuia în sistem cât și după operația de șlefuire.

9. Analiza computerizată a microstructurilor metalografice.

9.1. Considerații generale.

Analiza metalografică, realizată în mod curent, implică o mare varietate de analize, fiecare dintre ele fiind rezultatul aplicării unor cunoștințe testate în practică.

Fiecare tip de analiză prezintă particularități esențiale ce nu pot fi nesocotite în încercarea unor generalizări. Din această cauză automatizarea analizei metalografice poate fi rezolvată mai degrabă printr-o abordare de tip sistem expert, în care particularitățile fiecărui tip de analiză să fie specificate într-o bază de cunoștințe.

Având în vedere mediul tehnico-economic în care se dorește utilizarea CFLM și informațiile cunoscute în prealabil despre eșantionul ce urmează a fi analizat, din multitudinea de analize posibile, s-au sintetizat două categorii importante.

Cele două categorii de analize sunt:

- identificarea microstructurii – în acest caz se consideră că se cunoaște în prealabil că microstructura analizată face parte dintr-un grup de microstructuri și trebuie identificată o anumită microstructură din acest grup, în acest sens s-au testat două subcategorii :
 - tipuri de microstructuri cu aspect vizual asemănător - tip A;
 - tipuri de microstructuri la care se cunoaște compoziția chimică și trebuie determinată starea materialului (laminat la cald, normalizat, recopt) – tip B;
- determinarea punctajului microstructurii – în acest caz pe lângă identificarea microstructurii dintr-un anumit grup (în cadrul grupului microstructurile sunt plasate într-o anumită ordine ierarhică definită prin *punctaj* sau procentaj), va trebui stabilită o relație univocă între caracteristica determinată din analiza imaginii și punctaj – tip C;

Pentru tipurile de analize menționate s-au stabilit grupe de imagini pentru care s-au testat algoritmi de procesare realizați.

Pentru analizele de tip A au fost stabilite grupe de imagini care diferă din punct de vedere microstructural dar au aspecte vizuale apropiate (ANEXA 9.1.). Pentru analizele de tip B s-au stabilit grupe de imagini ce reprezintă microstructuri care au aceeași compoziție chimică dar diferă prin starea materialului (ANEXA 9.3.). Pentru analizele de tip C s-au stabilit grupe de imagini a căror microstructură diferă prin variația unei caracteristici specifice, ca de exemplu finețea, procentajul din suprafața totală, lungimea (ANEXA 9.5.).

Pentru a elimina pericolul testării algoritmilor pe cazuri particulare, imaginile utilizate sunt preluate din [RADULESCU], admițându-se că imaginile prezentate în acest catalog caracterizează corect aspectul vizual al microstructurilor. Colecții similare de imagini ale structurilor metalografice pot fi găsite și în cataloagele în format electronic [METALLOROM] și [METADEX] sau în standardul SR 5000:1997

(domeniu B02).

Metoda de lucru este similară cu cea folosită pentru analiza rugozității suprafeței eșantioanelor unde s-au utilizat etaloane de rugozitate.

Observând imaginile prezentate în ANEXELE 9.1., 9.3., 9.5. se constată că acestea nu prezintă o anizotropie evidentă, așadar în acest caz se pot utiliza doar algoritmi de analiză invarianți la rotație (nu se pot utiliza algoritmi de tipul MDSPF2D). O altă observație importantă este legată de modul în care se realizează iluminarea eșantionului. Microscopul optic metalografic utilizează iluminarea prin obiectiv, aceasta conferind o omogenitate a intensității luminoase reflectate ceea ce simplifică preprocesarea imaginii. Eliminarea zonelor din imagine ce nu conțin informații utile se poate realiza în mod similar cu cel menționat în cazul analizei rugozității (capitolul 8).

Pentru fiecare din imaginile prezentate în ANEXA 9.1., 9.3., 9.5. sunt prelevate patru imagini de dimensiuni mai mici (512x512 pixeli) pentru a verifica comportamentul algoritmului la procesarea unor zone diferite din aceeași microstructură.

Datorită faptului că la analiza rugozităților s-a utilizat cu succes metoda fractală, această metodă a fost aplicată în primul rând și în cazul analizei microstructurilor. În cazurile în care rezultatele nu au fost pe deplin concludente la parametrii fractali s-au adăugat parametrii statistici, s-au utilizat alte tipuri de parametrii de caracterizare. În majoritatea cazurilor a fost suficientă completarea parametrilor fractali cu un singur parametru statistic pentru realizarea unei discriminări acceptabile.

Datorită faptului că imaginile microstructurilor nu prezintă o anizotropie clară nu se poate aplica algoritmul MDSPF2D. Utilizarea algoritmului MNC3D este teoretic posibilă dar s-a renunțat la testarea acestuia datorită timpilor de rulare mari care ar face dificilă folosirea acestui algoritm în cadrul CFLM, în condițiile în care se lucrează cu mediul de programare MATLAB.

9.2. Tratarea analizelor de tip A.

Metoda spectrală de calcul a parametrilor fractali se poate extinde și pentru cazul 3D (MDSPF3D) utilizându-se diagrame ale densității spectrale de putere, obținute prin aplicarea algoritmului FFT2 descris în capitolul 3.3.4. Rezultatul aplicării algoritmului FFT2 asupra imaginii unei microstructuri este arătat sub forma diagramei DSP spațiale în figura 9.1.

Relația (8.24.), pentru cazul 3D [COSTA], devine:

$$DSP(k) = \frac{1}{\|k\|^{p_1-1}} \quad (9.1)$$

unde $\|\mathbf{k}\| = \sqrt{f_x^2 + f_y^2}$; f_x – frecvența spațială pe linii; f_y – frecvența spațială pe coloane; $\|\mathbf{k}\|$ reprezintă frecvența spațială pe o direcție oarecare.

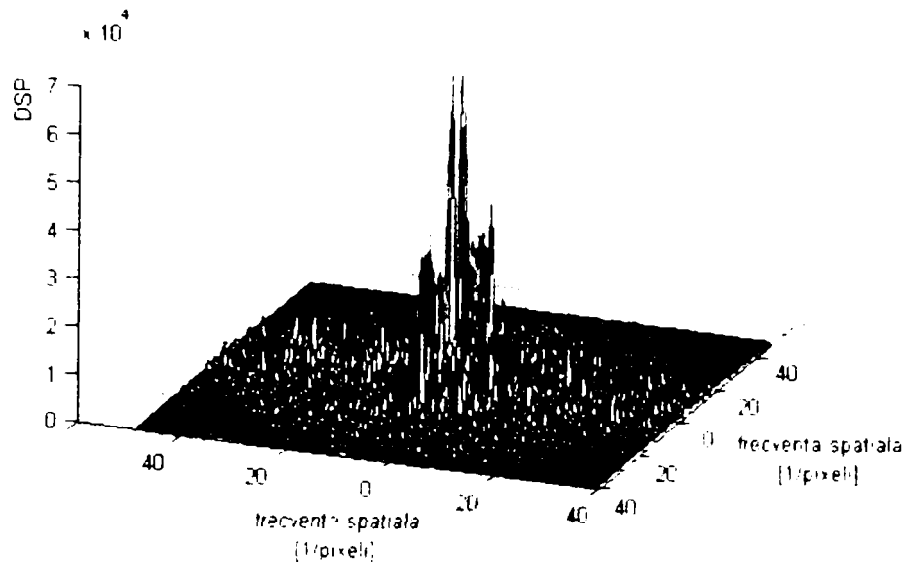


Fig.9.1. Diagrama DSP realizată cu algoritmul FFT2.

Dimensiunea fractală în acest caz se calculează cu relația :

$$D = \frac{(8 - p_1'')}{2} \quad (9.2)$$

Calculul dimensiunii fractale în funcție de DSP în cazul 3D este utilizată și în lucrarea [JIANG1]. Dimensiunea fractală D nu are o relevanță deosebită în contextul în care se urmărește discriminarea relativă a imaginilor microstructurilor. Din această cauză caracteristica fractală utilizată în continuare este parametrul p_1'' .

Din relația 9.1 se observă că valorile corespunzătoare unei distanțe $\|\mathbf{k}\|$ de la originea DSP 3D (figura 9.1), reflectă amplitudinea DSP la aceeași frecvență. Dacă se consideră cercuri cu centrul în origine de rază $\|\mathbf{k}_n\|$, $n = 1 \dots (N-1)/2$, (matrice imagine de dimensiunea $N \times N$) pentru fiecare cerc se poate defini media $DSP_{med}(k_n)$, dată de relația:

$$DSP_{med}(k_n) = \frac{1}{m_n} \sum_{h=1}^{m_n} DSP_h(k_n) \quad (9.3)$$

unde m_n reprezintă numărul de puncte pe cercul de rază k_n .

Setul de valori $DSP_{med}(k_n)$ calculat cu relația (9.3) se reprezintă în funcție de k_n în scară dublu logaritmică obținându-se setul de puncte (marcate cu *) reprezentat în

diagrama din figura 9.2. Dreapta de aproximare (calculată cu metoda celor mai mici pătrate) exprimă dependența :

$$\log(\text{DSP}_{med}(k_n)) = p_1''(\log(k_n)) + p_2'' \quad (9.4)$$

a cărei pantă p_1'' și ordonată la origine p_2'' sunt parametrii fractali utilizați pentru caracterizarea microstructurilor.

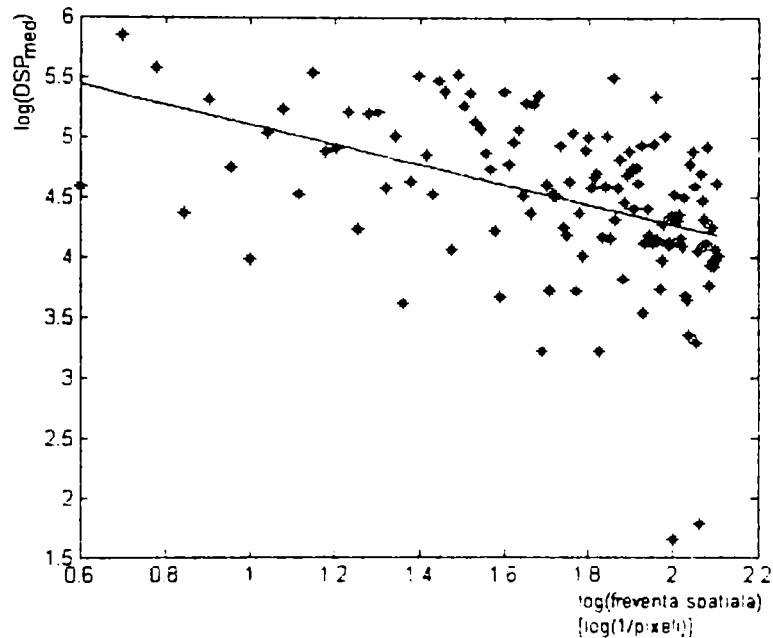


Fig.9.2. Diagrama dreptei de aproximare a setului de valori DSP_{med} .

Diagrama din figura 9.2. este similară cu diagrama Richardson având pe ordonată $\log(\text{DSP}_{med})$ o măsură a amplitudinii iar pe abscisă $\log(k_n)$ o măsură a frecvenței spațiale ce poate fi privită ca o mărime similară cu scara sau rezoluția.

Caracteristica fractală poate fi calculată și după o anumită direcție, de-a lungul unei drepte trasate din originea diagramei DSP 3D, sub un anumit unghi α , dar această metodă se utilizează cu precădere în cazul unor structuri anizotrope.

După calculul parametrilor fractali p_1'' și p_2'' se generează diagrama $P_f(p_1'', p_2'')$ în mod similar cu cea definită în cazul analizei rugozităților (capitolul 8) și se studiază modul în care se grupează punctele corespunzătoare imaginilor diferitelor structuri, stabilindu-se apoi domenii de discriminare pentru fiecare din acestea.

În cursul testării algoritmilor s-au constatat două probleme importante. Au existat cazuri în care discriminarea diferitelor microstructuri nu a fost suficient de pregnantă din care cauză a fost nevoie să se adauge un parametru statistic. Acest parametru s-a notat p_h și a fost calculat cu ajutorul histogrammei imaginii $h(i)$ (relația (3.62), capitolul 3), fiind definit ca:

$$p_h = i_{max} \quad (9.5)$$

unde i_{max} este intensitatea corespunzătoare valorii maxime a histogramei $h(i_{max}) = \max(h(i))$, $i=1...256$. Pentru ca parametrul să reprezintă mai bine o caracteristică globală se utilizează un număr mai mic de intervale i_{hmed} (20, 50, 100) în locul a 256 de nivele de intensitate. Se obțin astfel diagrame P_f - P_h spațiale în care p_h este reprezentat pe verticală.

