

# **CONTRIBUȚII LA ANALIZA DEPENDABILITĂȚII SISTEMELOR AUTOMATE**

Teză destinată obținerii  
titlului științific de doctor inginer  
la  
Universitatea "Politehnica" din Timișoara  
în domeniul AUTOMATICĂ  
de către

**Ing. Adrian Ștefan Korodi**

Conducător științific: prof.univ.dr.ing. Toma Leonida Dragomir  
Referenți științifici: prof.univ.dr.ing. Tiberiu Coloși  
prof.univ.dr.ing. Eugen Iancu  
prof.univ.dr.ing. Virgil Tiponut

Ziua susținerii tezei: 16.11.2007

Seriile Teze de doctorat ale UPT sunt:

- |                        |   |
|------------------------|---|
| 1. Automatică          | 7. Inginerie Electronică și Telecomunicații |
| 2. Chimie              | 8. Inginerie Industrială                    |
| 3. Energetică          | 9. Inginerie Mecanică                       |
| 4. Ingineria Chimică   | 10. Știința Calculatoarelor                 |
| 5. Inginerie Civilă    | 11. Știința și Ingineria Materialelor       |
| 6. Inginerie Electrică |   |

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

Copyright © Editura Politehnica – Timișoara, 2007

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

România, 300159 Timișoara, Bd. Republicii 9,  
tel. 0256 403823, fax. 0256 403221  
e-mail: editura@edipol.upt.ro

## Cuvânt înainte

Teza de doctorat a fost elaborată pe parcursul activității mele în cadrul Universității „Politehnica” din Timișoara, Facultatea de Automatică și Calculatoare, Departamentul de Automatică și Informatică Aplicată.

Mulțumiri deosebite adresez coordonatorului meu de doctorat prof. dr. ing. Toma Leonida Dragomir, alături de care am avut oportunitatea și plăcerea să lucrez.

Timișoara, Noiembrie 2007

Adrian Ștefan Korodi

Părinților mei.

Korodi, Adrian Ștefan

**Contribuții la analiza dependabilității sistemelor automate**

Teze de doctorat ale UPT, Seria 1, Nr. 7, Editura Politehnica, 2007, 206 pagini, 205 figuri, 8 tabele.

ISSN: 1842-5208

ISBN: 978-973-625-543-4

Cuvinte cheie:

Dependabilitate, Disponibilitate, Redundanță, Roboți mobili, Centrală electrică geotermală, Regulator interpolativ, Toleranță la defecte.

Rezumat,

Dependabilitatea sistemelor exprimă calitatea acestora de a prezenta încredere justificată. În cazul sistemelor automate problema are nuanțe specifice conferite de structura acestor sisteme. În acest context teza sintetizează noțiuni referitoare la dependabilitatea sistemelor automate (partea I-a), dezvoltă aspecte de analiză, dezvoltare și implementare de structuri redundante (partea a II-a), prezintă cercetări și contribuții referitoare la reglarea și disponibilitatea unei centrale electrice geotermale (partea a III-a) și tratează probleme de analiză a dependabilității roboților mobili, dezvoltându-se un regulator interpolativ tolerant la defecte, respectiv o metodă de corecție a direcției de deplasare a roboților bazată pe imagini (partea a IV-a).

## CUPRINS

Lista de notații și abrevieri .....	1
Lista de tabele .....	6
Lista de figuri .....	7
Introducere .....	13
1. Dependabilitatea sistemelor automate .....	15
1.1 Dependabilitatea .....	15
1.2 Fiabilitate și disponibilitate .....	15
1.3 Defectul .....	19
1.4 Mentenanța .....	20
1.5 Elemente de proiectare orientată pe calitate .....	22
1.6 Toleranța la defecțiuni .....	24
1.7 Sistemele automate .....	25
2. Studii privind analiza și implementarea unor structuri redundante .....	27
2.1 Clasificarea structurilor redundante .....	28
2.1.1 Lucrări semnificative .....	28
2.1.2 Clasificarea redundanțelor în funcție de nivelul la care se aplică .....	31
2.1.3 Clasificarea redundanțelor în funcție de reacția sistemului la defect .....	32
2.1.4 Clasificarea redundanțelor în funcție de zonă și de modalitatea de realizare .....	33
2.1.5 Aspecte privind utilizarea structurilor redundante în cazuri critice de dezastru și atacuri voite .....	47
2.2 Studiul unei structuri redundante hibride – structura m-n-k .....	48
2.2.1 Redundanța m-n-k hibridă .....	48
2.2.2 Implementarea redundanței hibride m-n-k pe DSP .....	52
2.2.3 Utilizarea structurii cu redundanță hibridă 2-3-1 în cadrul sistemului de conducere al unei mașini electrice .....	59
2.3 Studiul structurilor cu redundanță activă omogenă cu k componente reparabile.....	61
2.3.1 Structura redundanță activă cu k entități reparabile identice, aspecte referitoare la ratele de tranziție .....	61
2.3.2 Studiu de caz: un sistem redundant cu 3 entități reparabile .....	64
2.3.3 Calculul ratei de reparare a unui ansamblu serie .....	68
2.4 Tranziții simultane în cazul structurilor reparabile cu redundanță activă ...	71
3. Studiul unor aspecte referitoare la reglarea și disponibilitatea unei centrale electrice geotermale .....	74
3.1 Centrala electrică geotermală .....	74
3.2 Implementarea pe un caz real a dezvoltărilor teoretice referitoare la structurile cu redundanță activă cu k componente reparabile identice .....	76
3.3 Studiu de caz pentru dezvoltările teoretice referitoare la tranzițiile simultane în cazul utilizării structurilor reparabile cu redundanță activă ....	83
3.4 Regulatori interpolative .....	87
3.5 Soluție de reglare a temperaturii bazată pe interpolare .....	89
3.5.1 Regulator tripozițional cu reacție operațională cu acțiuni de tip PI .....	90
3.5.2 Structura de reglare al temperaturii la ieșirea din blocul vaporizatoare al centralei electrice geotermale .....	91
3.5.3 Soluția de reglare a temperaturii CO <sub>2</sub> la ieșirea blocului vaporizatoare ..	94
3.6 Creșterea gradului de disponibilitate în cadrul blocului vaporizatoare .....	101

3.6.1 Creșterea disponibilității prin redundanță a structurii de reglare a temperaturii $\theta_1$ de la ieșirea blocului vaporizatoare .....	102
3.6.2 Performanțele structurii propuse pentru reglarea temperaturii $\theta_1$ din perspectiva disponibilității .....	104
3.7 Modul mobil de detecție și diagnoză a defectelor pentru sisteme automate .....	107
3.7.1 Algoritmul de detecție și diagnoză a defectelor .....	107
3.7.2 Implementarea modului mobil de detecție și diagnoză a defectelor ....	113
4. Studiul unor aspecte referitoare la dependabilitatea roboților mobili .....	119
4.1 Structuri de reglare utilizate pentru sisteme robotizate .....	119
4.1.1 Tipuri de regulatoare utilizate în sisteme robotizate .....	119
4.1.2 Cautarea și interpolarea tridimensională locală .....	124
4.1.3 Tratarea depășirilor pentru tabelele de interpolare .....	126
4.2 Defecte în sisteme robotizate .....	128
4.2.1 Roboți care evoluează în medii cu hazard ridicat .....	128
4.2.2 Roboți mobili care evoluează în spații semicunoscute .....	131
4.2.3 Manipulatoare robotice .....	133
4.3 Conceperea unui regulator tolerant la defecte pentru sisteme robotizate ..	134
4.3.1 Regulatorul interpolativ propriu-zis .....	135
4.3.2 Redundanța bazată pe referință .....	141
4.3.3 Mecanismul de corecție al regulatorului .....	152
4.3.4 Experimente .....	156
4.3.5 Analiză de disponibilitate .....	167
4.4 Aspecte referitoare la creșterea disponibilității și siguranței în deplasarea roboților mobili cu roți .....	170
4.4.1 Robotul X80 .....	171
4.4.2 Deplasarea autonomă a robotului .....	172
4.4.3 Erori de odometrie .....	175
4.4.4 Procedeu de corecție a direcției de deplasare a robotului prin preluarea și procesarea de imagine .....	176
4.4.5 Studiu practic .....	180
4.4.6 Analiză de fiabilitate și disponibilitate .....	184
5. Concluzii. Contribuții personale. Diseminarea rezultatelor. ....	189
Anexa .....	192
Bibliografie .....	197
Index .....	206

## Lista de notații și abrevieri

$\nearrow$	- valoarea actuală a mărimii este mai mare decât valoarea nominală (cap. 3)
$\searrow$	- valoarea actuală a mărimii este mai mică decât valoarea nominală (cap. 3)
$\varepsilon_1$	- parametru pentru zona de insensibilitate (cap. 3)
$\{p\}$	- vector al parametrilor de proiectare (cap. 1)
$\{r\}$	- vector al intrărilor aleatoare (cap. 1)
$\{x\}$	- vectorul caracteristicilor de performanță continue (cap. 1)
$A(t)$	- disponibilitatea (cap. 1, 2, 3, 4)
$A_\infty$	- disponibilitatea în regim staționar (cap. 1)
$A_{\text{inerentă}}$	- disponibilitatea inerentă (cap. 1, 4)
$A_{\text{operațională}}$	- disponibilitatea operațională (cap. 1, 4)
$A_p$	- disponibilitatea structurii paralel (cap. 1)
AP, PLC	- automat programabil (cap. 3)
$A_{\text{rezultată}}$	- disponibilitatea rezultată (cap. 1)
$A_s$	- disponibilitatea structurii serie (cap. 1, 2)
BC	- bloc de corecție (cap. 4)
C	- comparator (cap. 2)
$C_1, C_2$	- blocuri de decizie și comandă (cap. 4)
CD	- comparator și element de decizie (cap. 2)
CE	- corecție operațională (cap. 3)
$C_L$	- costul mediu al echipei de reparare pe oră (cap. 1)
$C_M$	- costul mediu al materialelor pentru service (cap. 1)
$C_{PB}$	- costul procesului pe ora la defectare (cap. 1)
$C_{PP}$	- costul pierderilor procesului pe ora pierdută (cap. 1)
$C_R$	- costul mediu al materialelor pentru reparare (cap. 1)
$C_{\text{total\_defecte}}$	- costul total al defectelor pentru mentenanța breakdown (cap. 1)
$C_{\text{total\_preventivă}}$	- costul total al mentenanței preventive (cap. 1)
d	- distanța (cap. 4)