A doua problemă constatată în cursul testărilor a fost aranjarea punctelor caracteristice în diagramele P_f după o anumită direcție, sub un unghi δ de aproximativ -45° . Pentru a elimina această tendință s-au realizat diagramele P_{rf} (de coordonate p_{r1}'' - p_{r2}'') și diagramele 3D, P_{rf} - P_h , prin rotirea sistemului de coordonate p_1'' - p_2'' cu unghiul δ (figura 9.3.).

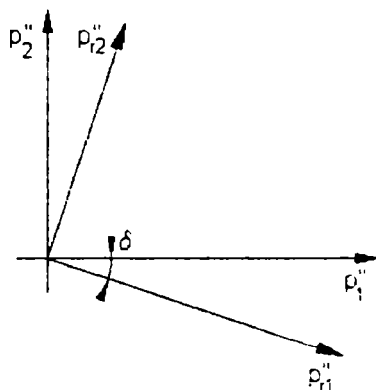


Fig.9.3. Rotirea sistemului de coordonate p_1'' - p_2'' .

Rotirea sistemului de coordonate este realizată pentru o mai bună observabilitate vizuală a discriminării punctelor caracteristice. În cazul prelucrării și interpretării automate a rezultatelor această operație poate să lipsească.

Algoritmul MDSPF3D poate fi particularizat pentru fiecare tip de analiză prin stabilirea domeniilor de variație a parametrilor k_n și i_{hmed} . Determinarea acestor parametri se realizează prin rularea algoritmului pentru diferite valori și observarea rezultatelor obținute. Stabilirea unui anumit domeniu pentru k_n este echivalentă cu stabilirea valorilor limită a unui filtru „trece-bandă” în domeniul frecvențelor spațiale, iar stabilirea unui anumit număr de intervale i_{hmed} este echivalentă cu definirea unui filtru de mediere.

S-au stabilit grupe de microstructuri ce au aspect vizual asemănător, considerându-se că dacă acestea pot fi discriminate acceptabil atunci structurile ce au aspecte vizuale mult diferite vor fi și mai ușor discriminabile.

Gruparea microstructurilor testate cât și simbolizarea punctelor caracteristice în diagrame este prezentată în tabelul A.9.1.1 (ANEXA 9.1).

Algoritmul MDSPF3D a fost implementat sub mediul MATLAB, codul sursă al programului („Identificare_A”) fiind dat în ANEXA 9.2. Timpul de rulare al programului pe un sistem Pentium IV, cu o memorie RAM de 256 Mb, pentru o imagine, este de 2 secunde.

Rezultatele testării algoritmului MDSPF3D în cazul identificării este arătat în ANEXA 9.1. figurile A.9.1.18-A.9.1.29. Pentru fiecare tip de analiză de tip A s-au prezentat atât diagramele P_f sau P_{rf} cât și diagramele P_f - P_h sau P_{rf} - P_h .

Se observă că în toate cazurile s-a obținut o discriminare bună a punctelor caracteristice pentru $k_n=1...255$ și $i_{hmed}=20$, mai puțin pentru cazul microstructurilor cu grafit nodular unde a fost necesară utilizarea unui domeniu $k_n=16...96$. În acest caz structurile nu au o redundanță satisfăcătoare parametrii fractali reușind o caracterizare mai slabă în domeniul $k_n=1...255$.

Ca și în cazul diagramelor P_f realizate pentru cazul rugozității și aici se pot stabili domenii și subdomenii conceptul de liniar separabilitate extinzându-se la 3D în loc de liniile ce despart domeniile utilizându-se plane de separare.

9.3. Tratarea analizelor de tip B.

Pentru analizele de tipul B s-au studiat microstructurile a patru tipuri de oțeluri cu conținut de carbon diferit, fiecare dintre acestea având trei stări : laminat la cald, normalizat și recopt. Diagramele P_{rf} și $P_{rf}-P_h$ rezultate sunt prezentate în ANEXA 9.3 figurile A.9.3.13-A.9.3.20. La toate microstructurile testate s-a obținut o discriminare bună pentru $k_n=1..255$ și $i_{hmed}=20$. Programul „Identificare_B” prezentat în ANEXA 9.4. diferă de programul utilizat la analizele de tip A prin utilizarea relației :

$$\log(\log(DSP_{med}(k_n))) = p_1^*(\log(k_n)) + p_2^* \quad (9.6)$$

în locul relației (9.4). La acest tip de analiză, prin utilizarea parametrilor fractali definiți de relația (9.6), s-a obținut o discriminare mai bună a punctelor caracteristice decât în cazul utilizării relației (9.4). Simbolizarea punctelor caracteristice este dată în tabelul A.9.3.1.

9.4. Tratarea analizelor de tip C.

Tratarea analizelor de tip C diferă de tratarea celorlalte tipuri de analize prin necesitatea obținerii unei dependențe univoce între valoarea caracteristicii obținute prin analiza imaginii și numărul de ordine al microstructurii într-o ierarhie stabilită în cadrul unui anumit grup. Acest număr de ordine este denumit punctaj al microstructurii.

Au fost testate patru analize de acest tip, elaborându-se algoritmi specifici pentru fiecare caz:

- determinarea punctajului microstructurilor fero-perlitice în șiruri pentru oțeluri cu 0,15%C (ANEXA 9.5 figurile A.9.5.1-A.9.5.6);
- determinarea punctajului ce caracterizează finețea perlitei în oțeluri cu 0,15%C (ANEXA 9.5 figurile A.9.5.7-A.9.5.16);
- determinarea procentului suprafeței perlitei pentru fonte cu grafit nodular (ANEXA 9.5. figurile A.9.5.17-A.9.5.22);
- determinarea lungimii lamelelor la fonte cu grafit lamelar (ANEXA 9.5

figurile A.9.5.23-A.9.5.28).

Simbolizarea punctelor caracteristice în diagrame este dată în tabelul A.9.5.1.

Determinarea punctajului microstructurilor ferito-perlitice în șiruri.

Inițial s-a încercat discriminarea microstructurilor și stabilirea unei corelații între caracteristica fractală și punctaj pe baza algoritmului MDSPF3D. Rezultatele obținute sunt arătate în diagramele din figurile 9.4. și 9.5. pentru care s-au utilizat parametrii $k_n = 1 \dots 255$, $i_{hmed} = 50$.

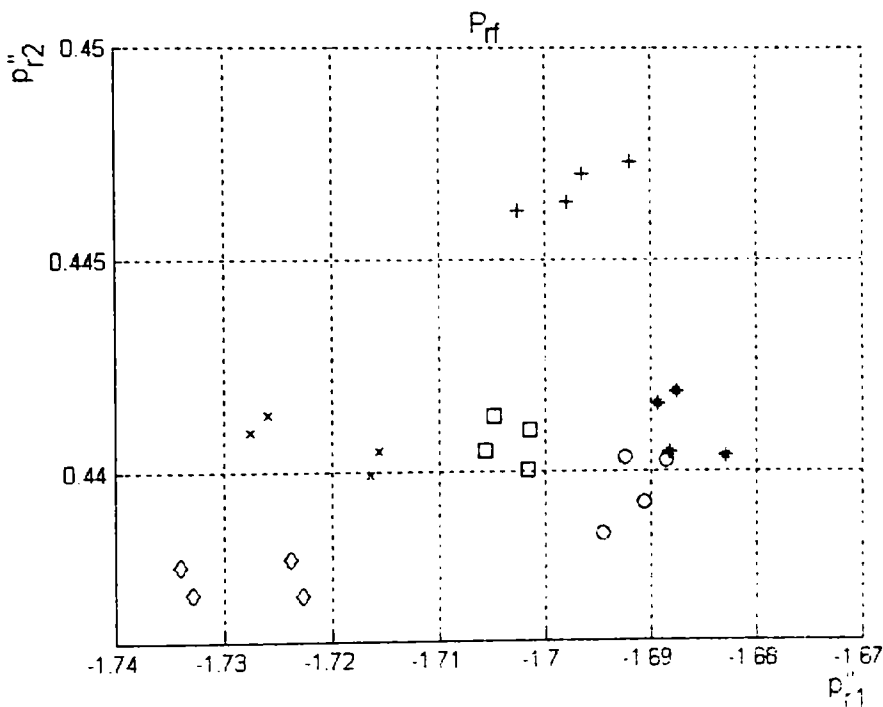


Fig.9.4.Diagrama P_{rf} pentru structuri ferito-perlitice.

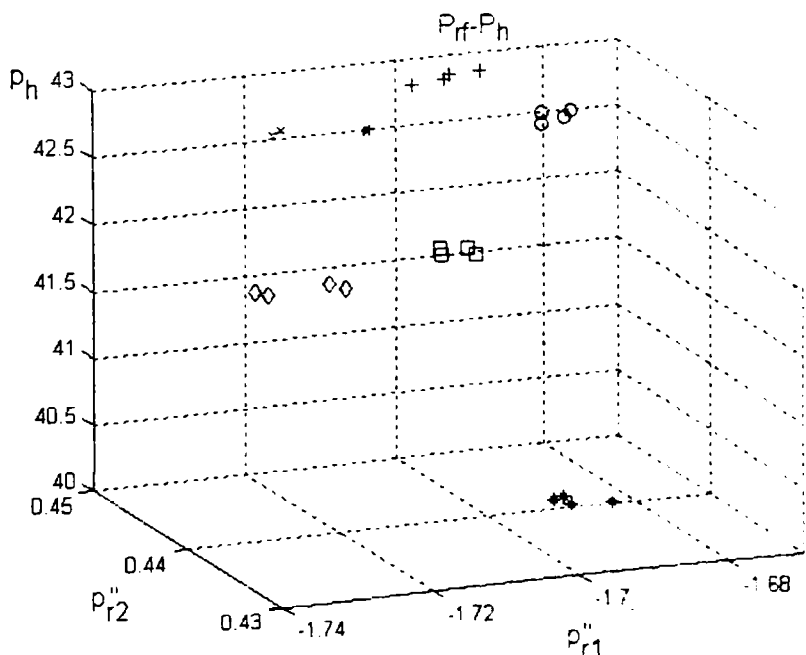


Fig.9.5.Diagrama $P_{rf}-P_h$ pentru structuri ferito-perlitice.

Deși se obține o discriminare relativ bună a microstructurilor în diagrama P_{rf} -

Ph, corelația caracteristicilor imaginii cu punctajul este nesatisfăcătoare.

Din această cauză s-a utilizat o altă caracteristică (E_{max}) calculată tot din diagrama DSP, de tipul celei prezentate în figura 9.1. calculată ca suma valorilor DSP, cu ajutorul relației :

$$E_{max} = \sum_{i_{DSP}=n}^k \sum_{j_{DSP}=m}^l DSP(i_{DSP}, j_{DSP}) \quad (9.7)$$

unde $i_{DSP}=(n...k)$ și $j_{DSP}=(m...l)$ reprezintă frecvențele spațiale în intervalele $n-k$ respectiv $m-l$. Limitarea domeniului de frecvențe spațiale corespunde definirii unui filtru trece-bandă pentru care caracteristica E_{max} este cea mai reprezentativă.

Diagrama E_{max} în funcție de punctajul structurilor ferito-perlitice în șiruri (*punctaj f-p*) (figura 9.6) pentru domeniul de frecvențe spațiale $i_{DSP}=(248...264)$, $j_{DSP}=(1...64)$, arată o corelație bună pe domenii : de la 0 la 2 respectiv de la 3 la 5.

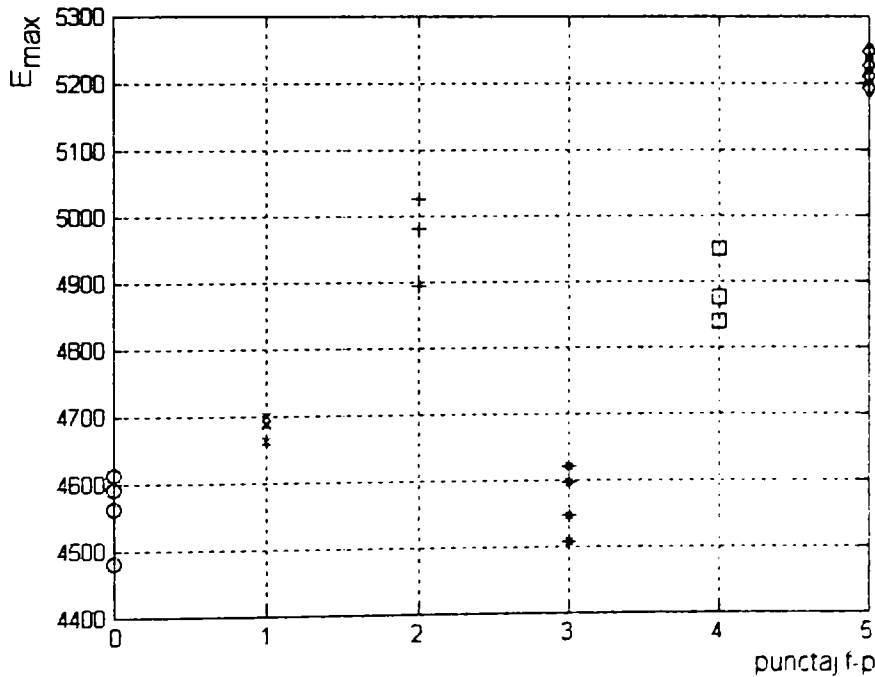


Fig.9.6. Diagrama E_{max} în funcție de punctaj f-p.

Se remarcă din diagrama $P_{rf}-P_h$ că punctele caracteristice corespunzătoare punctajelor 0-2 și respectiv 3-5 pot fi discriminate pe baza parametrului p_h , definindu-se o caracteristică derivată E'_{max} , astfel:

$$E'_{max} = \begin{cases} E_{max} \cdot p_h >= i_{hmedp} \\ E_{max} \cdot k_p \cdot p_h < i_{hmedp} \end{cases} \quad (9.8)$$

unde $i_{hmedp} = 42,5$; $k_p = 1,15$;

Reprezentând caracteristica E'_{max} în funcție de punctaj, diagrama din figura 9.7,

se constată o corelație satisfăcătoare. Pe baza valorilor determinate se poate defini o aproximare prin metoda celor mai mici pătrate cu o funcție polinomială de gradul 3:

$$E'_{\max} = ap_{fp}^3 + bp_{fp}^2 + cp_{fp} + d \quad (9.9)$$

unde cu p_{fp} s-a notat punctajul iar valorile determinate ale coeficienților sunt :
 $a = -0,0006$; $b = 0,0115$; $c = -0,0053$; $d = 0,7537$;

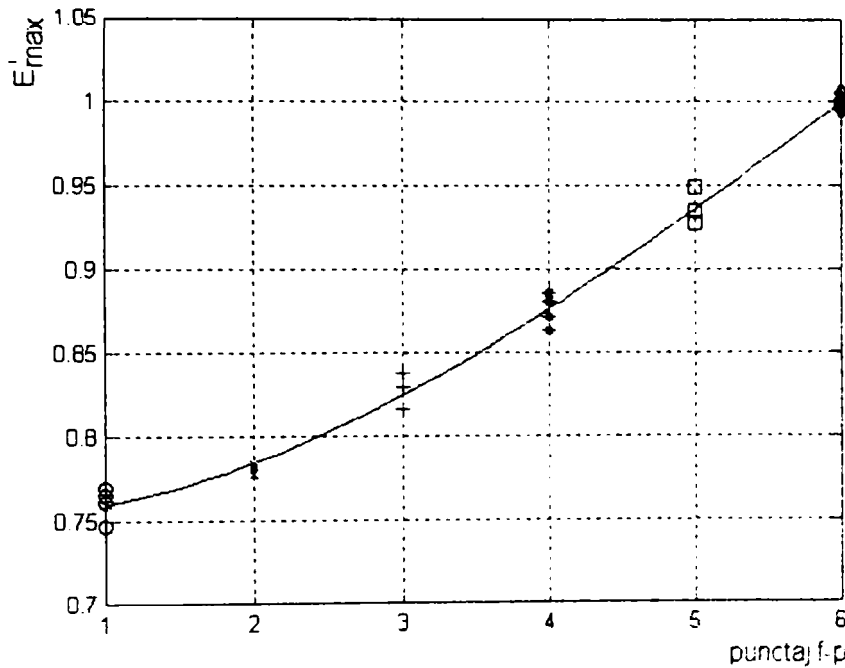


Fig.9.7. Diagrama E'_{\max} în funcție de punctaj f-p.

Codul sursă al programului ce utilizează acest algoritm și cu ajutorul căruia a fost realizată diagrama din figura 9.7 este prezentat în ANEXA 9.4.

Determinarea punctajului pentru finețea perlitei.

La determinarea punctajului pentru finețea perlitei s-a utilizat algoritmul MDSPF3D ce a dat rezultate satisfăcătoare atât în ce privește discriminarea cât și corelarea parametrului fractal p''_{rl} cu punctajul p_p (diagrama $P_{rl}-P_h$, figura 9.8), pentru parametrii $k_n = 5...64$, $i_{hmed} = 100$. Similar cazului descris anterior și aici apar două domenii distincte de corelație, ce se observă în mod evident în diagrama prezentată în figura 9.9, domeniile de punctaje 1...5 și respectiv 6...10. Cele două domenii pot fi discriminate (diagrama $P_{rl}-P_h$, figura 9.8) prin stabilirea unui prag pentru valorile parametrului p_h notat i_{hmedp} . În acest caz s-a stabilit o valoare a pragului de $i_{hmedp} = 75$. Pentru ambele domenii se pot defini aproximări cu polinoame de gradul 3 date de relația (9.10) în cazul $i_{hmed} < 75$ (figura 9.10) și respectiv (9.11) în cazul $i_{hmed} > 75$ (figura 9.11). În al doilea caz, pentru a realiza o funcție ascendentă s-a utilizat valoarea absolută a parametrului fractal p''_{rl} . În diagramele din figurile 9.9, 9.10 și 9.11 s-au reprezentat mediile pe cele patru valori caracteristice corespunzătoare unei

singure microstructuri.

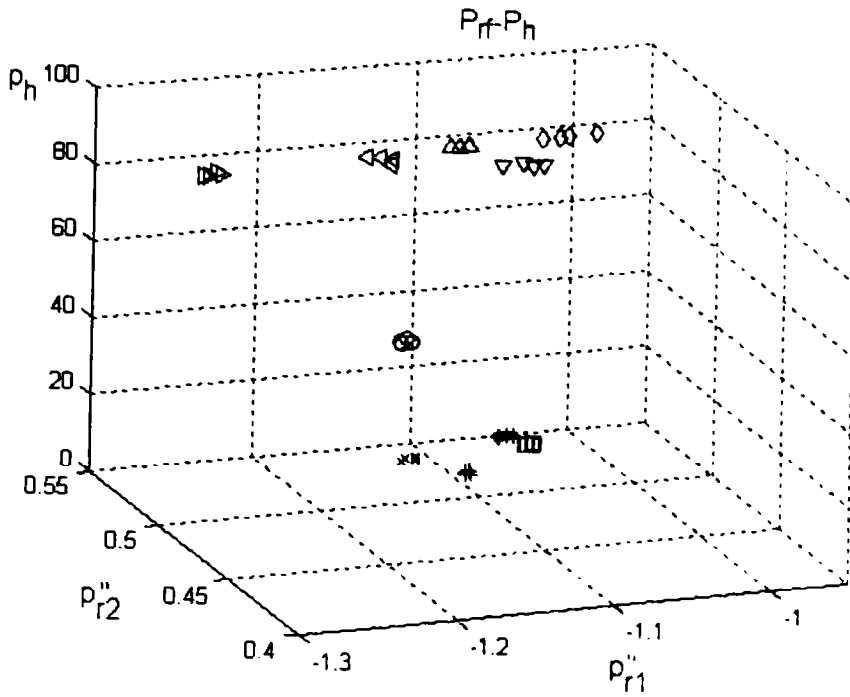


Fig.9.8.Diagrama $P_{r1}-P_h$ pentru determinarea punctajului fineței perlitei.

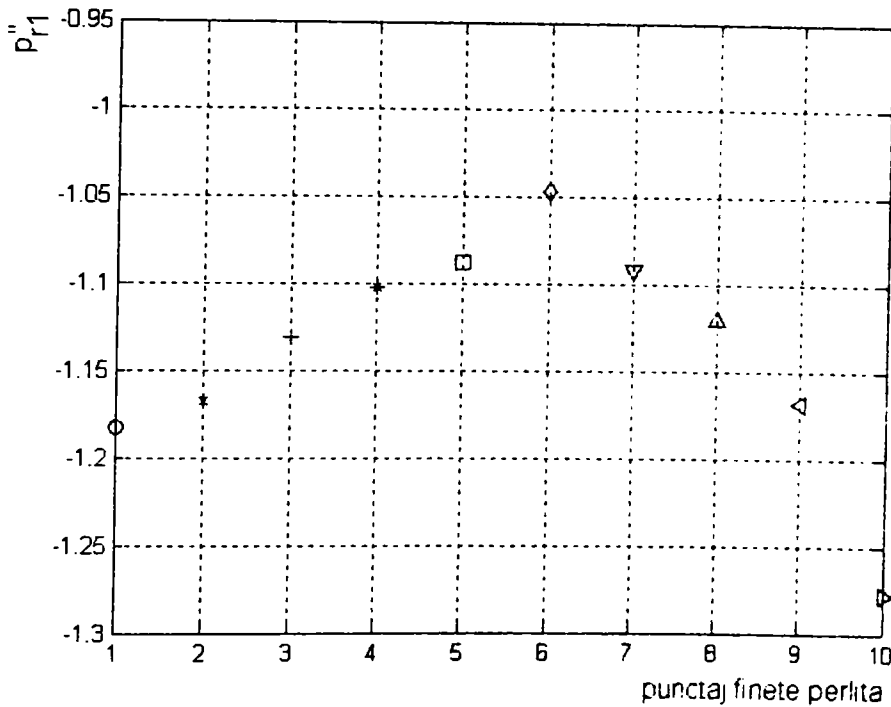


Fig.9.9.Diagrama p_{r1}'' în funcție de punctaj finețe perlita.

$$p_{r1}'' = a_1 p_{pl}^3 + b_1 p_{pl}^2 + c_1 p_{pl} + d_1 \quad (9.10)$$

unde p_{pl} reprezintă punctajul fineței perlitei pentru domeniul 1...5 iar coeficienții polinomului de aproximare sunt :

$$a_1 = -0,0300; b_1 = 0,0262; c_1 = -0,0408; d_1 = -1,1651;$$

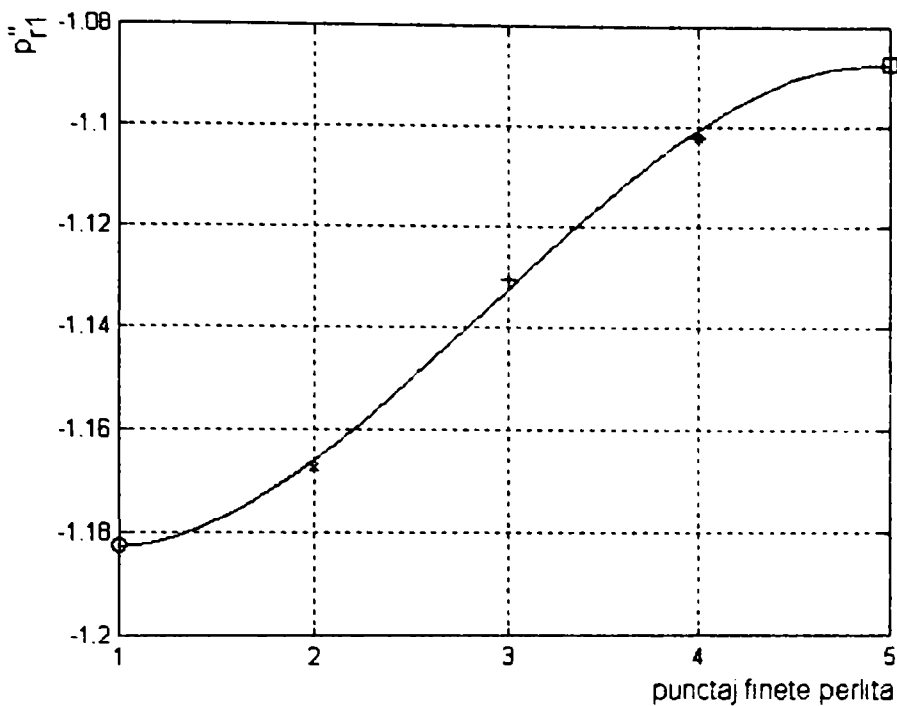


Fig.9.10. Diagrama p''_{r1} în funcție de punctaj finețe perlitică pentru $i_{hmed} < 75$.