---

D	- bloc de decizie (cap. 2)
$d_p$	- distanța parcursă (cap. 4)
$d_r$	- distanța rămasă de parcurs (cap. 4)
DSP	- procesor de semnal numeric (cap. 2)
e	- semnalul furnizat de senzorul de distanță (cap. 4)
E	- senzor de distanță (cap. 4)
$e_b$	- semnalul furnizat de senzorul bază (cap. 4)
$E_i$	- elementul i al structurii (cap. 2)
$e_r$	- semnalul furnizat de senzorul referință (cap. 4)
ETA	- analiza pe baza arborelui evenimentelor (cap. 1)
f	- factor ajustabil (cap. 2)
$F(t)$	- non-fiabilitatea (cap. 1)
FC	- robinet blocat închis (cap. 3)
FMEA	- metoda analizei defectelor mod-efect (cap. 1)
FOC	- robinet blocat complet deschis (cap. 3)
FON	- robinet blocat deschis, nu complet deschis (cap. 3)
FTA	- metoda arborelui defectelor (cap. 1)
G	- grup de entități (cap. 2)
$h_1, h_3$	- deschiderile valvelor $RB_1$ și $RB_3$ (cap. 3)
IM1	- imaginea sursă (cap. 4)
IM2	- imaginea destinație (cap. 4)
$IP-K_{CE}, IP-T'_{CE}$	- tabelele de interpolare prin care se ajustează parametrii de corecție ai regulatorului $K_{CE}, T'_{CE}$ (cap. 3)
$K_{CE}, T'_{CE}$	- parametrii de corecție ai regulatorului (cap. 3)
lim	- valoare limită (cap. 2)
LSL, USL	- limita inferioară, respectiv superioară de specificație (cap. 1)
m	- frecvență de mentenanță (cap. 1)
M	- bloc de măsurare (cap. 4)
M&C	- măsurare și reconfigurare (cap. 4)
MAT1	- matricea în care se memorează IM1 (cap. 4)
MAT2	- matricea în care se memorează IM2 (cap. 4)
MDT	- timpul mediu cât durează un defect (cap. 1, 2, 4)
MFDDM	- modul mobil de detecție și diagnoză a defectelor (cap. 3)
MMT	- timpul mediu de mentenanță (cap. 1)



---

m-n-k	- caracteristicile redundanței hibride m-n-k (cap. 2)
MPDT	- timpul mediu de cădere pentru mentenanța preventivă (cap. 1)
MR	- mecanismul de corecție (cap. 4)
MTBF	- timpul mediu între defectiuni (cap. 1, 4)
MTBM	- timpul mediu între acțiunile mentenanței preventive (cap. 1)
MTTF	- timpul mediu până la defectare (cap. 1)
MTTR	- timpul mediu de reparare (cap. 1, 4)
N	- numărul de entități monitorizate (cap. 1)
$N_1, N_2$	- numărătoare (cap. 4)
NB	- bloc de control nelinear (cap. 3)
$N_F$	- numărul de defecte identificate în perioada de timp T (cap. 1)
NL	- deviații largi negative ale ieșirii senzorului (cap. 3)
$n_p$	- numărul de pași de eșantionare stabiliți pentru identificarea unui defect permanent (cap. 4)
NS	- deviații reduse negative ale ieșirii senzorului (cap. 3)
OB1	- obiectul căutat (cap. 4)
$p_a$	- prag admisibil (cap. 4)
$P_i$	- probabilitatea sistemului de a se găsi în starea i (cap. 2, 3, 4)
$P_{i,j,k}$	- punct de sprijin (cap. 3, 4)
PL	- deviații largi pozitive ale ieșirii senzorului (cap. 3)
PS	- deviații reduse pozitive ale ieșirii senzorului (cap. 3)
$q_{ac}$	- debitul de apă caldă (cap. 3)
$q_{ar}$	- debitul de apă rece (cap. 3)
$q_{co2, x-y}$	- debitul de CO <sub>2</sub> între două subansamble consecutive x și y (cap. 3)
R	- regulatorul interpolativ propriu-zis (cap. 4)
R(t)	- fiabilitatea (cap. 1, 3, 4)
$RA_1'$	- regulatorul din cadrul blocului vaporizatoare (cap. 3)
$RA_1''$	- regulatorul cu caracter redundant din cadrul blocului vaporizatoare (cap. 3)
$RB_1$	- valva de acces (robinet) a apei calde în vaporizatoare (cap. 3)
$RB_3$	- valva de acces (robinet) a apei reci în condensatoare (cap. 3)
RBR	- redundanța bazată pe referință (cap. 4)
red	- coeficient de reducere a vitezei robotului mobil (cap. 4)
$R_{i,j,k}$	- valoare de sprijin (cap. 3, 4)

---

RSC	- realitate sporită de calculator (cap. 2)
SB	- sistem de bază (cap. 1)
SW	- element de comutare (cap. 2)
T	- perioada de funcționare, monitorizare (cap. 1)
TCLK0	- pin al temporizatorului 0 (cap. 2)
TCLK1	- pin al temporizatorului 1 (cap. 2)
$T_i$	- timpul de viață al unei entități (cap. 1)
TLOC	- prețul pe întreg ciclul de viață (cap. 1)
$T_P$	- pas de predicție (cap. 2)
$TT_1$	- senzorul care măsoară $\theta_1$ (cap. 3)
$TT_3$	- senzorul care măsoară $\theta_3$ (cap. 3)
$U(t)$	- non-disponibilitatea (cap. 1)
$u_1, u_3$	- semnalele de comandă pentru motoarele valvelor $RB_1$ , respectiv $RB_3$ , furnizate de regulatoarele din cadrul buclelor de reglare ale blocurilor vaporizatoare, respectiv condensatoare (cap. 3)
$U_p$	- non-disponibilitatea structurii paralele (cap. 1)
V	- element de votare (cap. 2)
$V_{ai}$	- locuri de sprijin (cap. 3, 4)
$v_c$	- viteza curentă (cap. 4)
XF0, XF1	- pini liberi ai procesorului de semnal (cap. 2)
$x_{iT}$	- limite de specificație (cap. 1)
$y_d$	- intrarea structurii de achiziție (cap. 4)
$\alpha$	- factorul de corecție (cap. 4)
$\alpha_{lim}$	- limita factorului de corecție (cap. 4)
$\Delta x$	- deviația care conduce la corecția orientării (cap. 4)
$\theta_1$	- temperatura $CO_2$ la ieșirea din vaporizatoare (cap. 3)
$\theta_{1nom}, \theta_{3nom}$	- valorile nominale pentru $\theta_1$ și $\theta_3$ (cap. 3)
$\theta_{1r}$	- temperatura de pe reacție în vaporizatoare (cap. 3)
$\theta_2$	- temperatura $CO_2$ la intrarea în blocul condensatoare (cap. 3)
$\theta_3$	- temperatura $CO_2$ la ieșirea din condensatoare (cap. 3)
$\theta_{3r}$	- temperatura de pe reacție în condensatoare (cap. 3)
$\theta_4$	- temperatura $CO_2$ la intrarea în blocul vaporizatoare (cap. 3)
$\theta_{ac}$	- temperatura apei calde la ieșirea din vaporizatoare (cap. 3)

---

$\theta_{acmax}, \theta_{armin}$	- praguri pentru temperaturile apei calde și reci la ieșirea din blocul vaporizatoare, respectiv condensatoare (cap. 3)
$\theta_{ar}$	- temperatura apei reci la ieșirea din condensatoare (cap. 3)
$\lambda$	- rata de defectare (cap. 1, 2, 3, 4)
$\mu$	- rata de reparare (cap. 1, 2, 3, 4)

## Lista de tabele

Tabelul 3.1	Ratele de tranziție ale entităților din figura 3.2.3, precum și ratele de reparare determinate experimental pentru structurile redundante .....	77
Tabelul 3.2	Ratele de defectare și de reparare ale entităților subansamblului 7 din tabelul 3.1, ratele de reparare și ratele de tranziție simultane ale structurii redundante .....	84
Tabelul 3.3	Ratele de defectare și de reparare ale entităților subansamblului 7 din tabelul 3.1, precum și ratele de reparare ale structurii redundante .....	84
Tabelul 3.4	Tabelul de interpolare pentru parametrul $K_{CE}$ .....	98
Tabelul 3.5	Tabelul de interpolare pentru parametrul $T'_{CE}$ .....	99
Tabelul 3.6	Sintetizarea celor 12 scenarii de identificare a defectelor din cadrul subcapitolului 3.7 .....	113
Tabelul 4.1	Defectarea subsistemelor roboților mobili .....	130
Tabelul 4.2	Elemente de fiabilitate pentru doi roboți mobili .....	130

## Lista de figuri

Fig. 1.1.1	Structură serie cu n componente .....	18
Fig. 1.1.2	Conexiunea paralel .....	19
Fig. 1.1.3	Model Markov cu două stări .....	19
Fig. 1.1.4	Procesul de proiectare .....	23
Fig. 1.1.5	Sistem tolerant la defecțiuni destinat realizării funcției sistemului de bază SB .....	24
Fig. 2.1.1	Clasificarea ISCAN a redundanțelor .....	29
Fig. 2.1.2	Clasificarea redundanțelor după Zalewsky .....	30
Fig. 2.1.3	Structuri prevăzute cu redundanță la diferite nivele .....	31
Fig. 2.1.4	Clasificarea redundanțelor în funcție de reacția sistemului la defect .....	32
Fig. 2.1.5	Clasificarea redundanțelor în funcție de sursa defectului .....	33
Fig. 2.1.6	Clasificarea redundanțelor .....	33
Fig. 2.1.7	Structura paralel cu n entități .....	34
Fig. 2.1.8	Redundanță mascată .....	35
Fig. 2.1.9	Redundanță mascată de ordin trei .....	35
Fig. 2.1.10	Redundanța standby hot .....	36
Fig. 2.1.11	Redundanța standby cold .....	36
Fig. 2.1.12	Redundanța standby hibridă .....	37
Fig. 2.1.13	Evoluția șirului memorat .....	38
Fig. 2.1.14	Redundanță de timp simplă (repetarea necondiționată de sesizarea defectului) .....	40
Fig. 2.1.15	Filtrul de întârziere .....	41
Fig. 2.1.16	Exemplu de redundanță de timp (repetarea condiționată de sesizarea defectului) .....	41
Fig. 2.1.17	Redundanță hibridă, exemplul 1 .....	42
Fig. 2.1.18	Redundanță hibridă, exemplul 2 .....	43
Fig. 2.1.19	Redundanță hibridă, exemplul 3 .....	43
Fig. 2.1.20	Mentenanța cu ajutorul tehnicii RSC .....	46
Fig. 2.2.1	Reprezentarea schematică a redundanței hibride m-n-k .....	48
Fig. 2.2.2	Schema de funcționare a redundanței hibride m-n-k .....	49
Fig. 2.2.3	Reprezentarea unui element al structurii redundante .....	49
Fig. 2.2.4	Schema logică pentru redundanța hibridă 2-3-1 .....	51
Fig. 2.2.5	Arhitectura generală a sistemului .....	52
Fig. 2.2.6	Achiziția semnalelor de selecție .....	53
Fig. 2.2.7	Schema logică pentru preluarea semnalelor .....	54
Fig. 2.2.8	Semnalul de intrare .....	57
Fig. 2.2.9	Evoluția celor patru traductoare .....	58
Fig. 2.2.10	Ieșirea sistemului .....	58
Fig. 2.2.11	Motor sincron cu magneți permanenți prevăzut cu o structură redundantă hibridă 2-3-1 .....	59
Fig. 2.2.12	Evoluția traductorului 1 .....	60
Fig. 2.2.13	Evoluția traductorului 2 .....	60
Fig. 2.2.14	Evoluția traductorului 3 .....	60
Fig. 2.2.15	Evoluția traductorului 4 .....	60