$$|p''_{r1}| = a_2 p_{p2}^3 + b_2 p_{p2}^2 + c_2 p_{p2} + d_2 \quad (9.11)$$

unde p_{p2} reprezintă punctajul fineței perlitei pentru domeniul 6...10 iar coeficienții polinomului de aproximare sunt : $a_2 = -0,0064$; $b_2 = 0,0470$; $c_2 = -0,1411$; $d_2 = -0,9464$;

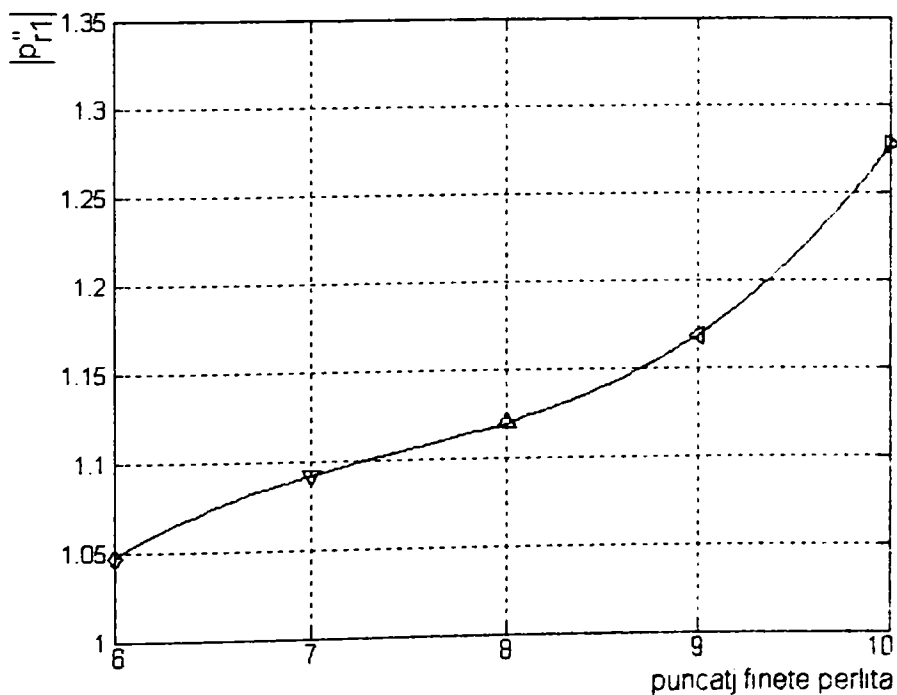


Fig.9.11. Diagrama p''_{r1} în funcție de punctaj finețe perlitică pentru $i_{hmed} > 75$.

Determinarea procentului suprafeței perlitei la fonte cu grafit nodular.

Pentru acest caz s-a utilizat deasemenea algoritmul MDSPF3D metoda de determinare a punctajului fiind similară cu cea aplicată în cazul determinării punctajului fineței perlitei. Intevalele de frecvență și mediere utilizate sunt: $k_n=5...64$, $i_{hmed} = 20$. Din diagrama $P_{rf}-P_h$ (figura 9.12) valoarea pragului de delimitarea a celor două domenii ale punctajului este $i_{hmedp} = 10$. Pentru primul domeniu de punctaj de la 2% la 20% se utilizează relația (9.12) iar pentru al doilea domeniu de punctaj de la 50% la 95% relația (9.13).

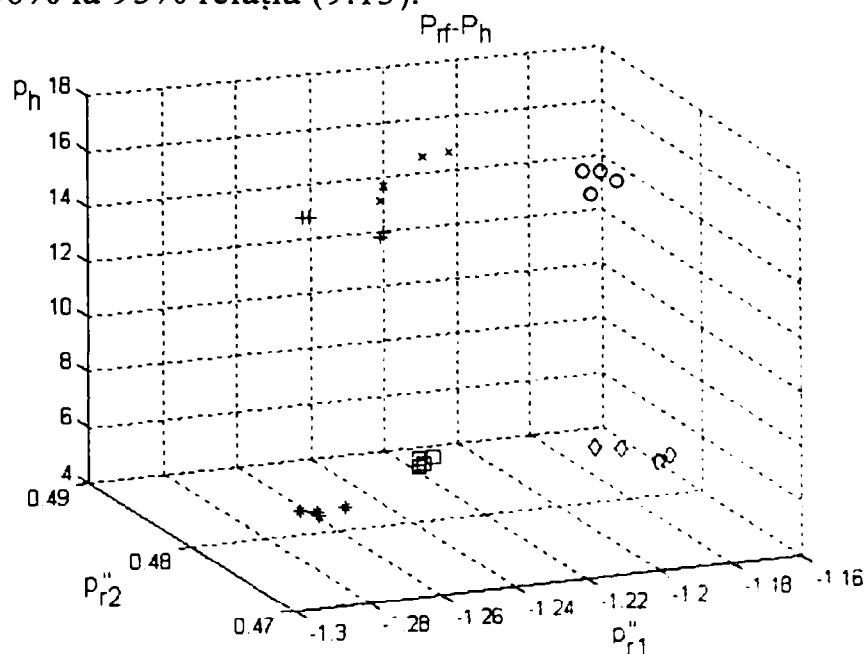


Fig.9.12. Diagrama $P_{rf}-P_h$ determinarea procentului suprafeței perlitei

$$p_{r1}'' = a_1 c_p + b_1 \quad (9.12)$$

unde c_p reprezintă procentul suprafeței perlitei pentru domeniul 6...10 iar coeficienții dreptei de aproximare sunt : $a_1 = -0,0006$; $b_1 = 0,0115$;

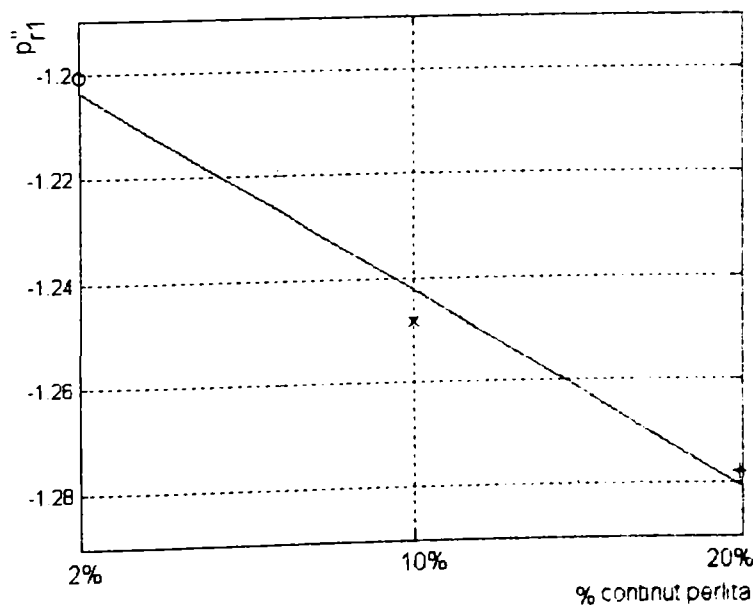


Fig.9.13. Diagrama p_{r1}'' în funcție de % conținut perlită pentru $i_{hmed} = 1...10$.

$$p''_{r1} = a_2 c_p + b_2 \quad (9.13)$$

unde c_p reprezintă procentul suprafeței perlitei pentru domeniul 50...95% iar coeficienții drepte de aproximare sunt : $a_2 = -0,0053$; $b_2 = 0,7537$;

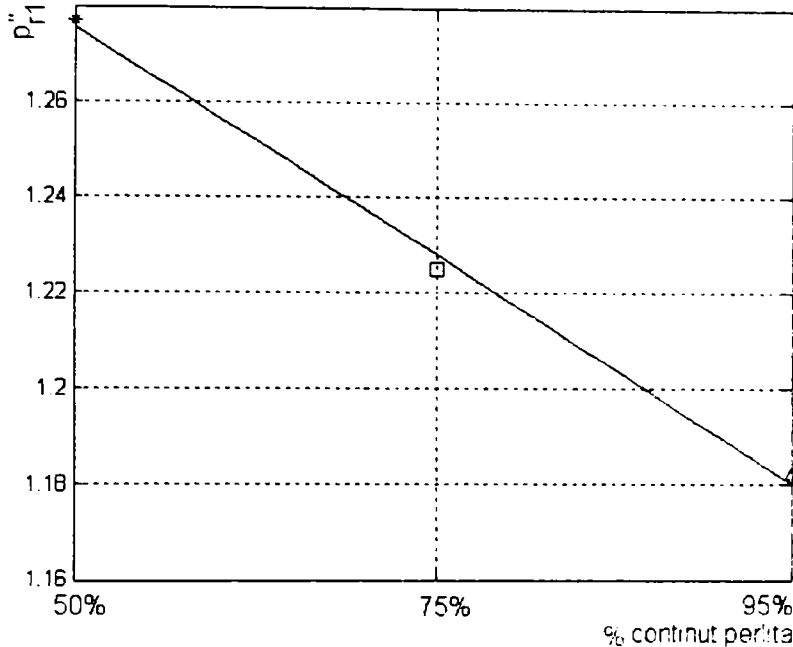


Fig.9.14. Diagrama p''_{r1} în funcție de % conținut perlita pentru $i_{hmed} = 10...20$.

Determinarea punctajului corespunzător lungimii lamelor la fonte cu grafit lamelar.

În acest caz utilizarea parametrilor fractali s-a dovedit dificilă, deși s-au încercat mai multe domenii de variație a frecvenței spațiale. experimentările au dus la rezultate nesatisfăcătoare. Discriminarea imaginilor a fost bună dar nu s-a reușit obținerea unei corelații univoce a parametrilor fractali cu punctajul lungimii lamelor de grafit.

Se observă însă că aceste tipuri de imagini sunt ușor binarizabile conținând două nivele de intensitate pregnante. Acest lucru face posibilă construirea unui algoritm la baza căruia se găsesc metode mai simple :

- se aplică un filtru de mediere pentru a micșora numărul de nivele de intensitate din imagine;
- se realizează binarizarea imaginii cu stabilirea automată a pragului de binarizare (figura 9.15.a);
- se etichetează fiecare obiect din imagini și se marchează cu o culoare diferită (figura 9.15.b);
- fiecare obiect etichetat se încadrează într-o formă eliptică la care se calculează valoarea axei mari a elipsei (LAM);
- se calculează media valorilor LAM pentru toate obiectele din imagine și se obține LAM_{med} ;
- se reprezintă grafic caracteristicile LAM_{med} în funcție de punctajul lungimii

lamelilor:

Din diagrama caracteristicilor LAM_{med} (figura 9.16) se observă că există o dependență exponențială a acestora în funcție de punctaj.



Fig.9.15. Imaginea binarizată a., și respectiv etichetată b. a imaginii reprezentate în figura A.9.5.24.

S-a utilizat o funcție de aproximare exponențială:

$$LAM_{med} = ae^{bx} \quad (9.14)$$

unde x reprezintă punctajul lungimii lamelilor de grafit iar coeficienții funcției exponențiale de aproximare sunt: $a = 0,7314$; $b = 0,8145$; $c = -2,7333$

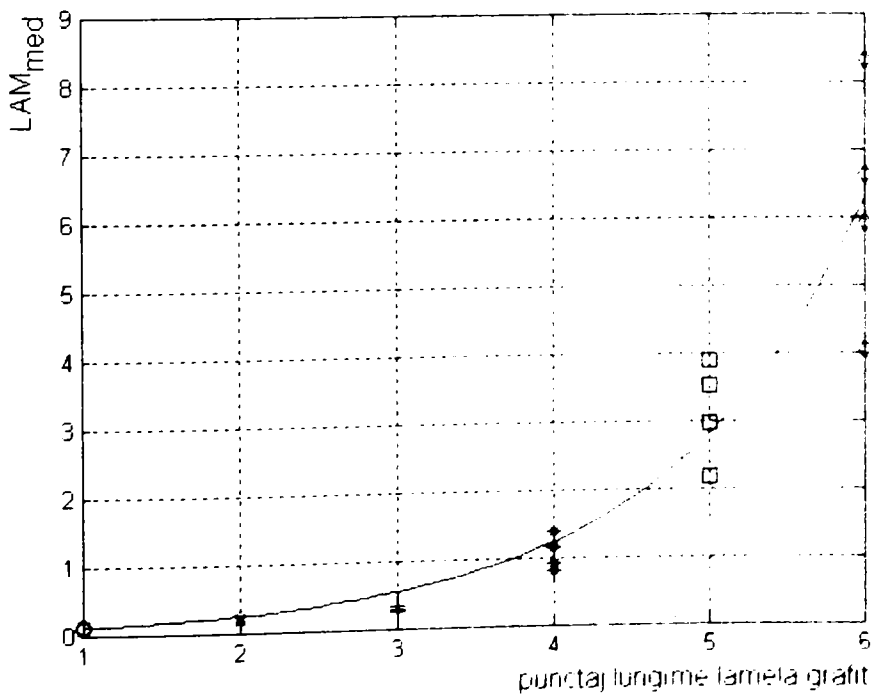


Fig.9.16. Diagrama LAM_{med} în funcție de punctaj lungime lamele grafit.

10. Concluzii și contribuții.

10.1. Concluzii finale.

Deși cele mai multe dintre laboratoarele de analiză metalografică sunt de tip clasic, în ultimul deceniu, odată cu dezvoltarea elementelor de automatizare, se folosesc din ce în ce mai multe echipamente și aparate ce înglobează asemenea elemente, în scopul ușurării muncii operatorului uman. Totuși în aceste sisteme factorul de decizie primordial rămâne omul. Majoritatea operațiilor de pregătire se realizează de către operatorul uman, neexistând echipamente sau aparate care să lucreze la un nivel de automatizare corespunzător unei celule flexibile. În ceea ce privește analiza propriu-zisă a microstructurii s-au realizat pași importanți în automatizarea acestora, dar operațiile arătate sunt permanent supervizate de către operatorul uman care participă la toate fazele de decizie.

În principiu, achiziția și prelucrarea imaginilor metalografice parcurge etape similare cu cele realizate în vederea oricărei aplicații de acest gen (recunoașterea formelor sau conducere vizuală).

Particularități importante derivă din faptul că în acest caz se lucrează cu imagini achiziționate prin intermediul unui microscop și că aceste imagini nu reprezintă obiecte ci structuri abstracte numite texturi.

Analiza texturilor se realizează printr-o mare varietate de metode specifice aplicației concrete. Din această cauză alegerea unei anumite metode în scopul analizei rugozității suprafețelor sau a microstructurilor metalografice necesită un important volum de experimentări.

Din analiza caracteristicilor laboratoarelor automatizate, rezultă că pentru laboratoarele metalografice se pretează în primul rând conceptele modulare ce permit o flexibilitate ridicată. Pe de altă parte o serie de proceduri și instalații utilizate în domeniul bio-medical pot fi luate ca model pentru rezolvarea problemelor în domeniul metalografic.

În urma experimentărilor realizate cu ajutorul celulei flexibile, s-a demonstrat că utilizarea obiectelor virtuale în cadrul programelor de comandă, utilizarea programelor de aplicație distribuite și a algoritmului fuzzy de stabilire a priorităților operative, duce la ridicarea flexibilității.

În urma analizei rezultatelor simulării și a funcționării sistemului de șlefuire ce utilizează logica fuzzy și rețele neuronale artificiale, s-a desprins concluzia că regulatorul fuzzy folosit necesită ajustări, în continuare, pentru mărirea robusteții reglajului. Deși, pe anumite domenii de lucru reglajul este acceptabil, se impune îmbunătățirea regulatorului pentru mărirea robusteții acestuia.

Studiind cercetările recente în domeniul analizei texturilor și a prelucrării de imagini într-o primă fază s-au încercat mai multe metode ce fac parte din grupa metodelor statistice (co-ocurență, momente statistice de diferite ordine) și frecvențiale (filtre Gabor, wavelet) care însă, pentru setul de imagini ce reprezintă

rugozități ale suprafețelor prelucrate, au produs rezultate nesatisfăcătoare. În a doua fază s-au încercat metode fractale ce au produs rezultate mai bune, din care cauză s-a optat pentru utilizarea acestora pentru experimentări mai aprofundate. Utilizând metode fractale s-a reușit elaborarea unor algoritmi ce pot să realizeze discriminarea diverselor tipuri de imagini.

În urma încercărilor s-a stabilit că metoda fractală ce utilizează densitatea spectrală de putere este aproximativ la fel de rapidă ca și metoda numărării casetelor dar precizia de orientare este mai mică. Metoda ce utilizează convoluția cu filtre Gabor este mai precisă dar există o serie de parametri a căror determinare este dificilă și este mai lentă decât celelalte două metode.

Pentru imaginile suprafețelor eșantioanelor, diagramele P_f (pentru ambele mărimi: $\times 10$ și $\times 25$) stau la baza alcătuirii unui algoritm euristic pe baza căruia să se poată decide apartenența unei epruvete la o anumită grupă.

O observație importantă este aceea că pe baza caracterizării fractale se poate dezvolta un algoritm fuzzy care să realizeze decizia de stabilire a rugozității.

În cazul imaginilor ce reprezintă microstructuri metalografice este util să se utilizeze parametri de intensitate și energetici împreună cu parametrii fractali.

Algoritmii de analiză a imaginilor prezentați pot fi utilizați și în alte domenii decât analiza rugozității sau a microstructurilor metalografice, în aplicații biomedicale sau nano-structuri.

10.2. Contribuții

10.2.1. Contribuții originale.

- a. Dezvoltarea unui nou algoritm de analiză a imaginilor ce reprezintă rugozitatea suprafețelor prelucrate prin așchiere, utilizând metode fractale.
- b. Dezvoltarea unui nou algoritm de analiză a imaginilor ce reprezintă microstructuri metalografice, utilizând metode fractale și de analiză frecvențială cât și parametrii statistici.
- c. Conceperea unor diagrame specifice (diagrame P_f) pentru analiza fractală a imaginilor texturate, prin definirea unor noi parametri fractali : p_2 , p_3 , μ_x , μ_y .

10.2.2. Contribuții personale.

- a. Sintetizarea cunoștințelor prezentate în sursele bibliografice în domeniul:
 - organizării laboratoarelor de analiză metalografică, a aparaturii de laborator și a automatizării fazelor de prelucrare și analiză din laboratoarele metalografice;

- celulelor flexibile și cu precădere a celulelor flexibile pentru laboratoare;
- conducerii și comenzii roboților utilizând logica fuzzy și rețele neuronale artificiale;
- achiziției computerizate a imaginilor digitale;
- metodelor de definire și stabilire a rugozității suprafețelor prelucrate prin așchiere;
- prelucrării imaginilor în general utilizând diferite metode : contrastare, filtrare, segmentare, transformări integrale;
- prelucrării imaginilor reprezentând texturi prin metode: statistice, wavelet și fractale;

b. Conceperea și realizarea unei celule flexibile de analiză metalografică, utilizând două variante ale sistemului de comandă.

c. Conceperea și realizarea automatizării unui microscop de analiză metalografică, cu comandă computerizată, pentru deplasarea mesei microscopului cu motoare pas cu pas și cameră video digitală pentru achiziția de imagini.

d. Conceperea și realizarea programelor de comanda a mesei microscopului metalografic, pentru micro-calculatorul BlueEarth 485 și a programelor de transmisii de date.

e. Conceperea și realizarea unui dispozitiv de prehensiune, pentru robotul AIDH5, a epruvetelor metalografice în vederea șlefuirii.

f. Conceperea și realizarea programelor de transmisie de date, sub mediul MATLAB, între un calculator PC și echipamentul de comandă CNC-EN5.

g. Conceperea și realizarea programelor de achiziție de date, utilizând mediul de programare MATLAB și modulul de achiziție de date PCI 6023E.

h. Conceperea și realizarea unor programe de transmitere a informațiilor între calculatoare PC, pe rețea LAN Windows, sub mediul MATLAB.

i. Conceperea și realizarea programelor de comandă a celulei flexibile, pentru:

- preprocesarea programelor de aplicație;
- utilizarea și implementarea conceptului obiectelor virtuale;
- vizualizarea fluxului tehnologic;

j. Modelarea și simularea unui sistem de șlefuire a epruvetelor metalografice, utilizând algoritmi de comandă bazați pe logica fuzzy și rețele neuronale

artificiale.

k. Modelarea și simularea unui algoritm fuzzy pentru luarea deciziilor referitor la prioritatea de prelucrare a epruvetelor metalografice, potrivit unor criterii impuse.

l. Implementarea unor programe de analiză a imaginilor ce reprezintă:

- rugozitatea suprafețelor, utilizând algoritmi fractali;
- microstructuri metalografice, utilizând algoritmi fractali și metode frecvențiale;

Bibliografie

- [AMANN1] Amann P., Perronne J.,M., Gissinger G.,L., Frank P., M., Identification of fuzzy relational models for fault detection, *Control Engineering Practice* 9 (2001)555.
- [ARIMOTO1] Arimoto S., Nguyen P.,T.,A., Naniwa T., Learning of robot tasks on the basis of passivity and impedance concepts, *Robotics and Autonomous Systems* 32 (2000) 79–87.
- [ASTROM1] Astrom K.J.A., Agglund T.H., The future of PID control, *Control Engineering Practice* 9 (2001)1163-175.
- [BALEIX1] S., Baleix, S., Le Roux, G., Bernhart, P., Luors - Surface and image analysis of oxides grown and spalled on heat resistant cast steels exposed on thermal cycles, Ecole des Mines d'Albi-Carmaux, Albi, France, *Journal of Materials processing Technology*, Elsevier Science BV, 2001.
- [BARONI1] Baroni P., Fogli D., Modeling robot cognitive activity through active mental entities, *Robotics and Autonomous Systems* 30 (2000) 325–349.
- [BARABAS1] Barabás, T. **Vesselényi, T., Using Matlab for remote control of industrial robots through LAN type Ethernet under TCP/IP**, *Robotica & Management, International Journal*, vol 9, No.2., December, 2004
- [BARABAS2] Barabás T., **Vesselényi T., Conducerea și programarea roboților industriali** – ISBN 973-613-497-0, Universitatea din Oradea 2004.
- [BARABAS3] – Teza de doctorat, Universitatea “Politehnica Timișoara”, 2003.
- [BARABAS4] – Barabás T., **Vesselényi T., On the Control Structure of a Flexible Manufacturing Cell Developed at the University of Oradea**, *Revista „Robotica & Management”*, vol.9, No. 1, pag. 31-36. June 2004, .ISSN 1453-2069, Edited By Romanian Society for Industrial Robotics, the Caraș-Severin County Council and the „Eftimie Murgu” University of Reșița;
- [BARRALIS] Barralis, J., Maeder, G., *Metalurgie/Elaboration, structures-proprietes normalisation*, collection Les Precip AFNOR/ NATHAM, Edition Natham, 1997, ISBN Natham: 2-09-177491-X; ISBN AFNOR : 2-12-260121-6.
- [BASANTA1] D., Basanta, P.J., Bentley, M.A., Miodownik, E.A., Holm - *Evolving Cellular Automata to Grow Microstructures*, Kings Collage, University Collage, London, Sandia National Laboratories, New Mexico, USA
- [BELAK1] J., Belak, D.B., Boerker, J., Glosly – *Molecular Dynamics Simulation of High Strain-Rate Void Nucleation Nanocrystalline Copper*, Lawrence Livermore National Laboratory, Conference on Materials theory – Dynamics, Microstructure, Fracture. University of California, august –2003.
- [BERAHA1] Beraha, E., and Shpigler E., *Color Metallography*, American Society for Metals, Metals Park, Ohio, 1977.
- [BERAHA2] Beraha, E., *Metallographic Reagents Based on Sulfide Films*, *Praktische Metallographie*, Vol 7, 1970, p 242-248.
- [BERAHA3] Beraha, E., *Metallographic Reagents Based on Molybdate Solutions*,

Praktische Metallographie, Vol 11, 1974, p 271-275.

[BHALERAO1] A., Bhalerao, R., Wilson – Unsupervised image segmentation combining region and boundary estimation, Image and Vision Computing, Elsevier Science B.V. 2000.

[BLAGA] Florin Blaga, Radu Țarcă, **Vesselenyi Tiberiu**, Ioan Țarcă, **About fuzzy coloured Petri nets in modelling flexible manufacturing systems**, MICROCAD 2003, Misckolc, Hungary, feb. 2003, ISBN;

[BRINKMANN1] S., Brinkmann, H., Bodschwinn, H.W., Lemke - Accessing roughness in three-dimensions using Gaussian, regression filtering, University of Hanover, Germany, Intenational Journal of Machine tools & Manufacture, Elsevier Science Ltd, 2001

[BRUMBY1] S.P., Brumby, N.R., Harvey, S., Perkins, R.B., Porter, J. J., Szymanski, J., Theiler, J.J. Bloch - A genetic algorithm for combining new and exciting image processing tools for multispectral imagery. Los Alamos National Laboratory, New Mexico, U.S.A.