Fig. 2.2.16	Evoluția ieșirii structurii redundante .....	61
Fig. 2.3.1	Structura redundanță omogenă de ordin k .....	61
Fig. 2.3.2	Modelul Markov al structurii redundante ilustrată în fig. 2.3.1 .....	62
Fig. 2.3.3	Modelul Markov pentru structura redundanță cu trei entități reparabile .....	64
Fig. 2.3.4	Exemplu de structură serie cu n componente .....	68
Fig. 2.3.5	Modelul Markov asociat schemei serie .....	69
Fig. 2.4.1	Modelul Markov al structurii redundante admitând tranziții simultane .....	71
Fig. 2.4.2	Structura redundanță de ordin doi în care se admit tranziții simultane .....	72
Fig. 3.1.1	Privire generală asupra centralei electrice geotermale .....	74
Fig. 3.1.2	Structura centralei electrice geotermale .....	75
Fig. 3.2.1	Schemă de analiză a fiabilității sistemului de control al ciclului termodinamic .....	76
Fig. 3.2.2	Schema de fiabilitate a subsistemului condus al buclei de reglare a temperaturii $t_1$ .....	76
Fig. 3.2.3	Schema de fiabilitate cu redundanță a subsistemului condus al buclei de reglare a temperaturii $t_1$ .....	77
Fig. 3.2.4	Schema de implementare a sistemului prevăzut cu redundanțe duble și triple .....	79
Fig. 3.2.5	Schema de implementare a sistemului fără redundanțe .....	80
Fig. 3.2.6	Disponibilitatea subansamblului cu redundanță triplă ( $A_8'$ ) .....	81
Fig. 3.2.7	Disponibilitatea subansamblului neredundant ( $A_8$ ) .....	81
Fig. 3.2.8	Non-disponibilitatea subansamblului cu redundanță triplă .....	81
Fig. 3.2.9	Non-disponibilitatea subansamblului neredundant .....	81
Fig. 3.2.10	Disponibilitatea subansamblului cu redundanță dublă ( $A_7'$ ) .....	82
Fig. 3.2.11	Disponibilitatea subansamblului neredundant ( $A_7$ ) .....	82
Fig. 3.2.12	Reprezentarea disponibilităților în cazul în care structurile prezintă sau nu redundanță .....	82
Fig. 3.3.1	Ansamblul serie acționare debit motor+RB <sub>2</sub> .....	83
Fig. 3.3.2	Structură redundanță de ordin 2 a ansamblului din fig. 3.3.1 .....	83
Fig. 3.3.3	Schemele de implementare ale structurii care admite rate de tranziție simultane (sus), respectiv a structurii care nu admite rate de defectare simultane (jos) .....	85
Fig. 3.3.4	Evoluția în timp a probabilității ca sistemul să se afle în starea ① .....	85
Fig. 3.3.5	Evoluția în timp a probabilității ca sistemul să se afle în starea ① .....	86
Fig. 3.3.6	Evoluția în timp a probabilității ca sistemul să se afle în starea ② .....	86
Fig. 3.3.7	Reprezentarea disponibilităților pentru cele două cazuri .....	87
Fig. 3.4.1	Structura simplă a unui regulator interpolativ .....	88
Fig. 3.5.1	Regulator PI cu filtrare realizat cu element tripozițional cu histerezis și reacție operațională .....	90
Fig. 3.5.2	Poziția buclei actuale de reglare a temperaturii $\theta_1$ .....	91
Fig. 3.5.3	Blocul vaporizatoare .....	92
Fig. 3.5.4	Bucle de reglare modificată a temperaturii $\theta_1$ în cadrul centralei electrice geotermale .....	94

Fig. 3.5.5	Poziția noii bucle de reglare a temperaturii $\theta_1$ .....	95
Fig. 3.5.6	Mișcarea de deschidere a valvei RB <sub>1</sub> pentru $u_1(t)=u_{10}$ (a) și caracteristica elementului tripozițional (b) .....	96
Fig. 3.5.7	Răspunsul la semnal treaptă $\theta_1(t)$ al vaporizatorului (a. – pentru $h_1(t)=1$ , b. – pentru $u_1(t)=u_{10}$ ) .....	97
Fig. 3.5.8	Schema Simulink utilizată pentru a determina parametrii $K_{CE}$ și $T'_{CE}$ .....	98
Fig. 3.5.9	Răspunsul $\theta_1(t)$ al vaporizatorului pentru o valoare referință $\theta_1=60^\circ\text{C}$ , pentru $K_{CE}=0.001\text{kg}\cdot\text{s}^{-1}$ , $T'_{CE}=30\text{s}$ , $q_{\text{CO}_2\text{-pv}}=60\text{kg}\cdot\text{s}^{-1}$ , $\theta_4=30.5^\circ\text{C}$ , $\theta_1(0)=33.5^\circ\text{C}$ . (a-utilizând structura veche de reglare, b-utilizând noua structură de reglare, c-reprezentare detaliată pentru noua structură de reglare) .....	100
Fig. 3.5.10	Evoluția deschiderii $h_1$ a valvei RB <sub>1</sub> , în condiții identice ca în figura 3.5.9 .....	100
Fig. 3.5.11	Evoluția mărimii de comandă $u_1$ a valvei RB <sub>1</sub> în condiții identice ca în figura 3.5.9 .....	101
Fig. 3.6.1	Poziția structurii propuse de reglare a temperaturii $\theta_1$ .....	102
Fig. 3.6.2	Structura sistemului de măsură a temperaturii $\theta_1$ relativ la regulatorul de temperatură .....	103
Fig. 3.6.3	Structura modificată a buclei de reglare a temperaturii $\theta_1$ din cadrul centralei electrice geotermale .....	103
Fig. 3.6.4	Valorile temperaturilor măsurate de traductorul principal, $\theta'_{1r}(a)$ , de traductorul de rezervă, $\theta'_{1r}(b)$ , semnalul de pe reacție utilizat în cadrul reglării temperaturii, $\theta'_{1m}(c)$ și temperatura reglată, $\theta_1(d)$ , pentru scenariul <i>TT<sub>1</sub> funcțional</i> , <i>TT<sub>1r</sub> defect</i> .....	105
Fig. 3.6.5	Valorile temperaturilor măsurate de traductorul principal $\theta'_{1r}(a)$ , de traductorul de rezervă $\theta'_{1r}(b)$ , semnalul de pe reacție utilizat pentru reglarea temperaturii $\theta'_{1m}(c)$ și temperatura reglată $\theta_1(d)$ , pentru scenariul <i>TT<sub>1</sub> funcțional</i> , <i>TT<sub>1r</sub> defect</i> .....	106
Fig. 3.6.6	Variația detaliată a ieșirii blocului de măsurare a temperaturii $\theta_1$ corespunzătoare scenariului al doilea .....	106
Fig. 3.7.1a	Caracteristica senzorului TT <sub>1</sub> .....	109
Fig. 3.7.1b	Bucle de reglare a temperaturii $\theta_1$ .....	109
Fig. 3.7.2	Caracteristica senzorului TT <sub>3</sub> .....	112
Fig. 3.7.3	Arhitectura MFDDM .....	114
Fig. 3.7.4	Bloc de achiziție de semnale .....	114
Fig. 3.7.5a	O tranziție pozitivă a semnalului $u_1$ .....	115
Fig. 3.7.5b	O tranziție negativă a semnalului $u_1$ .....	115
Fig. 3.7.6a	Bloc care compară intrarea cu o valoare nominală superioară .....	115
Fig. 3.7.6b	Bloc care compară intrarea cu o valoare nominală inferioară .....	115
Fig. 3.7.6c	Bloc care verifică egalitatea cu o valoare nominală plus un interval de toleranță .....	115
Fig. 3.7.7	Defect de tip TT <sub>1</sub> -NL .....	116
Fig. 3.7.8	Defect de tip TT <sub>3</sub> -N .....	117
Fig. 3.7.9	Defect de tip RB <sub>1</sub> -FC .....	117
Fig. 3.7.10	Defect de tip RB <sub>3</sub> -FON .....	118
Fig. 4.1.1	Element bipozițional cu histerezis .....	120

Fig. 4.1.2	Reglarea adaptivă utilizată în [127] .....	121
Fig. 4.1.3	Punctele care contribuie la interpolarea în trei dimensiuni .....	124
Fig. 4.1.4	Căutarea locului de sprijin esențial în domeniul de intrare .....	124
Fig. 4.1.5	Valoarea de sprijin esențială .....	125
Fig. 4.1.6	Tabelul tridimensional rezultat în urma tratării depășirilor .....	127
Fig. 4.2.1	Schemă de clasificare a defectelor la roboții mobili din medii cu hazard .....	129
Fig. 4.2.2	Robotul Inuktum .....	130
Fig. 4.2.3	Robotul iRobot .....	130
Fig. 4.2.4	Robotul Panther .....	130
Fig. 4.2.5	Componentele hardware și software ale robotului mobil .....	131
Fig. 4.2.6	Defectele sistemului robotizat pe perioada testării și semnificația lor .....	132
Fig. 4.2.7	Evoluția MTBF pentru roboții testați în [113] .....	133
Fig. 4.3.1	Încadrarea regulatorului tolerant la defecte în bucla de reglare .....	135
Fig. 4.3.2	Schema bloc asociată procesului condus în cadrul sistemului cu regulator interpolativ .....	136
Fig. 4.3.3	Schemă generală a structurii de reglare .....	137
Fig. 4.3.4	Schema bloc a regulatorului interpolativ propriu-zis .....	138
Fig. 4.3.5	Schema de simulare a evoluției robotului mobil .....	138
Fig. 4.3.6	Distanța obstacolului mobil față de robot .....	140
Fig. 4.3.7	Distanța parcursă de robotul mobil .....	140
Fig. 4.3.8	Evoluția vitezei robotului mobil .....	140
Fig. 4.3.9	Evidențierea redundanței bazate pe referință în cadrul regulatorului tolerant la defecte .....	141
Fig. 4.3.10a	Corecție de tip discontinuu (M&C–blocul de măsurare și configurare) .....	142
Fig. 4.3.10b	Corecție de tip continuu (M – bloc de măsurare, BC – bloc de corecție) .....	142
Fig. 4.3.11	Corecție de tip hibrid .....	143
Fig. 4.3.12	Poziția pe care se situează dezvoltarea ulterioară a celorlalte două elemente care contribuie la realizarea regulatorului tolerant la defecte (schema de detecție și diagnoză a defectelor și mecanismul de corecție a regulatorului propriu-zis) .....	143
Fig. 4.3.13	Senzorul bază .....	146
Fig. 4.3.14	Senzorul referință .....	146
Fig. 4.3.15	Schemă funcțională a redundanței bazate pe referință .....	147
Fig. 4.3.16	Organigramă pentru funcționarea redundanței bazată pe referință .....	147
Fig. 4.3.17	Schema logică aferentă blocurilor N .....	149
Fig. 4.3.18	Schema logică a funcționării blocului C <sub>1</sub> .....	150
Fig. 4.3.19	Schema logică a funcționării blocului C <sub>2</sub> .....	152
Fig. 4.3.20	Schema regulatorului tolerant la defecte .....	152
Fig. 4.3.21	Semnalul eronat și cel corect se folosesc de aceleași puncte de sprijin (indică spre același hiper-cub) .....	154
Fig. 4.3.22	Semnalul eronat și cel corect se folosesc de puncte de sprijin diferite .....	154
Fig. 4.3.23	Ajustările realizate asupra locurilor de sprijin din tabelul de interpolare în urma apariției unui defect evolutiv .....	155



Fig. 4.3.24	Distanța reală a robotului față de obstacolul mobil .....	157
Fig. 4.3.25	Distanța față de obstacolul mobil furnizată de senzorul bază ..	158
Fig. 4.3.26	Evoluția factorului de corecție $\alpha$ .....	158
Fig. 4.3.27	Evoluția vitezei robotului mobil în timp .....	158
Fig. 4.3.28	Distanța parcursă de robotul mobil în timp .....	159
Fig. 4.3.29	Distanța față de obstacolul mobil furnizată de senzorul bază ..	159
Fig. 4.3.30	Evoluția factorului de corecție $\alpha$ .....	160
Fig. 4.3.31	Evoluția vitezei robotului mobil în timp .....	160
Fig. 4.3.32	Distanța parcursă de robotul mobil în timp .....	160
Fig. 4.3.33	Distanța față de obstacolul mobil furnizată de sistemul de achiziție .....	161
Fig. 4.3.34	Distanța reală față de elementul perturbator .....	162
Fig. 4.3.35	Distanța față de al doilea obiect mobil procesată de regulator în safe-mode .....	162
Fig. 4.3.36	Evoluția factorului de corecție .....	162
Fig. 4.3.37	Evoluția vitezei robotului mobil .....	163
Fig. 4.3.38	Distanța parcursă de robotul mobil .....	163
Fig. 4.3.39	Evoluția factorului de corecție .....	164
Fig. 4.3.40	Distanța reală și distanța furnizată de sistemul de achiziție față de obstacolul mobil .....	164
Fig. 4.3.41	Distanța față de obstacolul mobil considerată de sistemul de reglare .....	164
Fig. 4.3.42	Distanța parcursă de robotul mobil .....	165
Fig. 4.3.43	Evoluția vitezei în funcție de timp .....	165
Fig. 4.3.44	Distanța reală față de obiectul mobil .....	166
Fig. 4.3.45	Distanța față de obstacolul mobil considerată de sistemul de reglare .....	166
Fig. 4.3.46	Distanța parcursă de robotul mobil .....	166
Fig. 4.3.47	Evoluția vitezei robotului mobil în timp .....	167
Fig. 4.3.48	Modelul stabilit pentru analiza disponibilității .....	168
Fig. 4.3.49	Evoluția disponibilităților pentru analiza realizată în situația 1 .....	169
Fig. 4.3.50	Schema Simulink asociată analizei disponibilității .....	169
Fig. 4.3.51	Evoluția disponibilităților pentru analiza realizată în situația 2 .....	170
Fig. 4.4.1	Robotul mobil X80 .....	171
Fig. 4.4.2	Comunicația robot-calculator .....	172
Fig. 4.4.3	Organigrama care ilustrează mișcarea robotului .....	173
Fig. 4.4.4	Ocolirea unui obstacol static .....	175
Fig. 4.4.5	Obiectul căutat .....	178
Fig. 4.4.6	Imaginea inițială (IM2) .....	179
Fig. 4.4.7	Imaginea după prima prelucrare .....	179
Fig. 4.4.8	Obiectul identificat în imagine .....	179
Fig. 4.4.9	Orientarea se modifică în funcție de orientarea curentă și de orientarea nouă .....	179
Fig. 4.4.10	Robotul în poziția inițială .....	181
Fig. 4.4.11	După modificarea orientării .....	181
Fig. 4.4.12	Observarea obstacolului .....	181
Fig. 4.4.13	Reorientarea în vederea ocolirii obstacolului .....	181
Fig. 4.4.14	Detectarea dispariției obstacolului .....	181
Fig. 4.4.15	Reorientarea pentru a evolua pe traiectoria optimă spre	

---

	punctul final .....	181
Fig. 4.4.16	Evoluția spre punctul țintă .....	182
Fig. 4.4.17	Evoluția spre țintă .....	182
Fig. 4.4.18	Atingerea punctului final .....	182
Fig. 4.4.19	Robotul în poziția inițială .....	183
Fig. 4.4.20	După modificarea orientării .....	183
Fig. 4.4.21	Observarea obstacolului .....	183
Fig. 4.4.22	Reorientarea în vederea ocolirii obstacolului .....	183
Fig. 4.4.23	Detectarea dispariției obstacolului .....	183
Fig. 4.4.24	Reorientarea pentru a evolua pe traiectoria optimă spre punctul final .....	183
Fig. 4.4.25	Momentul preluării și procesării informației furnizate de țintă..	184
Fig. 4.4.26	Corecția orientării în vederea obținerii traiectoriei corecte .....	184
Fig. 4.4.27	Drumul spre destinație .....	184
Fig. 4.4.28	Atingerea punctului final .....	184
Fig. 4.4.29	Modelul stabilit pentru analiza fiabilității și disponibilității .....	185
Fig. 4.4.30	Schema Simulink utilizată pentru analiza de fiabilitate și disponibilitate .....	186
Fig. 4.4.31	Evoluția fiabilității în prima situație analizată (situația 1) .....	186
Fig. 4.4.32	Evoluția fiabilității pentru a doua situație analizată (situația 2) .....	187
Fig. 4.4.33	Evoluția disponibilității operaționale pentru situația 3 .....	187
Fig. 4.4.34	Evoluția disponibilității operaționale pentru situația 4 .....	188
Fig. 4.4.35	Evoluția disponibilității inerente pentru situația 5 .....	188
Fig. A1	Modulele periferice ale procesorului de semnal digital .....	194
Fig. A2	Pinii multiplexorului MAX 307 .....	195
Fig. A3	Numărătorul CD4040BE .....	196
Fig. A4	Demultiplexorul CD4051BE .....	196

## Introducere

Dependența societății moderne de sistemele tehnice este evidentă și în continuă creștere. Fiecare sistem tehnic este destinat realizării unei sarcini. În acest context, dependabilitatea sistemului respectiv constă în abilitatea de a realiza acea sarcină care prezintă în mod justificabil încredere. În deceniile anterioare asigurarea ansamblului de cerințe asociate conceptului de dependabilitate se realiza într-o fază ulterioară proiectării funcțional-constructive a sistemelor. În prezent, datorită diversificării funcțiilor și creșterii complexității sistemelor, se consideră că aspectele de dependabilitate trebuie să orienteze proiectarea și să fie asigurate simultan cu cerințele funcțional-constructive.

Referitor la sistemele automate, problema dependabilității este esențială, afirmație susținută de sporirea preocupărilor și studiilor teoretice și practice care se axează pe dependabilitatea sistemelor automate sau pe diverse atribute ale acestora [43], [21]. Cea mai frecventă metodă de creștere a gradului de dependabilitate se bazează pe folosirea de structuri redundante, acestea constituind obiectul studiului unei părți din cadrul tezei.

Datorită diversității tipurilor de sisteme automate problema dependabilității se analizează din perspective diferite. Scopul comun din punct de vedere tehnic este creșterea gradului de disponibilitate. În acest context, teza abordează două domenii aplicative diferite: o centrală electrică geotermală și roboții mobili cu roți.

Costul în creștere al energiei în lume determină utilizarea de surse alternative de energie. O sursă alternativă de energie este energia geotermală care prezintă avantaje din punctul de vedere al acțiunii asupra mediului, al resurselor și al costului [85]. În acest context există o preocupare intensă pentru studiul diverselor aspecte cu privire la utilizarea energiei geotermale [51]. Mai mult de 20 de țări utilizează resursele geotermale pentru a obține energie electrică [30]. Posibilitatea de a lucra într-un colectiv preocupat de centralele electrice geotermale și lipsa unor studii sistematice a problemelor referitoare la dependabilitatea acestora au făcut ca unul dintre domeniile aplicative ale tezei să îl constituie tocmai aprofundarea unor aspecte de evaluare și creștere a disponibilității la o astfel de centrală.

Sistemele robotice reprezintă un domeniu de cercetare actual [124] care va continua să se intensifice din cauză că roboții tind să se dezvolte într-o așa măsură încât vor ajunge să influențeze nu numai transformarea radicală a proceselor de producție industrială, dar și apariția unor schimbări majore în viața cotidiană. Se estimează că amploarea transformărilor care vor surveni datorită dezvoltărilor și extinderii sistemelor robotice se va compara cu impactul tehnologiilor internet spre finalul secolului XX. În consecință, există și se implementează în continuare strategii de cercetare-dezvoltare care au ca domeniu de activitate sistemele robotice.

Dintre numeroasele contexte aplicative în cadrul tezei s-au aprofundat câteva aspecte referitoare la deplasarea roboților mobili cu roți. Cercetarea întreprinsă se referă la dependabilitatea acestor tipuri de sisteme, atribute ale dependabilității pentru roboții mobili fiind intens studiate la ora actuală [23], [104], [113] datorită diversității foarte mari a domeniilor de utilizare.

În cele ce urmează se prezintă structura tezei.

Capitolul 1 sintetizează elemente teoretice privitoare la dependabilitatea sistemelor automate, în principal aspecte referitoare la dependabilitate, fiabilitate, disponibilitate, toleranță la defecțiuni.

Capitolul 2 este destinat analizei și implementării unor structuri redundante. În prima parte, se realizează o clasificare a structurilor redundante după mai multe criterii și se caracterizează tipurile de structuri redundante. Plecând de la analiza întreprinsă, se propune, se dezvoltă și se implementează pe un procesor de semnal numeric structura cu redundanță de tip  $m-n-k$  hibridă. Un alt subiect tratat în acest capitol este studiul redundanței omogene cu  $k$  entități reparabile identice, în sensul analizei tranzițiilor de la o stare la alta și determinării relațiilor dintre ratele de tranziție a sistemului și ratele de tranziție individuale ale entităților componente, precum și determinarea ratei de reparare pentru structura serială. Ultima parte a capitolului extinde acest studiu în sensul admiterii de către structura redundantă a tranzițiilor simultane, și se particularizează relațiile dintre ratele de tranziție ale sistemului și ratele de tranziție individuale ale entităților componente pentru o structură cu două entități.

Capitolul 3 prezintă cercetări și contribuții ale autorului la studiul unor aspecte referitoare la reglarea și disponibilitatea unei centrale electrice geotermale. Prima parte conține o descriere succintă a centralei electrice geotermale. Ea este urmată de aplicarea pentru centrala electrică geotermală, cu scopul analizei disponibilității, a dezvoltărilor teoretice referitoare la redundanța omogenă cu  $k$  entități reparabile, studiul incluzând și situația în care se admit tranzițiile simultane. În continuarea capitolului se sintetizează elemente referitoare la regulatoarele interpolative, iar pe această bază se prezintă o soluție de reglare a temperaturii bazată pe interpolare, soluție menită să asigure atât o reglare performantă, cât și o creștere a disponibilității blocului vaporizatoare, parte a centralei electrice geotermale. Ultima parte a capitolului se referă la conceperea și implementarea unui modul mobil de detecție și diagnoză a defectelor pentru centrala electrică geotermală, menit să identifice defecte pe baza identificării unor corelații de tip injectiv. Modulul în sine este utilizabil și pentru alte sisteme automate.

Capitolul 4 realizează un studiu al unor aspecte referitoare la dependabilitatea roboților mobili. Prima parte a capitolului sintetizează structurile de reglare utilizate în cadrul sistemelor robotizate, iar a doua parte descrie defectele care pot apărea la roboții mobili. În cadrul părții a treia se prezintă dezvoltarea unui regulator interpolativ tolerant la defecte pentru roboții mobili cu roți și se întreprinde o analiză de disponibilitate referitoare la studiul realizat. Ultima parte a capitolului redă elaborarea și implementarea unei metode de corecție a direcției de deplasare a unui robot mobil cu roți prin preluarea și procesarea de imagine, studiul încheindu-se prin prezentarea unei analize de fiabilitate și disponibilitate cu privire la dezvoltările întreprinse.