[BUEHLER] Buehler Ltd. 41 Waukegan Rd. Lake Bluff, Il. 60044.

[BUDAU] V., Budău, C.M., Crăciunescu, Studiul materialelor, Ghid individual pentru lucrări de laborator, Editura Mirton Timișoara, 1998.

[BUI1] S.H., Bui, V., Gopalan, J., Raja - An internet based surface texture information system Mechanical Engineering and Engineering Science, Charlotte, USA, Intenational Journal of Machine tools & Manufacture, Elsevier Science Ltd. 2001.

[BURDET1] Burdet E., Rey L., Codourey A., A trivial and efficient learning method for motion and force control, Engineering Applications of Artificial Intelligence 14 (2001)487-496.

[CARVALHO] Carvalho L.M.V., Silva Dias M.A.F., - An Application of Fractal Box Dimension to the recognition of Mesoscale Cloud Patterns in Infrared Satellite Images - Department of Atmospheric Sciences, Institute of Astronomy and Geophysics, University of Sao Paulo, Brazil, 1998.

[CASTELMAN] Castleman, K.R. Digital Image Processing, Prentice Hall, Englewood Cliffs, NJ, 1996.

[CHANG1] C., Y., Chang, P., C., Chung - Medical image segmentation using a contextual-constraint-based Hopfield neural cube, Shu-Te University, Cheng Kung University, Taiwan, Image and Vision Computing, Elsevier Science 2001.

[CHANG1] Chang C.A., Angkasith V., Using Hopfield neural networks for operational sequencing for prismatic parts on NC machines, Engineering Applications of Artificial Intelligence 14 (2001)357-368.

[CHEN] M.M., Chen, J.A., Fairwather, S.A., Green, EDUMECH. Mechatronic Instructional Systems. Case Study: Pneumatics Systems, Production of Shandor Motion Systems, Inc, 1999.

[CHIZHIK1] S. A., Chizhik, N. B., Gaiduk, N. K., Myshkin – Computer simulation of precision contact with account of microgeometrical and mechanical heterogeneity

of surfaces, Elsevier Science, 2001.

[CIMI1] ***CIMI Laboratory, W., Ruoyun, Image Processing for Prostate Ultrasound Images, www.cimi.com 2002.

[CLARCK] Clarck A., Thomas B.T., Campbell N.W., Greenway P., - Texture Deconvolution for the Fourier Based Analysis of Non-Rectangular Regions - Advanced Computing Research Centre, University of Bristol, UK. Angus.Clarck@bristol.ac.uk, BMVC99, 1999.

[CLEMEX1] ***Clemex Vision, Clemex Canada, www.clemex.com 2003.

[CMIS1] ***CSIRO, Mathematical and Information Sciences, CMIS Research, Image Analysis, www.csiro.comm 2003.

[COCQUEREZ] Cocquerez, J.P., Philipp, S. Analyse d'images: filtrage et segmentation, Masson, Paris, 1995

[CORKE] Corke, P.I., Visual control of robots high performance visual servoing, Research Studies Press LTD, 1996

[COSTA] Costa M., A., - Fractal Description of Rough Surfaces for Haptic Display – PhD Thesis, Stanford University, december 2000.

[CROWLY] Crowley J.L., Pourraz F. Continuity properties of the apperance manifold for mobile robot position estimation, Image and Vision Computing, 2000.

[CSR] Catalogul Standardelor Române – SR ISO 4287/1 : 1993, Standarde Rugozitate, Editat Indaco systems S.R.L.

[CULA1] Cula O.G., Dana K.J., - 3D Texture Recognition Using Bidirectional Feature Histograms – International Journal of Computer Vision 59(1), 33-60, 2004.

[CULA2] Cula O.G., Dana K.J., - Recognition Methods for 3D Textured Surfaces – Proceedings of Spie Conference on Human Vision and Electronic Imaging VI, San Jose, January 2001.

[DAVID] David, R., Alla, H., Du Grafctet aux réseaux de Petri. Ed. Hermes, Paris, 1992.

[DAVIS1] G.M., Davis, A., Nosratinia - Wavelet-based Image Coding: An Overview, www.geoffreydavis.com , 2001

[DEBACKER1] S., De Baker, P., Scheunders - Texture Segmentation by frequency-sensitive elliptical competitive learning, Image and Vision Computing, Elsevier Science Ltd. 2001.

[DONG1] Y., Dong, G.,R., Hillman – Three dimensional reconstruction of irregular shapes based on fitted mesh of contours. Image and Vision Computing, Elsevier Science, 2001.

[DOUGHERTY] Dougherty E.R., Giardina, C.R., Image Processing, Continous to Discrete, vol. 1, Geometric, Trasform and Statistical Methods, Prentice Hall Inc., Englewood Cliffs, 1987.

[DRAGU] Dragu D., Bădescu Gh., Sturzu A., Militaru C., Popescu I., - Toleranțe și măsurători tehnice – E.D.P. București 1980.

[DUMIT1] Dumitrescu D., Costin H., Rețele neuronale artificiale – Teorie și aplicații, Ed. Teora,1996.

- [ELSHOLZ] Elsholz F., - Rough surface characterization and determination of the rms roughness from coherent light scattering – Technische Universität, Berlin, september, 21, 2004.
- [EMAMI1] Emami M.R., Goldenberg A.A., Burhan T.R., Systematic design and analysis of fuzzy-logic control and application to robotics, Part I. Modeling, Robotics and Autonomous Systems 33 (2000) 65–88.
- [EMAMI2] Emami M.R., Goldenberg A.A., Burhan T.R., Systematic design and analysis of fuzzy-logic control and application to robotics, Part II. Control Robotics and Autonomous Systems 33 (2000) 89–108.
- [FARDIN] Fardin N., Stephansson O., Jing L., - The scale dependence of rock joint surface roughness – International Journal of Rock Mechanics and Mining Sciences, 38, 659-669, 2001.
- [FOSTER1] B., Foster, B., Fookes, M., Hunt - Image analysis for Materials Science, www.metallography.com 1996.
- [FRANCOIS] Francois A.R.J., Medioni G.G., Interactive 3D model extraction from a single image, Image and Vision Computing, 2000.
- [FUJITA] Fujita K., Nayar S.K., - Recognition of Dynamic Textures using Impulse Response of State Variables - Department of Aeronautics, Kyushu University, fuji@aerokyushu-u.ac.jp. 2004.
- [GACSADI1] A., Gacsadi, V., Tiponut, - Rețele neuronale celulare, aplicații, Editura Universității din Oradea, 2002.
- [GAUTIER1] Gautier M., Poignet P., Extended Kalman filtering and weighted least squares dynamic identification of robot, Control Engineering Practice 9 (2001)1361-372.
- [GIACINTO1] G., Giacinto, F., Roli - Design of effective neural network ensembles for image classification purposes University of Cagliari, Italy, Image and Vision Computing, Elsevier Science B.V. 2001.
- [GONZALES] Gonzales, R.C., Woods, R.E., Digital Image Processing, Addison Wesley, Reading MA, Publishing Comp, 1993.
- [GOTTSTEIN1] G., Gottstein, R., Sebald - Modelling of recrystallization textures, Institut für Metallkunde und Metalphysik, Aachen, Germany, Journal of Materials Processing Technology, Elsevier Science B.V. 2001.
- [HAMZE] Hamze F., Clark J.J. View based route learning with self Organizing neural networks, Image and Vision Computing, 2000.
- [HARALICK1] Haralick, R.M., Shapiro, L.G., “Glossary of Computer Vision Terms”, în Pattern Recognition, vol 24, no. 1, pag. 6993, 1991
- [HARALICK2] Haralick, R.M., Sternberg, S.R., Zhuang, X., “Image Analysis using Mathematical Morphology”, în IEEE Trans. On Pattern Analysis and Machine Intelligence, vol. 9, no. 4, Iulie 1987, pag. 532-549
- [HARBICK1] Harbick K., Sukhatme S., Speed Control of a Pneumatic Monopod using a Neural Network, INTERNET (2002).
- [HILTNER1] J., Hiltner, M., Fahti, B., Reusch – An approach to use linguistic and

- model-based fuzzy expert knowledge for the analysis of MRT images, Image and Vision Computing, Elsevier Science B.V. 2001.
- [JAIN] A.K., Jain, Fundamentals of digital images processing, Prentice Hall Englewood Cliffs, 1989.
- [JAHNE] Jahne, B. Practical Handbook on Image Processing, Prentice Hall, Englewood Cliffs NJ, 1989
- [JAIN] Jain, A.K., Fundamentals of Digital Image Processing, Prentice Hall, Englewood Cliffs NJ, 1989
- [JANG1] Jang, R., MATLAB – Fuzzy Toolbox- The MathWorks, Inc. Revision: 1.12 Date: 2000, 15.
- [JIANG1] Z., Jiang, H., Wang, B., Fei - Research into the application of fractal geometry in characterising machined surfaces, Xi'an Jiaotong University, People's Republic of China, International Journal of Machine tools & Manufacture, Elsevier Science Ltd, 2001.
- [JIANG2] X.Q., Jiang, L., Blunt, K.J., Stout - Lifting wavelet for three-dimensional surface analysis, University of Huddersfield, UK, International Journal of Machine tools & Manufacture, Elsevier Science Ltd, 2001.
- [JOSSO1] B., Josso, D.R., Burton, M.J., Lalor - Wavelet strategy for surface roughness analysis and characterisation, Liverpool Jhon Moores University, UK, Computer Methods in Applied Mechanics and Engineering, Elsevier Science, 2001.
- [JOURNAUX1] S., Journaux, P., Gouton, G., Thauvin - Evaluating Creep in metals by grain boundary extraction using directional wavelets and mathematical morphology Universite de Bourgogne, France, ALSTOM energy and Serma Technologies, Belfort, France, Journal of Materials Processing Technology. Elsevier Science B.V. 2001.
- [KARKANIS] Karkanis S.A., Magoulas, G.D., Iakovidis D., Maroulis D.E., Schurr M.O., - On the Importance of Feature Descriptors for the Characterisation of Texture - University of Athens, Department of Informatics, TYPA Build., 15784 Athens, Greece, sk@di.uoa.gr. 2004.
- [KARLIK1] Karlik B., Aydin S., An improved approach to the solution of inverse kinematics problems for robot manipulators, Engineering Applications of Artificial Intelligence 13 (2000) 159-164.
- [KEHL] Kehl, G.L., Metlay, M., The Mechanism of Metallographic Etching, Journal of the Electrochemical Society, Vol 101, March 1954, p 124-127.
- [KOVACS1] Kovacs, Fr., Túsz, F., Varga, Șt., Fabrica viitorului, Ed. Multimedia Internațional, Arad, 1999.
- [KOVACS2] Kovacs, Fr., Țarcă, R., Blaga, F., Tripe, V., Sisteme de fabricație flexibilă, Ed. Univ. Oradea, 1999.
- [LACHTINE] Lachtine, I., Metalographie et traitements thermiques des metaux, quatrieme edition, Edition MIR- Moscou, 1986.
- [LAU1] Lau H.C.W., Wong T.T., Ning A., Incorporating machine intelligence in a parameter-based control system: a neural-fuzzy approach, Artificial Intelligence in

Engineering, 15 (2001), 253-264.

[LEACH1] R.K., Leach - NanoSurf IV: traceable measurement of surface texture at the National Physical Laboratory, UK National Physical Laboratory, Middlesex, UK, International Journal of Machine tools & Manufacture, Elsevier Science Ltd, 2001

[LEE1] B.Y., Lee, Y.S., Tarng - Surface roughness inspection by computer vision in turning operations National Huwei Institute of Technology, National Taiwan University of Science and Technology, International Journal of Machine tools & Manufacture, Elsevier Science Ltd, 2001.

[LEU] Jia Guu Leu On indexing the periodicity of image textures, Image and Vision Computing, 2000.

[LIENARD] Lienard, P., Pacque, C., Analysis of the Selective Coloring Mechanism for Identification of Different Phases in Al-Si-Cu Foundry Alloys, Hommes Fonderie, Vol 126, June-July 1982, p 27-35.

[LIFESPAN1] ***LifeSpan BioSciences - Drug Target, CMIS Research - Image Analysis August 04, 2003, www.lifespan.com

[LU1] S., Lu, Y., Gao, T., Xie, F., Xie, X., Q., Jiang, Z., Li, F., Wang - A novel contact/non-contact hybrid measurement system for surface topography characterization Huazhong, Hong Kong University of science and Technology, Second Research Institute of the Public Security Ministry, China, International Journal of Machine tools & Manufacture, Elsevier Science Ltd, 2001

[LUCIA1] ***Lucia Spectral-system for spectrometry and colorimetry, Universal Imaging Corporation - Device Control & System Automation , www.lucia.com 2004

[MACGILLAVRY] MacGillavry, E., - Textural pattern recognition - Department of Cartography Faculty of Geographical Sciences University of Utrecht, DIP5, final project, november, 1998.