Capitolul 5 prezintă concluziile, contribuțiile personale ale autorului și diseminarea rezultatelor din cadrul tezei.

Pentru elaborarea tezei s-au referit 129 titluri dintre care 10 aparțin autorului în calitate de unic autor sau coautor.

# 1. Dependabilitatea sistemelor automate

Pe parcursul acestui capitol sunt prezentate elemente teoretice legate de problematica dependabilității sistemelor automate. Sunt sintetizate aspecte referitoare la dependabilitate, fiabilitate, disponibilitate, defect, mentenanță, iar pe baza lor, noțiuni legate de proiectarea orientată pe calitate și de toleranța la defectiuni. În final sunt descrise pe scurt sistemele automate care au furnizat baza aplicativă a dezvoltărilor din teză.

## 1.1 Dependabilitatea

Dependabilitatea sistemelor automate reprezintă un domeniu de cercetare intensă și de actualitate [43], [50]. Conceptul de dependabilitate, precum și noțiunile de bază care furnizează termenii integrați în acesta sunt prezentate în [6], [69].

Definiția de bază a dependabilității este abilitatea de a furniza un serviciu care prezintă în mod justificabil încredere. Această definiție necesită o explicitare a termenului încredere. O definiție alternativă care argumentează semnificația termenului încredere este: un sistem este depenabil dacă are abilitatea de a evita, la nivelul serviciilor pe care le oferă, defectele de o severitate și de o frecvență mai mare decât gradul de acceptabilitate.

Dependabilitatea este un concept integrat care include următoarele atribute:

- disponibilitate
- fiabilitate
- siguranță
- integritate
- mentenabilitate

Dependabilitatea ca și concept nu are asociată o unitate de măsură, dar gradul de încredere este justificat pe baza atributelor.

## 1.2 Fiabilitate și disponibilitate

Dependabilitatea înglobează conceptele de fiabilitate și disponibilitate. Fiabilitatea la un moment dat  $t$  reprezintă probabilitatea unui sistem de a funcționa cu succes în intervalul de timp  $[0, t]$  în condițiile respectării unor restricții de utilizare. Fiabilitatea [47],  $R(t)$ , este evident o funcție de timp. O valoare a fiabilității exprimată fără precizarea unui interval de timp pe parcursul căruia este valabilă afirmația asociată este fără nici o semnificație. Totodată, este clar, din definiție, că teoria fiabilității necesită stăpânirea teoriei probabilităților.

Funcționarea sigură a unui sistem implică în primul rând un nivel ridicat de fiabilitate. Nivelul ridicat de fiabilitate înseamnă o valoare a lui  $R$  cât mai apropiată de valoarea 1 pe orizontul de timp care interesează.

Din punctul de vedere al posibilității de reparare, sistemele se împart în trei categorii:

- *sisteme nereparabile*, care permit funcționarea până la apariția primului defect,
- *sisteme parțial reparabile*, care în cazul unor tipuri de defecte pot să fie recuperate și să continue funcționarea, dar în cazul altor defecțiuni intră în stări care nu mai permit continuarea operării,
- *sisteme total reparabile*, care pot să fie recuperate în cazul oricărui defect.

Asupra definirii termenilor „defect” și „defecțiune” se va reveni în cadrul subcapitolului 1.3.

Datorită ipotezei de funcționare continuă, fiabilitatea se utilizează pentru sistemele nereparabile sau parțial reparabile. Experimental și din perspectivă statistică, în contextul conceptului de fiabilitate se folosește ca punct de plecare timpul mediu până la defectare (*MTTF*) care înseamnă intervalul de timp scurs până la defectarea sistemului.

$$MTTF = \frac{1}{N} \sum_{i=1}^N T_i \quad (1)$$

unde:

- *N*-numărul de sisteme monitorizate
- *T<sub>i</sub>*-timpul de viață al fiecărui sistem

Non-fiabilitatea, *F(t)*, este complementara fiabilității și se definește ca probabilitatea sistemului de a se defecta în intervalul de timp  $[0, t]$ . Astfel,

$$F(t) = 1 - R(t) \quad (2)$$

Sistemele total reparabile [47], [73] sunt predominante în industrie, iar în cazul acestora în loc de fiabilitate se folosește ca mărime caracteristică disponibilitatea *A(t)* care se definește ca fiind probabilitatea de funcționare cu succes (în limitele caracteristicilor de performanță impuse) a sistemului la momentul de timp *t*. Din punct de vedere aplicativ interesează atât valorile momentane *A(t)* cât mai ales valoarea de regim staționar:

$$A_{\infty} = \lim_{t \rightarrow \infty} A(t) \quad (3)$$

Disponibilitatea este o măsură a succesului pentru sistemele reparabile, iar în cazul sistemelor nereparabile se confundă cu fiabilitatea.

Non-disponibilitatea *U(t)* este complementara disponibilității, definindu-se ca o măsură a non-succesului sistemului la momentul de timp *t*.

$$U(t) = 1 - A(t) \quad (4)$$

Experimental și din perspectivă statistică, în cazul sistemelor reparabile se operează, ca și indicator de fiabilitate, cu mărimea *MTBF* reprezentând timpul mediu între defecțiuni, care se calculează cu formula:

$$MTBF = \frac{\text{Timpul total de nedefectare}}{N_F} = \frac{N \cdot T - N_F \cdot MDT}{N_F} \quad (5)$$

unde:

- *N*-numărul de sisteme identice monitorizate pe intervalul de timp *T*

- $N_F$  - numărul total de defecțiuni monitorizate pe intervalul de timp  $T$  pentru cele  $N$  sisteme
- $MDT$  - timpul mediu cât durează un defect pe intervalul de monitorizare.

Pentru sistemele reparabile se folosește și o a doua mărime, timpul mediu de reparare  $MTTR$ , care reprezintă doar o parte din  $MDT$  și anume timpul necesar exclusiv reparării sistemului.

Disponibilitatea este influențată în mod evident de rata de reparare. Elementul care intră în calculul ratei de reparare este după [14]  $MDT$ , iar după [21]  $MTTR$ . În acest context, în [121] se definesc mai multe tipuri de disponibilități:

i) *Disponibilitatea inerentă ( $A_{inerentă}$ )*

În calculul disponibilității inerente se consideră doar  $MTTR$ , adică doar repararea propriu-zisă a sistemului. Se presupune că elementul uman, elementele de rezervă și uneltele sunt disponibile 100%, fără întâzieri, excluzându-se astfel toate perioadele de timp conexe procesului de reparare propriu-zis. Disponibilitatea inerentă are următoarea formulă:

$$A_{inerentă} = \frac{MTBF}{MTBF + MTTR} \quad (6)$$

ii) *Disponibilitatea rezultată ( $A_{rezultată}$ )*

Acest tip de disponibilitate se asociază implicit cu mentenanța preventivă (vezi paragraful 1.4.2). Disponibilitatea rezultată presupune aceleași aspecte ca și disponibilitatea inerentă cu diferența că timpii necesari reparării sunt considerați timpii de cădere asociați acțiunilor preventive (acești timpii sunt mai mari decât cei considerați pentru disponibilitatea inerentă). Formula de calcul pentru disponibilitatea rezultată uzează de timpul mediu dintre acțiunile de mentenanță preventivă ( $MTBM$ ) și de timpul mediu de cădere din timpul mentenanței preventive ( $MPDT$ ):

$$A_{rezultată} = \frac{MTBM}{MTBM + MPDT} \quad (7)$$

iii) *Disponibilitatea operațională ( $A_{operațională}$ )*

Disponibilitatea operațională se definește prin următoarea formulă și reprezintă raportul dintre timpul în care sistemul funcționează cu succes și timpul total de funcționare:

$$A_{operațională} = \frac{MTBF}{MTBF + MDT} \quad (8)$$

Disponibilitatea operațională este disponibilitatea pe care utilizatorul (clientul) o experimentează în mod real.  $MDT$  reprezintă în mod real timpul în care sistemul stă în starea de defectare. Evident disponibilitatea operațională poate fi îmbunătățită alegând un tip de mentenanță convenabil.

Abordarea considerată de [21] în calculul disponibilității nu este relevantă deoarece în studiul disponibilității trebuie să se aibă în vedere  $MDT$ . După cum se precizează și în [22] disponibilitatea inerentă (furnizată de respectivele lucrări) trebuie să fie completată cu indicatorul  $MDT$  pentru a prezenta relevanță în

studiul dependabilității sistemelor. Pe parcursul tezei se va utiliza termenul de *disponibilitate în accepțiunea de disponibilitate operațională*.

Analiza fiabilității unui sistem se poate face din diverse perspective prin diferite metode. Câteva dintre aceste metode sunt [14], [47], [34]:

- metoda arborelui defectelor (*FTA*),
- metoda analizei defectelor mod-efect (*FMEA*),
- analiza pe baza arborelui evenimentelor (*ETA*),
- modele Markov.

Sistemele sunt formate din module, care la rândul lor sunt formate din componente. Din punctul de vedere al fiabilității și disponibilității condiționările din sistemele reale se surprind prin modele de fiabilitate sau disponibilitate (numite și rețele), a căror evaluare se bazează pe reguli probabilistice. Pe scurt se va prezenta modalitatea în care este influențată disponibilitatea în cazul unor modele de bază (serie, paralel).

În domeniul fiabilității, structura serie reprezintă modelul asociat sistemelor fără redundanțe caracterizate prin faptul că defectarea oricărui element component compromite funcționarea sistemului. Prin urmare este vorba despre situații cu un risc mare de defectare, respectiv cu o siguranță scăzută în funcționare. Din acest motiv, o structură serie trebuie să fie constituită din componente cu un grad mare de fiabilitate pentru a prezenta un grad de siguranță acceptabil. Structura serie se simbolizează ca în figura 1.1.1, orice defectare a unei componente fiind echivalentă cu întreruperea lanțului (funcționării sistemului). Probabilitatea ca sistemul să funcționeze la momentul de timp  $t$  este dată de probabilitatea ca toate componentele să funcționeze la momentul respectiv.

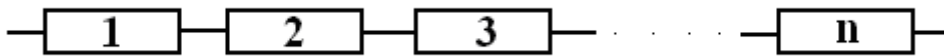


Fig. 1.1.1 Structură serie cu  $n$  componente

Notând cu  $A_i(t)$  disponibilitatea componentei  $i$ , disponibilitatea structurii serie este:

$$A_s(t) = \prod_{i=1}^n A_i(t) \quad (9)$$

Structura paralel reprezintă conceptual modelul de bază pentru noțiunea de redundanță, respectiv toleranță la defecțiuni. Ea redă, de regulă, cazul în care 2 (sau  $k$ ) entități sunt apte pentru a îndeplini aceeași funcție în sistem (sistemul prezintă redundanță), disponibilitatea întregului sistem crescând. Structura este ilustrată în figura 1.1.2. Există două modalități de percepere a structurii paralel: funcționarea sistemului necesită funcționarea unei singure entități, respectiv defectarea sistemului este echivalentă cu defectarea tuturor entităților componente. Problematika structurilor redundante este dezvoltată în capitolul 2, paragraful 2.1.4.2. În acest context mai jos sunt precizate câteva elemente referitoare la structura paralel de tip standby hot (vezi paragraful 2.1.4.4). Astfel, non-disponibilitatea ( $U_P$ ), respectiv disponibilitatea ( $A_P$ ) acestei structuri se calculează în funcție de non-disponibilitățile entităților componente cu formulele:

$$U_P(t) = \prod_{i=1}^n U_i(t) \quad (10)$$



$$A_P(t) = 1 - U_P(t) \quad (11)$$

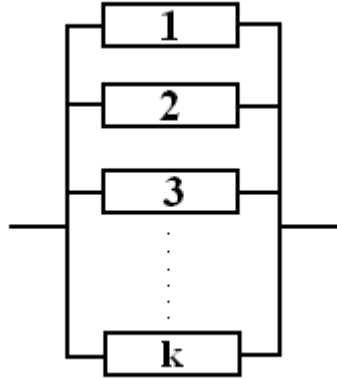


Fig. 1.1.2 Conexiunea paralel

Dacă se cunosc ratele de defectare și reparare ale entităților componente, disponibilitatea se poate calcula cu ajutorul modelelor Markov. Calculul se bazează pe conceptele de stare (stare funcțională sau nefuncțională) și de tranziție (defectare sau reparare). Ca exemplu se consideră o singură entitate cu rata de defectare  $\lambda_{01}$ , respectiv rata de reparare  $\mu_{10}$ . Entitatea se poate găsi în timpul funcționării în două stări,  $\textcircled{0}$  stare de succes (funcțională) și  $\textcircled{1}$  stare de defectare (nefuncțională). Stările și tranzițiile sunt simbolizate în fig. 1.1.3.

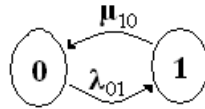


Fig. 1.1.3 Model Markov cu două stări

Plecând de la faptul că disponibilitatea este reprezentată de probabilitatea sistemului de a se afla în starea de succes, iar non-disponibilitatea de probabilitatea sistemului de a se găsi în starea de defectare, prin calcule bazate pe un model Markov, se obțin, de exemplu, următoarele formule pentru disponibilitatea și non-disponibilitatea în regim permanent constant:

$$A_\infty = \frac{\mu_{10}}{\lambda_{01} + \mu_{10}}, \quad U_\infty = \frac{\lambda_{01}}{\lambda_{01} + \mu_{10}} \quad (12)$$

### 1.3 Defectul

De cele mai multe ori defectul este perceput prin următoarea definiție: un element este defect dacă este în imposibilitatea de a funcționa. În acest context [47] a clasificat defectele în defecte fizice și defecte funcționale:

- defectele fizice datorate incidentelor survenite la nivelul entităților sau ansamblului ca urmare a condițiilor de funcționare, a materialelor primare etc.;

- defecte funcționale care survin în urma unor greșeli de proiectare, de mentenanță, sau de instalare.

Tot pe această definiție s-a axat și clasificarea surselor defectelor:

- interne
  - defecte de proiectare
  - defecte de producție
- externe
  - mediu
  - de mentenanță
  - operaționale

Definiția dată neprecizând ce înseamnă „a funcționa” este relativ vagă. O definiție mai bună ar fi: un element este considerat defect atunci când se află în imposibilitatea de a funcționa corect. Mai clar, un sistem este considerat defect dacă se află în incapacitatea de a atinge caracteristicile de performanță, adică de a se afla în limitele de specificație.

Defectul poate fi clasificat și dintr-o altă perspectivă ca fiind brusc sau evolutiv, așa cum se descrie în cadrul paragrafului 4.3.2, sau ca fiind periclitant sau nepericlitant, așa cum se descrie în paragraful 4.2.1.

## 1.4 Mentenanța

Prețul relevant pe care îl plătește consumatorul atunci când folosește un produs este prețul pe întreg ciclul de viață (*total lifetime operation cost - TLOC*). TLOC se compune din costul inițial la care se adaugă costurile defectelor și mentenanței. Astfel, strategia de mentenanță a produsului este esențială în calculul costului final.

Acest subcapitol prezintă caracteristicile celor trei tipuri de mentenanță de bază [14]: mentenanța breakdown, mentenanța preventivă și mentenanța on-condition.

### 1.4.1 Mentenanța breakdown

Mentanța breakdown este denumită în domeniu și mentenanță neplanificată sau la reparare. Ea constă în repararea sau înlocuirea echipamentului în momentul în care acesta se defectează.

Mentanța breakdown se întâlnește la entități pentru care timpul de cădere este în comparație cu alte tipuri de mentenanță cel mai mare, repararea necesitând de regulă parcurgerea a numeroși pași care conduc la reparare sau înlocuire, fiecare având alocată o perioadă de timp: timpul până când defectul devine aparent, timpul de acces, timpul de diagnoză (cel mai mare), timpul necesar pentru logistică, timpul de reparare, înlocuire, timpul de verificare etc.

Costul total al defectelor se calculează după formula:

$$C_{total\_defecte} = [C_R + (C_L + C_{PB}) \cdot MDT] \cdot \lambda \cdot T \quad (13)$$

unde:

$C_R$  – costul mediu al materialelor pentru reparare

$C_L$  – costul mediu al echipei de reparare pe oră

$C_{PB}$  – costul procesului pe ora la defectare

$\lambda$  – rata de defectare  
 T – perioada de funcționare

O altă deficiență este faptul că utilizatorul produsului prevăzut cu o strategie de mentenanță breakdown nu poate reduce în nici un fel rata de defectare.

### 1.4.2 Mentenanța preventivă

Mentenanța preventivă se mai numește și mentenanță planificată, implicând service-ul sau înlocuirea echipamentului la intervale regulate de timp. Costul total al mentenanței preventive este:

$$C_{total\_preventiva} = [C_M + (C_L + C_{PP}) \cdot MMT] \cdot m \cdot T \quad (14)$$

unde:

$C_M$  – costul mediu al materialelor pentru service  
 $MMT$  – timpul mediu de mentenanță  
 $C_{PP}$  – costul pierderilor procesului pe ora pierdută  
 $m$  – frecvență de mentenanță

Mentenanța preventivă este caracterizată printr-o reducere semnificativă a numărului de defecțiuni. Costul unei defecțiuni în proces (în cazul breakdown) este mai mare decât costul rutinei de verificare, o scoatere din funcțiune neplanificată implicând pierderi însemnate.

Pe de altă parte și această strategie de mentenanță are deficiențele ei. De exemplu, un defect care nu a fost analizat și inclus în rutina de intervenții, poate apărea oricând, iar în acest caz costurile sunt ca și în cazul mentenanței breakdown.

Strategiile de mentenanță preventivă sunt de două categorii:

- strategii bazate pe înlocuirea de echipamente la intervale regulate de timp
- strategii bazate pe verificarea sistemelor la anumite intervale de timp, urmată atunci când este cazul, de înlocuirea de echipamente

Primul caz poate conduce la risipă de materiale, mai precis dacă un element nu este încă uzat el va fi schimbat conform strategiei de rutină. Al doilea caz se poate solda cu defecte care nu sunt observate la verificare, sau nu sunt incluse în gama de analiză. În ambele cazuri elementul esențial este alegerea perioadei la care să se efectueze rutina preventivă.

### 1.4.3 Mentenanța on-condition

Mentenanța on-condition are la bază următorul principiu: starea echipamentului este monitorizată periodic sau continuu, iar când prezintă semne de uzură, sau alte indicii de defecțiune iminentă, se intervine prin reparare sau înlocuire. Monitorizarea sistemului permite detectarea cazurilor de defectare iminentă. Mentenanța on-condition este practic identică din punctul de vedere al intervenției cu mentenanța breakdown, cu excepția momentului în care se face intervenția asupra sistemului. În cazul mentenanței breakdown, intervenția asupra echipamentului se face după apariția defectului, pe când în cazul mentenanței on-condition reparația sau înlocuirea se produce înainte de producerea defecțiunii.

Se poate spune că mentenanța on-condition prezintă avantaje evidente față de celelalte două tipuri, în situația în care monitorizarea este realizată corect [48].

Principala problemă care apare în această situație este posibilitatea de a întâmpina greutăți în procesul de observare a sistemului.

Mentenanța on-condition se realizează în două variante:

- cu monitorizare condiționată periodică,
- cu monitorizare continuă.

În primul caz se definește noțiunea de interval de potențial defect, care reprezintă perioada în care este probabil să se producă o defecțiune iar sistemul este monitorizat în funcție de acel interval de timp.

### 1.5 Elemente de proiectare orientată pe calitate

Funcționarea sigură și performanța ridicată a oricărui sistem (produs) implică utilizarea de componente de o calitate superioară. Calitatea unui produs sau serviciu în general se definește ca și capacitatea acestuia de a asigura satisfacerea într-o măsură cât mai completă a consumatorului. Termenul de calitate după standardul ISO 9000-2000 este măsura în care un ansamblu de caracteristici intrinseci îndeplinește cerințele. Calitatea în termeni ingineresti [14] poate să fie măsurată printr-un număr de caracteristici (caracteristici de performanță) care contribuie la satisfacerea necesităților consumatorului.

Calitatea sistemului trebuie să fie menținută pe tot parcursul ciclului de viață, adică produsul trebuie să realizeze caracteristicile de performanță de la începutul până la sfârșitul perioadei de funcționare. Aceste caracteristici de performanță pot fi:

- continue (măsurabile, obiective, nu depind de utilizatorul produsului)
- discrete (subiective, în funcție de opțiunea fiecărui utilizator)
- binare (atribute, care sunt sau nu prezente).

Având în vedere că dintre cele trei tipuri de caracteristici, singurele măsurabile și obiective sunt caracteristicile continue, în studiul siguranței în funcționare se analizează, în principal, obținerea acestor caracteristici de performanță continue care trebuie să fie cât mai bune, pe toată durata funcționării.

În urma cerințelor pieței, producătorul stabilește un set de specificații care definesc produsul pentru potențialul client. Aceste specificații conduc la impunerea caracteristicilor de performanță continue sub forma unor valori țintă  $x_{1T}$ ,  $x_{2T}$ , ...,  $x_{nT}$  și a unor domenii limită.

Produsele ingineresti sunt compuse din subcomponente care au fiecare o limită de suportabilitate cu privire la mediul în care vor fi amplasate, mediu care acționează și are efect aleator asupra produsului. Vis-a-vis de produs, aceste efecte se consideră ca intrări aleatoare de care trebuie să se țină seamă și care influențează caracteristicile continue de performanță. Defectele pot să apară în momentul în care o formă a stresului depășește gradul de rezistență (suportabilitate) al produsului. Stresul reprezintă influența negativă a diverselor elemente din mediu asupra produsului, iar susceptibilitatea reprezintă sensibilitatea acestuia la factorul de stres. Stresul poate proveni de la mai multe componente care pot să acționeze separat sau în diverse combinații asupra sistemului. Susceptibilitatea și rezistența produsului la factorii de stres sunt rezultate ale proiectării. Modulele cu un factor ridicat de siguranță prezintă o rezistență mai mare la factorii de stres, adică o susceptibilitate mai mică a produsului respectiv.

Pentru o analiză corectă, așa cum s-a precizat, se definesc de la început limitele de posibilă variație a mărimilor în jurul valorilor țintă. Astfel se specifică pentru fiecare caracteristică de performanță o limită superioară și una inferioară

între care sistemul se consideră că funcționează cu succes. Acestea se numesc limite de specificație,  $X_{IT} \in [X_{imin}, X_{imax}]$ .