[MARCHAND] Marchand, E., Bouthemz, P., Chaumette, F., A 2D, 3D modelbased approach to realtime visual tracking, Image and Vision Computing, 2000.

[MARX1] V., Marx, D., Raabe, G., Gottstein - Simulation of Recrystallization using a 3D Cellular Automaton Approach, Aachen, Germany, Conference on Materials Theory – Dynamics, Microstructure, Fracture. University of California, 2003.

[MATERKA] Materka A., Strzelecki M., - Texture Analysis Methods – A Review - Technical University of Lodz, Institute of Electronics, COST B11 report, Brussels, 1998.

[MATLAB1] ***MATHWORKS – MatlabR12 –Programing Guide, 2000.

[MATLAB2] .*** Mediu de programare MATLAB* V 6.0.0.88.R12, Licență Universitatea din Oradea.

[MESER] Messer K., de Ridder D., Kittler J.,- Adaptive Texture representation methods for automatic target recognition - Centre for Vision, Speech and Signal Processing, Dept. of Electronical and Electrical Eng., University of Surrey, Guildford, The Netherlands K.Messer@surrey.ac.uk, BMVC99, 1999.

[METADEX] ***METADEX, CSA, www.csa.com 2004.

[METALLOROM] ***MetalloROM, HDH Thermal, Dr. Sommer Werkstofftechnik,

www.hbthermal.com, 2004.

[METALHANDBOOK] ***Metal Handbook, Ninth Edition, vol. IX, Metallography and Microstructure, 1995.

[MITELEA] Mitelea, I., Lugscheider, L., Tillmann, W., Știința materialelor în construcția de mașini, Editura Sudura, Timișoara 1999, ISBN 973-98968-6-3.

[MMS] ***International Standard ISO/FDIS 9506-1-2:2002 Industrial automation systems —Manufacturing Message Specification.

[MOGA] Moga, I., Brîndeu, L., Polojîntef-Corbu, N., **Vesselenyi, T., Computer model of tool path and surface roughness in axial vibratory turning**, Buletinul Științific al Univ. „Politehnica” Timișoara; (Journal of Mechanical Engineering); Tom 47 (61); 2002; Fascicula 2; ISSN 1244.

[MOKHTARIAN] Mokhtarian F., Khalili N., Yuen P. Multi scale free form 3D object recognition using 3D models, Image and Vision Computing, 2000.

[MORPHO1] ITK Kitware Inc. – NLM Insight Segmentation & Registration Toolkit, Morphogenesis. www.insight.com

[MUDURA1] Mudura P. Ungur, P. Vesselenyi T. Pop T. M. Moga I., Steel quenching liquid cooling capacity definition by means of measure “M”. Quality and reliability of machines. International Machinery Faire 2002 Nitra, 29.5-30.5. 2002., SPU v Nitre, 2002, ISBN 80-8069-034-0 138-140.

[MUDURA2] Mudura, P., **Vesselenyi, T., Cuc, C., Prelucrarea asistată de calculator a datelor achiziționate în cazul răcirilor rapide**, Analele Universității din Oradea, fascicula Mecanică, secțiunea Tehnologii în Construcția de Mașini. 1998.

[MUDURA3] Mudura, P., **Vesselenyi, T., Computer program for temperature variation calculus in a cylindrical face cooled probe**, în Lucrările celei de-a 5-a Conferințe Internaționale de ELEMENTE FINITE și de FRONTIERĂ. Universitatea din Oradea , România, 25-27 mai 2000.

[MUDURA4] Mudura, P., **Vesselenyi, T., Thermal diffusivity computation software for metals**, în Lucrările celei de-a 5-a Conferințe Internaționale de ELEMENTE FINITE și de FRONTIERĂ, Universitatea din Oradea , România, 25-27 mai 2000.

[MUDURA5] Mudura, P., **Vesselenyi, T., Contribuții privind automatizarea testării lichidelor de răcire**, în: „ROBOTICA la sfârșit de mileniu”, Lucrările celui de-al XV-lea Simpozion Național cu participare internațională. „ROBOTICĂ 2000”, ISBN 973-8083-67-2, Oradea-Băile Felix, România, 12-14 octombrie 2000, Editura Universității din Oradea.

[MUDURA6] Mudura, P., **Vesselenyi, T., TĂTARU, B., Analiza microstructurilor folosind FFT**, în: „ROBOTICA la sfârșit de mileniu”, Lucrările celui de-al XV-lea Simpozion Național cu participare internațională, „ROBOTICĂ 2000”, ISBN 973-8083-67-2, Oradea-Băile Felix. România, 12-14 octombrie 2000, Editura Universității din Oradea.

[MUDURA7] Mudura, P., **Vesselenyi, T., Țarcă, I., Țarcă, R., Metodă de recunoaștere a legii de distribuție a valorii experimentale**, comunicată: Al IX-lea

Congres Academia de Cibernetică“ Ștefan Odobleja“ –Tendințe actuale în cibernetică și filozofia științei , 17-19 oct. 1996, Oradea -Băile Felix.

[MUKHERJEE1] S., Mukherjee, B., R., Cooper, Coarsening in the presence of vacancies, Conference on Materials theory – Dynamics, Microstructure, Fracture. University of California, 2003.

[NAKANO1] T., Nakano, T., Morie, M., Nagata, A., Iwata - A Cellular–Automaton Type Image Extraction Algorithm and its Implementation Using an FPGA, Hiroshima University, Kyushu Institute of Technology, Kobe University, Japan.

[NI1] ***NATIONAL INSTRUMENTS – PCI1200 – User’s Guide, 1999.

[NI2] ***NATIONAL INSTRUMENTS – Product Catalog 2004.

[NOVAK1] Novakovic, B., Scap, D., Novakovic D., An analytic approach to fuzzy robot control synthesis, Engineering Applications of Artificial Intelligence 13 (2000) 71-83.

[OLIVER] Oliver D., - Fractali – Editura Teora, București 1996.

[OTSU] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms." IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 62-66, 1979.

[PARDO1] X.M., Pardo, M.J., Carreira, A., Mosquera, D., Cabello – A snake for CT image segmentation integrating region and edge information, image and Vision Computing, Elsevier Science B.V. 2001.

[PHAM1] Pham D.T., Karaboga D., Self-tuning fuzzy controller design using genetic optimisation and neural network modelling, Artificial Intelligence in Engineering 13 (1999) 119–130.

[PITAS] Pitas, L, Venetsanopoulos, A. N. Nonlinear Digital Filters, Principles and Application Kluwer Academic Publ., Norwell MA, 1990

[POHLHEIM1] R., Pohlheim - The Genetic an Evolutionary Algorithm Toolbox, pohlheim@systemtechnik.tu-ilmeneau.de, 1993

[POPOVICI1] A., Popovici, D., Popovici, - Cellular Automata in Image Processing University of West Timisoara, Romania, 2001.

[PRECUP1] Precup, E., Preitl, Șt., Sisteme de reglare avansată, Centrul de multiplicare Univ. “Politehnica”, Timișoara, 1995.

[PREITL1] Preitl, Șt., Precup, E., Introducerea în conducerea fuzzy a proceselor. Ed. Tehnică, București, 1997.

[PRESS] Press W. H., Flannery, B. P., Teukolsky, W. T., Vetterling, W. T. Numerical Recipes in C. The art of scientific computing, Cambridge University Press, 1988

[RADULESCU] Rădulescu M., Drăgan N., Hubert H., Opriș C., Atlas metalografic. Editura Tehnică, București, 1972.

[REICH1] Reich Y., Barai S.,V., Evaluating machine learning models for engineering problems, Artificial Intelligence in Engineering 13 (1999) 257–272.

[REICH2] Reich Y., Barai S.V., A methodology for building neural networks models from empirical engineering data, Engineering Applications of Artificial Intelligence 13 (2000) 685-694.

[RESH1] R.V., Resh, M., Hunt - How to Choose an Imaging System,

www.metallography.com, 1998,

[REZNIK1] Reznik L., Ghanayem O., Bourmistrov A., PID plus fuzzy controller structures as a design base for industrial applications, *Engineering Applications of Artificial Intelligence* 13 (2000) 419-430.

[ROSENBERG] Rosenberger C., Cariou C., - Contribution to texture analysis - LVR - ENSI de Bourges, France christophe.rosenberger@ensi-bourges.fr, 2004.

[RUS1] Rus, A., Vesselenyi, T., Baban, C., **Assembly task modelling whit anthropomorphic hands**, *International Scientific Conference Modern Technologies, Quality, Restructuring*, pag. 199-202, Mai 2000 Iași.

[RUS2] Rus, A., Vesselenyi, T., Pele A., **Assembly using anthropo-morphical robot hands**, *International Scientific Conference Modern Technologies, Quality, Restructuring*, 2000, Iasi.

[SAISAN] Saisan P., Doretto G., Wu Y.N., Soatto S., - Dynamic Texture Recognition – UCLA, Electrical Eng. Los Angeles, CA 90095 saisan@ee.ucla.edu, 2004.

[SCHMITT] Schmitt, M., Mattioli, J. "Reconnaissance de formes planaires par morphologie mathematique et reseaux de neurones", in *Revue Technique Thomson CSF*, vol. 22, no. 4, Decernbrie 1990, pag. 573609, Ed. Gauthiers Villaxs: Paris

[SEDRICKS] Sedriks, A.J., Mulhearn T.O., Austenitic Manganese Steel: Structure and Properties of Decarburized Layer, *J.I.S.I.*, Vol 202, November 1964, p 907-911.

[SERRA] Serra, J. *Image Analysis and Mathematical Morphology*, Academic Press, London, 1982

[SHARMA] Sharma M., Markou M., Singh S., - Evaluation of texture methods for image analysis - m.sharma@exeter.ac.uk. PANN Research Department of Computer Science, University of Exeter, Exeter EX4 4PT U.K. 2004.

[SICIL1] Siciliano B., Villani L., An inverse kinematics algorithm for interaction control of a flexible arm with a compliant surface, *Control Engineering Practice* 9 (2001) 191-198.

[SLEPIAN] Slepian, R.M., Prohaska. J.P., An Improved Etchant for Copper and Copper Alloys, *Metallography*, Vol 9, 1976, p 51-61.

[SMITH] Smith S.W., *The Scientist and Engineer's Guide to Digital Signal Processing*, California Technical Publishing, San Diego, California.

[SOFRON] Sofron, E., *Sisteme de control fuzzy – Modelare și proiectare asistate de calculator*, Ed., All, București, 1998.

[SONGLING] Han Songlin, Shuxin Li, David J. Smith Comparison of phenological and crystallographic models for single crystal nickel base superalloys. II. Numerical simulation, *Image and Vision Computing*, 2000.

[SPATARU] Spătaru, A. *Teoria Transmisiunii Informației*, Ed. Didactică și Pedagogică, București, 1984

[SPITTLE1] J.A., Spittle, S.G.R., Brown - A cellular automaton model of steady-state columnar-dendritic growth in binary alloys University of Wales Swansea, Swansea, UK 15 August 1995, *Journal of Materials Science*, 16, 1995.

[STRUERS] ***Struers Inc., 20102 Progress Dr., Cleveland, Ohio 44136.