Caracteristicile produsului pot suferi modificări din următoarele cauze:

- variația calității materialelor și componentelor utilizate
- variația calității procesului de fabricație

În principiu un model al calității utilizat în proiectarea orientată pe calitate are în vedere relația a trei mărimi:

$$\{x\} = f[\{p\}, \{r\}] , \quad (15)$$

unde  $\{x\}$ ,  $\{p\}$ ,  $\{r\}$  reprezintă pe rând vectori ai caracteristicilor de performanță continue, ai parametrilor de proiectare și ai intrărilor aleatoare considerate ca variabile.

Implementarea unui astfel de model al calității înseamnă obținerea unor caracteristici de performanță continue cât mai bune ale produsului. În cadrul acestei construcții se fixează diferite valori pentru intrările aleatoare și în mod recursiv se determină parametri optimi de proiectare pentru a satisface criteriile de performanță. În urma analizelor statistice [14] se determină dacă un produs este acceptat sau nu, dacă atinge performanțele dorite. Procesul de proiectare este ilustrat în figura 1.1.4. În figură, *USL* și *LSL* reprezintă limitele de specificație (limita superioară, respectiv inferioară).

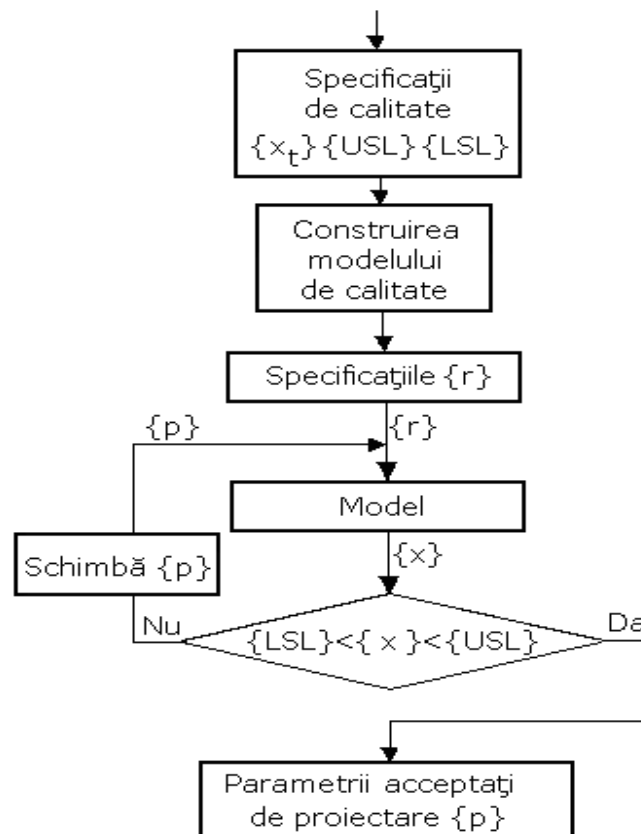


Fig. 1.1.4 Procesul de proiectare

## 1.6 Toleranța la defecțiuni

Prin sistem tolerant la defecțiuni se înțelege un sistem capabil să funcționeze în cadrul specificațiilor inițiale sau într-un regim special, atunci când sunt prezente una sau mai multe defecțiuni. O astfel de capabilitate se obține prin structuri și funcțiuni adecvate.

Orice cădere a unui sistem în cazul unor defecte periclitanțe înseamnă pierderi. Tendința actuală este de a elimina aceste pierderi prin proiectarea sistemelor astfel încât ele să funcționeze în ciuda apariției defectului. Aceasta implică în primul rând capabilitatea de a detecta și a localiza automat defectele apărute și de a proteja sistemul precum și mediul înconjurător. Instrumentul principal de realizare a toleranței la defecțiuni este introducerea în sistem a structurilor redundante, descrise pe larg în capitolul 2.

Evident, nu în orice situație elementul suplimentar care oferă sistemului proprietatea de toleranță la defecte este identic cu elementul defect. Astfel, există cazuri în care funcția este preluată de un alt tip de element, lucru care înseamnă trecerea sistemului într-o cu totul altă stare, admisibilă, dar de multe ori mai puțin eficientă decât cea inițială, numită stare de siguranță. Pe perioada evoluției în această stare, un sistem complet poate să permită efectuarea de reparații și înlocuiri, astfel încât structura să poată reveni în starea inițială, ieșind astfel din starea de siguranță.

Structura unui sistem tolerant la defecțiuni are după [14], aspectul din figura 1.1.5:

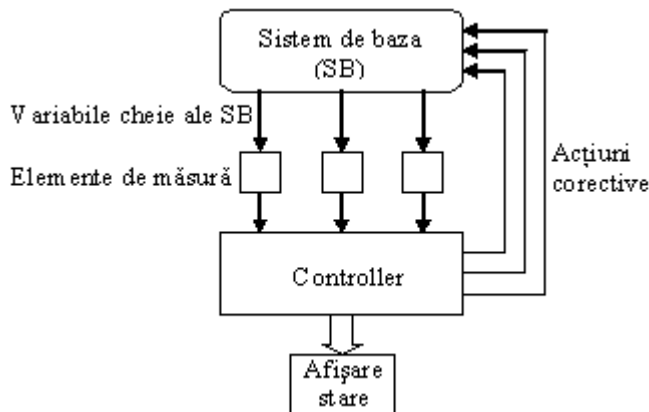


Fig. 1.1.5 Sistem tolerant la defecțiuni destinat realizării funcției sistemului de bază SB

Sistemul de baza, SB, reprezintă un sistem destinat realizării unei funcții specificate. În el se produc defecțiuni. SB conține elemente de reconfigurare care pot asigura un caracter redundant. Principale trăsături prin care se asigură toleranța la defecțiuni a structurii sunt:

- Efectuarea unor operații de măsurare a unor mărimi din SB, numite variabile cheie și care sunt relevante din punctul de vedere al stărilor sistemului afectabile de către defecțiuni.
- Variabilele cheie măsurate sunt verificate de un controller. Verificarea înseamnă stabilirea faptului dacă ele sunt situate între limitele specificate. Fiind variabile continue, putem considera asociată o scară de valori pentru fiecare variabilă. Pe această scară, controllerul decide dacă

- valoarea este periclitantă (de exemplu, peste limita maximă a domeniului specificat) sau nepericlitantă (sub limita minimă a domeniului specificat). Dacă mărimea se încadrează în domeniul specificat se consideră că variabila cheie are o valoare normală, deci starea respectivă este normală. De regulă la nivelul fiecărei variabile cheie se folosește o codare prin 3 valori (sub limita inferioară, între limite și peste limita superioară), iar pe aceasta baza, dintr-un tabel multidimensional care îmbină toate variabilele se obține în mod automat starea SB-ului, iar corespunzător ei acțiunile corective necesare. Prin aceasta controlerul realizează și o diagnosticare automată.
- Sistemul este capabil de acțiuni corective comandate de către controler. Ele se realizează prin intermediul elementelor redundante comandabile aflate în SB.
- Sistemul este capabil să afișeze starea diagnosticată. De regulă, se afișează și acțiunea corectivă întreprinsă.

## 1.7 Sistemele automate

Lucrarea de față are ca scop analiza unor aspecte ale dependibilității în cazul unor sisteme automate. Sistemele automate sunt de mai multe tipuri, fiecare tip având caracteristici proprii. Elementul comun al acestora în momentul în care se pune problema creșterii gradului de dependabilitate este redundanța (vezi capitolul 2). Datorită specificului diferitelor tipuri de sisteme automate, modalitatea de aplicare și tipul de redundanță aplicat diferă semnificativ de la un sistem la altul.

Cu scopul ca pentru diversitatea de sisteme automate cercetarea să fie cât mai concludentă și să acopere o arie cât mai largă de aplicabilitate se va recurge la divizarea studiului. O parte a cercetării se axează pe o centrală electrică geotermală, iar o altă parte tratează roboții mobili cu roți.

Energia geotermală reprezintă o sursă alternativă de energie în contextul în care există o tendință generală de creștere a costului energiei pe plan mondial. Energia geotermală oferă avantaje certe din punctul de vedere al acțiunii asupra mediului, al resurselor și al costului. Implementarea și utilizarea energiei geotermale în ultimele decenii este prezentată în [76], [77].

Una din posibilitățile de utilizare a energiei geotermale este de a o transforma în energie electrică. Energia geotermală se convertește în energie electrică prin centralele electrice geotermale. Se studiază intens în acest domeniu, cercetări semnificative fiind referite în [51], [85], [31], [3]. Furnizarea de energie electrică provenită din energie geotermală, precum și impactul pe care îl are procesul asupra mediului a fost studiată în [30], [27], [31], [51]. La ora actuală 24 de țări produc energie electrică provenită din resurse geotermale, dar cantitatea de energie provenită din astfel de resurse este încă mică în raport cu totalul energiei produse pe glob și potențialul existent. Creșterea procentului de energie provenită din resursele alternative de energie, printre care și cele geotermale, reprezintă tendința de evoluție pentru viitor.

Cercetarea axată pe centrala electrică geotermală de la Universitatea din Oradea s-a realizat în cadrul unui colectiv de cercetare iar rezultatele diseminate pe parcursul tezei prezintă contribuțiile autorului din această perspectivă.

Al doilea tip de sistem automat analizat în teză este un robot mobil cu roți. Evident, sistemele robotice reprezintă o direcție actuală de cercetare [87], [1], iar preocuparea în domeniul roboților mobili cu roți este intensă [124], [29]. Analiza

diverselor atribute ale dependabilității reprezintă de asemenea actualitate și importanță [104], [113], [114]. Capitolul 4 al tezei este dedicat studiului elementelor legate de dependabilitatea roboților mobili.



## 2. Studiu privind analiza și implementarea unor structuri redundante

După [70] redundanța are următoarea definiție <sup>1)</sup>:

- multiplicare a informației pentru a garanta siguranța transmisiei,
- utilizare a mai multor structuri de elemente hardware și/sau software pentru preluarea unei funcții sau repetarea unei operații.

Redundanța în conformitate cu definiția furnizată în [83] este proprietatea unui sistem de a dispune pentru îndeplinirea unei funcții prevăzute de un număr de mijloace tehnice în stare de funcționare mai mare decât cel necesar pentru îndeplinirea funcției prevăzute.

Redundanța înseamnă în principiu capacitatea unui element (modul, sistem), vis-a-vis de un ansamblu, de a prelua sarcinile unui alt element în cazul în care primul se defectează. Defectul a fost definit în subcapitolul 1.3. În același timp redundanța conferă „protecție funcțională”, un sistem redundant fiind un sistem capabil să suporte anumite defecțiuni, detectându-le și tratându-le prin reconfigurare. Redundanța aduce o creștere a fiabilității, disponibilității, conferindu-i astfel sistemului o funcționare sigură.

Așa cum s-a arătat și în subcapitolul 1.6 conceptele de toleranță la defecte și redundanță sunt în strânsă legătură, un sistem fiind tolerant la defecte atunci când este apt de a-și continua funcționarea la standardele impuse chiar și după apariția unor defecțiuni. Aceasta implică, în primul rând, capacitatea de a detecta și de a localiza defectele apărute și de a proteja sistemul precum și mediul înconjurător. În al doilea rând, se presupune că toleranța la defecte înseamnă reconfigurarea sistemului prin izolarea într-o formă sau alta a elementului defect față de lanțul funcțional, respectiv transferarea sarcinii de realizare a funcției unui alt element disponibil și continuarea funcționării corecte în ciuda problemelor apărute. În consecință, toleranța la defecte se obține în principal prin redundanță.