- [SU1] Su M., Chen Y., Wang K., Tew H., Huang H., 3D arm movement recognition using syntactic pattern recognition, *Artificial Intelligence in Engineering* 14 (2000) 113–118.
- [SZIRANYI] Sziranyi T., - Texture Segmentation by the 64x64 CNN Chip - submitted to World Scientific, 2002, 05.03.
- [TAM1] Tam H., Chi-hang L., O., Mok A., C., K., Robotic polishing of free-form surfaces using scanning paths, *Journal of Materials Processing Technology* 95 (1999) 191-200.
- [TODER1] Todorean G., Costeiu M., Giurgiu M., Rețele neuronale artificiale, Ed. Microinformatica, București, 1995.
- [TRIPE1] Tripe Vidican, A., Vesselenyi, T., Tripe Vidican, C., Elaborarea unei metode de modelare și simulare pe calculator a funcționării unei axe hidraulice de translație *Analele Universității Oradea-Fascicola Mecanică*, 2000.
- [TRIPE2] Tripe-Vidican, A., Vesselenyi, T., Tripe-Vidican, C., Țocuț, P. A study of computer aided control of a translation hydraulic axis V-a conferință internațională de mașini hidraulice și hidrodinamică HMH – Timișoara, 2000.
- [TRIPE3] Tripe-Vidican, A., Vesselenyi, T., Tripe-Vidican, C., Țarcă, R., Study of controller system of a hydraulic translation axis by computer simulation. A V-a conferință internațională de mașini hidraulice și hidrodinamică HMH 2000-Timișoara
- [TSAI1] D.M., Tsai, S.K., Wu, M.C., Chen - Optimal Gabor filter design for texture segmentation using stochastic optimization, *Image and Vision Computing*. Elsevier Science B.V. 2000.
- [TUCERYAN] Tuceryan M., Jain A.K., - Texture Analysis - Department of Computer and Information Science, Indiana University – Handbook of Pattern Recognition and Computer Vision (2nd Edition) 207-248, World Scientific Publishing Co., 1998.
- [ȚARCĂ1] Radu Țarcă, Florin Blaga, Naiana Țarcă, Voichita Hule, **Vesselenyi Tiberiu, The 2002 FMS functional modelling and simulation using Taylor program**, MICROCAD 2003, Miskolc, Hungary, feb. 2003, ISBN;
- [ȚARCĂ2] Țarcă Radu, Blaga Florin, Țarcă Ioan, **Vesselenyi Tiberiu, A visual servoing robot system**, 7th International Research /Expert Conference “Trends in the Develop-ment of Machinery and Associated Technology” TMT 2003. Lloret de Mar, Barcelona, Spain, 15-16 September 2003;
- [ȚARCĂ3] Țarcă Radu, Țarcă Ioan, Tripe V. Aron, **Vesselenyi Tiberiu, Considerații teoretice privind recunoașterea unor obiecte de formă geometrică simplă**, Tehno '95 Timișoara A șaptea Conferință Internațională de Inginerie Managerială și Tehnologică, pag. 137-142, Timișoara, 1995;
- [ȚARCĂ4] Radu Țarcă, **Tiberiu Vesselenyi**, Florin Blaga, Ioan Țarcă, **Theoretical aspects of using formal language in pattern recognition**, ANNALS of the Oradea University, Management and Technological Engineering, ISSN 1583-0691, Editura Univ. din Oradea, vol. III (XIII), pag. 83, 2004;
- [ȚARCĂ5] Țarcă Radu, Țarcă Ioan, **Vesselenyi Tiberiu, Simularea conducerii adaptive – utilizând un model de referință – a unei axe de robot – partea I**,

- Simpozionul Național de Robotică, ediția a XIV a, pag. 255-258, Brașov, 1998;
- [ȚARCĂ6] Țarcă Radu, Țarcă Ioan, **Vesselenyi Tiberiu, Simularea conducerii adaptive – utilizând un model de referință – a unei axe de robot – partea II**, Simpozionul Național de Robotică, ediția a XIV a, pag. 259-262, Brașov, 1998;
- [ȚARCĂ7] Țarcă Radu, **Vesselenyi Tiberiu, Țarcă Ioan, Aplicații ale limbajelor formale în recunoașterea imaginilor**, Al IV-lea Congres al Academiei de Cibernetică "Ștefan Odobleja", pag. 336-341. Băile Felix, 1996, ISBN 973 – 9183 – 53 – 0;
- [ȚARCĂ8] Țarcă Radu, Țarcă Ioan, Tripe V. Aron, **Vesselenyi Tiberiu, Utilizarea produselor tensoriale în algoritmi de recunoaștere a formelor**, Analele Universității din Oradea, vol.I, Fascicola Mecanică, 1995, 1221 – 555x, pag. 85-88;
- [ȚARCĂ9] Țarcă Radu, Țarcă Ioan, **Vesselenyi Tiberiu, Modelarea dinamică a roboților manipulatori utilizând ecuațiile Gibbs**, Analele Universității din Oradea, vol.I, Fascicola Mecanică, 1995, 1222 – 5517, pag. 123-126;
- [UIC1] ***Universal Imaging Corporation Device Control & System Automation, www.uic.com, 2003.
- [VANDERVOORT] VanderVoort, G.F., Metallography: Principles and Practice, McGraw-Hill Book Co., New York, 1984.
- [VARMA1] M.R., Varma, R., Sasikumar, S.G.K., Pillai, P.K., Nair - Cellular automaton simulation of microstructure evolution during austenite decomposition under continuous cooling condition Indian Institute of Technology, Chennai, India, Bulletin of Material Science , Indian Academy of Science. June 2001
- [VENCES1] L., Vences, I., Rudomin - Genetic Algorithm for Fractal Image and Image Sequence Compression Instituto Tecnológico de Estudios Superiores de Monterrey Campus Estado de Mexico, Computacion Visual, 1997.
- [VESSELENYI1] **Vesselényi, T., Barabás, T., Moga, I., Flexible cell for metallographic laboratories**, Robotica & Management, International Journal, vol 9, No.1., November, 2004;
- [VESSELENYI2] **Vesselényi, T., Barabás, T., Studies regarding computerized analysis of metallographic images**, Robotica & Management, International Journal, vol 9, No.1., November, 2004;
- [VESSELENYI3] **Vesselényi, T., Ungur, P., Mudura, P., Moga, I., An algorithm for microstructure images concatenation** Advanced Materials And Structures, Second International Conference, 19-21 sept. 2002, Timișoara Ed. Orizonturi Univ. Timișoara 2002, ISBN 973-8391-50-4, pag. 377-380;
- [VESSELENYI4] **Vesselenyi T., Mudura P., Ungur P., Position and field correction algorithm for microstructure image acquisition**, Advanced Materials And Structures, Second International Conference, 19-21 sept.2002, Ed. Orizonturi Univ.Timișoara, pag. 381-384, Timișoara. 2002, ISBN:973-8391-50-4;
- [VESSELENYI5] **Vesselenyi Tiberiu, Tripe-Vidican Călin, Metallographical analysis automation a flexible system approach**, 5`th International Conference on Engineering of Modern Electric Systems, May 27-29, Felix SPA, 1999;

- [VESSELENYI6] **Vesselenyi Tiberiu**, Mudura Pavel, Țarcă Radu, **Algoritmi statistici pentru recunoașterea formelor vizuale**, A V-a Conferință Internațională de Matematici Aplicate, pag. 255-260, Băile Felix, 1997;
- [VESSELENYI7] **Vesselenyi Tiberiu**, Țarcă Radu, Țarcă Ioan, Tripe V. Aron, **Algoritmi de analiză a imaginii folosiți în recunoașterea formelor**, Tehno '95 Timișoara A șaptea Conferință Internațională de Inginerie Managerială și Tehnologică, pag. 143-148, Timișoara, 1995,
- [VESSELENYI8] **T. Vesselenyi**, I. Moga, N. Fântână, **Celula flexibilă pentru laboratoare metalografice**, ANNALS of the Oradea University, Management and Technological Engineering, ISSN 1583-0691, Editura Univ. din Oradea, vol. III (XIII), pag.87 , 2004;
- [VESSELENYI9] **Vesselenyi Tiberiu**, Mudura Pavel, Țarcă Radu, Sistem de reglare automat al clarității imaginii obținute la microscopul optice metalografice, Simpozion materiale avansate, tratamente termice și calitatea managementului, Zilele Academice Timișene, pag. 305-308, 2001, ISBN 973 – 8247 – 32 – 2;
- [VESSELENYI10] **T.Vesselenyi**, P. Mudura, **Analiza caracteristicilor transformatei Fourier rapide ale imaginilor digitale ale unor structuri metalografice**, Simpozion materiale avansate, tratamente termice și calitatea managementului, Zilele Academice Timișene, pag. 301-304, 2001, ISBN 973 – 8247 – 32 – 2;
- [VESSELENYI11] **Tiberiu Vesselenyi**, C. Tripe Vidican, Robotizarea operațiilor aferente analizei metalografice, Al XIV-lea Simpozion Național ROBOTICĂ '98, 5-7 nov. 1998, Ed. Universității Transilvania din Brașov, pag. 447-450, Brașov, 1998, ISBN 973-98796-1-6;
- [VESSELENYI12] **Vesselenyi Tiberiu**, Țarcă Ioan, Țarcă Radu, Algoritmi de identificare a imaginilor simple, Al IV-lea Congres al Academiei de Cibernetică "Ștefan Odobleja", pag. 342-347, Băile Felix, 1996, ISBN 973 – 9183 – 53 – 0;
- [VESSELENYI13] **Tiberiu Vesselenyi**, Pavel Mudura, Ioan Moga, **A study of microstructure recognition and image processing methods**, 11th Conference on Applied and Industrial Mathematics, University of Oradea & Romanian Society of Applied and Industrial Mathematics, la Oradea, 29-31 May 2003, (comunicare fără tipărirea lucrării);
- [VERTAN] Vertan, C., Prelucrarea și analiza imaginilor, Editura Printech 1999.
- [VOLOVICI1] Volovici D., Aplicații ale rețelelor neuronale artificiale la conducerea proceselor tehnologice, Editura Universității din Sibiu, 1995.
- [WAHL] Wahl, F.M. Digital Image Signal Processing, Artech House, Boston, 1987
- [WANG] Wang L., Liu J., - Texture classification using multiresolution Markov random field models – Pattern Recognition Letters, 20,1999, 171-182.
- [WENMEI1] Wenmei H., Yong Y., Yali T., Adaptive neuron control based on predictive model in pneumatic servo system. INTERNET (2002).
- [WIECZROWSKI1] M., Wieczorowski - Spiral sampling as a fast way of data acquisition in surface topography, International Journal of Machine tools &

Manufacture, Elsevier Science Ltd, 2001.

[WOHLER1] C., Wohler, J.K., Anlauf - Real-time object recognition on image sequences with the adaptable time delay neural network algorithm – application for autonomous vehicles, Ulm, Bonn, Germany, Image and Vision Computing, Elsevier Science Ltd. 2001.

[WOLFRAM1] S., Wolfram – Publications by Stephen Wolfram, Cellular Automata Los Alamos Science 1983, www.stephenwolfram.com

[WOO1] H., Woo, E., Kang, S., Wang, K.H., Lee – A new segmentation method for point cloud data, International Journal of Machine Tools & Manufacture, Elsevier Science Ltd, 2001.

[XIAO1] S.J., Xiao, X.Q., Jiang, L., Blunt, P.J., Scott - Comparison study of the biorthogonal spline wavelet filtering for areal rough surfaces, University of Huddersfield, UK

[XIONG] Feng Xiong, Xiang Qian Jiang, Yongsheng Gao, Zhu Li, Evaluation of engineering surface using a combined fractal modeling and wavelet analysis method, International Journal of Machine Tools & Manufacture, 2001 vol.41, pag2187-2183, Elsevier Science Ltd.

[YANAGI1] K., Yanagi, S., Hara, T., Endoh - Summit identification of anisotropic surface texture and directionality assessment based on asperity tip geometry, Nagaoke University of Technology, Tokyo Institute of Technology, Japan, International Journal of Machine tools & Manufacture, Elsevier Science Ltd, 2001.

[YOUSSEF1] Youssef .H, Sait S.,M., Adiche H., Evolutionary algorithms,simulated annealing and tabu search: a comparative study, Engineering Applications of Artificial Intelligence 14 (2001)167-181.

[YUAN1] C., Yuan, H., Niemann - Neural networks for the recognition and pose estimation of 3D objects from single 2D perspective view, University of Erlangen-Nuremberg, Germany, Image and Vision Computing , Elsevier Science B.V. 2001.

[ZAHOUANI1] H., Zahouani, M., Assoul, R., Vargiolu, T., Mathia - The morphological tree transform of surface motifs. Incidence in tribology, Ecole Centrale de Lion, Universite France Comte, France, Intenational Journal of Machine tools & Manufacture, Elsevier Science Ltd, 2001.

[ZANPERONI] Zanperoni, P. “Image. Enhancement”, Advances in Imaging and Electron Physics, vol. 92, pp. 177, Academic Press, 1995

[ZERVAKIS1] M., E., Zervakis, V., Sundararajan, K., K., Parhi - Vector processing of wavelet coefficients for robust image denoising, Technical University of Crete, Greece University of Minnesota, Minneapolis, USA, Image and Vision Computing, Elsevier Science 2001.

[ZHANG1] Zhang J., Knoll A., Schmidt R., A neuro-fuzzy control model for fine-positioning of manipulators, Robotics and Autonomous Systems 32 (2000) 101–113.

[ZHONG] Zhong, J., Scarlaroff, S.– Temporal Texture Recognition Model Using 3D Features - Department of Computer Science. Boston University, 2004.