În cadrul proiectării, pentru ca produsul să atingă standardele impuse (vezi fig. 1.1.4 din cadrul subcapitolului 1.5) și de asemenea să fie sigur și robust, fiabilitatea, respectiv disponibilitatea lui se controlează prin antecalcul. Astfel, când se observă că anumite elemente au o rată mare de defectare, ceea ce înseamnă o fiabilitate redusă, în general, se aplică un anumit tip de redundanță pentru a mări siguranța sistemului și pentru a reduce influența defectelor asupra ansamblului, în locurile unde se găsesc elementele respective.

Este discutabil care dintre următoarele două situații este mai viabilă: folosirea componentelor cu o fiabilitate ameliorată sau utilizarea redundanței. După [11], soluția se prezintă în felul următor: în prima perioadă a utilizării unui ansamblu este de preferat o structură redundantă, pentru ca ulterior componentele rămase considerate ca având o fiabilitate ameliorată să se prezinte mai sigure. Trebuie însă menționat că de multe ori acele componente cu fiabilitate ameliorată nu există sau costă foarte mult.

---

<sup>1</sup> Termenul de redundanță provine din teoria informației și a fost introdus și definit de Shannon.

În contextul problematicei prezentate mai sus, în prima parte a capitolului se realizează o clasificare a structurilor redundante după mai multe criterii. Într-o a doua parte se prezintă dezvoltări teoretice și practice pentru structuri redundante, iar pe această bază, în partea a treia, se analizează structura redundanță activă cu  $k$  componente reparabile identice. Dezvoltările se încheie cu studiul tranzițiilor simultane în cazul structurii reparabile cu redundanță activă.

## 2.1 Clasificarea structurilor redundante

### 2.1.1 Lucrări semnificative

Există mai multe tipuri de redundanță și ele se aplică în funcție de necesități și posibilități. Datorită multiplelor aspecte prin care diferitele tipuri de redundanțe se caracterizează din punct de vedere conceptual și din punct de vedere practic clasificarea acestora este greu de realizat, iar numeroasele încercări din literatură rezolvă numai parțial problema clasificării.

În acest context se prezintă în cadrul subcapitolului 2.1 o clasificare mai cuprinzătoare decât cele cunoscute din literatură, și ea perfectibilă.

Lucrările lui Avizienis și Laprie [7], [8], [9] furnizează niște definiții și o clasificare de început a tipurilor de redundanță pentru sisteme de calcul. Autorii considerau calculatorul ca un ansamblu hardware-software în care au loc în timp interacțiuni. Se notează cu  $H$  și  $S$ , redundanța hardware, respectiv software iar cu  $T$  redundanța de timp. Atunci, în opinia autorilor un sistem de calcul tolerant la defecte este descris sub forma:  $\langle mH, nS, xT \rangle$ , unde  $m$ ,  $n$  și  $x$  reprezintă nivelele proiectate de redundanță hardware, software și de timp.

Se vor da câteva exemple pentru a înțelege conceptul prezentat anterior. Dacă sistemul de calcul are două dispozitive înlocuibile (identice sau diferite) atunci nivelul de redundanță hardware este  $m = 2$ . Dacă de exemplu în sistem există 5 versiuni ale unui program software (identice sau diferite) atunci sistemul tolerant la erori arată astfel:  $\langle ?H, 5S, xT \rangle$ . Se observă „?” înainte de  $H$ , ceea ce reprezintă faptul că cele 5 versiuni ale programului pot fi procesate pe aceeași structură hard sau pe unități diferite, deci redundanța hardware nu mai conține, iar „ $x$ ” înainte de  $T$  reprezintă faptul că redundanța de timp este nedeterminată.

Un alt exemplu ar fi dacă sistemul de calcul tolerant la defecte este descris prin  $\langle ?H, ?S, 2T \rangle$ . În acest caz se va permite repetiția unui program, deci sistemul de calcul este prevăzut cu redundanță de timp. Din nou „?” reprezintă necunoașterea nivelului de redundanță pentru  $H$  și  $S$ .

Avizienis și Laprie observă că tipurile de redundanță diferă prin aplicabilitate și prin proprietățile prin care furnizează toleranța la defecte.

Se observă că această clasificare are curențe, printre care:

- i) Nu se cunoaște absolut nimic în detaliu referitor la tipurile de redundanță. Singurele elemente cunoscute sunt: nivelul redundanței (nu întotdeauna al tuturor celor trei tipuri simultan) și clasa din care aparține (hardware  $H$ , software  $S$  sau de timp  $T$ ).
- ii) Clasificarea nu face diferența între unitățile hardware identice și cele diferite care realizează redundanța (de asemenea pentru software).
- iii) Nu se cunoaște nimic despre platforma (identică sau diferită) pe care se realizează redundanța software prin  $n$  versiuni, etc.

În esență, Avizienis și Laprie introduc o manieră de descriere și etichetare. Prin eforturi ulterioare, ei reușesc să extindă termenii prin care să fie definit și descris un sistem de calcul tolerant la defecte luând în considerare și greșelile utilizatorului, erorile software și defectele hardware. Ei ajung în final să definească sistemele dependabile.

În cadrul Academiei Ruse de Științe (ICSAN) au avut loc de asemenea cercetări cu privire la clasificarea redundanțelor [93]. Aceasta urma să fie baza pentru dezvoltări ulterioare a sistemelor de calcul tolerante la defecte.

Clasificarea prezentată în [93] consideră că tipurile de redundanță sunt valabile pentru orice obiect. A fost introdus conceptul de redundanță minimală în spațiu, timp și energie. Se spune că obiectul poate fi „natural” (cu redundanță minimală) sau „artificial” redundant. Cu alte cuvinte, redundanța unui sistem poate fi proiectată și astfel nu e minimală. Redundanța artificială implantată într-un sistem asigură calea spre toleranță la defecte. Figura 2.1.1 ilustrează diferite variante obținabile, respectiv relația între redundanța spațială (S), de timp (T), informațională (I), materială (M) și energetică (E).

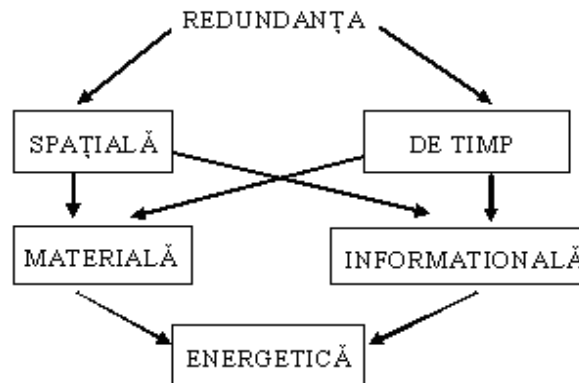


Fig. 2.1.1 Clasificarea ICSAN

Analizând această clasificare rezultă că: redundanța spațială sau de timp pot să fie implicit redundanța materială sau informațională, iar dacă redundanța materială sau informațională pot să fie implicit redundanță energetică (semnificația noțiunii are în vedere consumul suplimentar de energie care survine în urma implementării redundante).

O altă trăsătură a acestei clasificări este faptul că ea permite să se distingă caracteristici principale și caracteristici secundare ale tipurilor de redundanță. În cele ce urmează se prezintă diferite funcții redundante care pot fi valabile pentru sisteme tolerante la defecțiuni. În reprezentarea acestora caracteristica principală apare înaintea parantezei, iar caracteristicile secundare sunt enumerate între cele două paranteze.

- S(T,E,I) – detecția defectelor și corecție (realizată sub forma unei redundanțe spațiale, informaționale, care prezintă și elemente de redundanță de timp, energie suplimentară consumată),
- T() – reducerea specială a frecvenței în vederea reducerii ratei de defectare (redundanță de timp),
- T(E) – repetiția unor segmente de program pentru a detecta unele disfuncționalități (redundanță de timp, energie suplimentară consumată),

- $I(S,T)$  – realizarea unor copii informaționale ale unor date fără energie suplimentară,
- $I(S,T,E)$  – programe de testare pentru a detecta defectele permanente.

În general această clasificare prezintă numeroase neajunsuri dintre care, din perspectiva sistemelor automate, sunt importante următoarele:

- Clasa obiectelor analizate este prea mare și din cauză că nu este specifică este foarte greu de imaginat modul de realizare a tipului de redundanță respectiv.
- Lipsesc definiții riguroase ale tipurilor de redundanță și nu se face o categorisire clară a acestora.
- Relația dintre categoriile de redundanță, adică între spațiu, materie, timp, informație și energie, este vagă. Nu se poate proiecta un sistem redundant neexistând o bază clară.
- Sistemele automate și mai ales sistemele cu roboți nu pot fi descrise în termeni de putere și energie.

O clasificare mai reușită cu privire la sistemele de calcul tolerante la defecte este făcută de Zalewski [92]. El consideră toleranța la defecte drept capabilitatea de trecere a unei entități în cazul apariției unui defect din stare de funcționare cu succes într-o stare admisibilă de funcționare (apariția defectului, detectarea defectului și continuarea funcționării prin reconfigurarea sistemului).

Clasificarea [92] este ilustrată în figura 2.1.2:

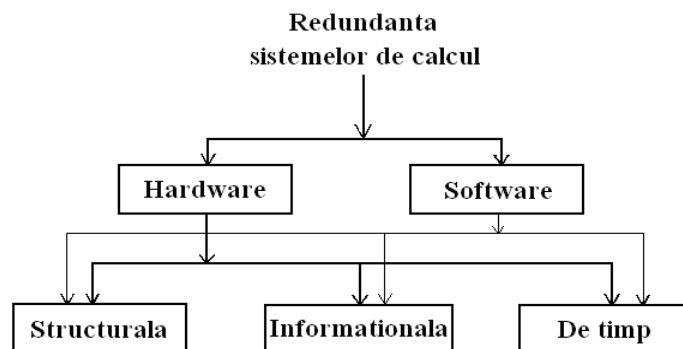


Fig. 2.1.2 Clasificarea redundanțelor după Zalewski

Principalele carențe ale acestei clasificări sunt următoarele:

- Clasificarea nu este completă. Clasificarea consideră redundanța informațională ca o parte componentă a redundanței hardware și software. Ori în cazul sistemelor automate redundanța informațională reprezintă un tip separat de redundanță care se aplică în cazul transmisiilor de date.
- Prin prisma faptului că se consideră doar redundanța de tipul hardware și software, clasificarea elimină alte tipuri esențiale.
- Redundanța de timp nu se poate reprezenta datorită naturii ei pe același nivel cu redundanța informațională. Redundanța de timp reprezintă un tip de redundanță informațională. De exemplu, în momentul în care se detectează o transmisie eronată de date, redundanța de timp oferă principala modalitate de corecție a erorii apărute și anume prin retransmisia datelor.

- Clasificarea nu detaliază în suficientă măsură tipurile de redundanță (modalitatea de realizare și implementare, încadrarea în diferite categorii în funcție de clasificări după alte criterii, specificarea modului de funcționare și intervenție, etc.), fapt ce pune sub semnul întrebării întreaga posibilitate de utilizare a celor descrise.

Studii cum sunt [41], [99], [11], [42], [57] prezintă într-o manieră incompletă diferite tipuri de redundanță, unele realizând clasificări doar pe diferite fragmente. Alte lucrări cum sunt [115], [83], prezintă clasificări de redundanțe parțial viabile dar incomplete. Ele au și neajunsul de a nu face o detaliere suficientă a tipurilor de redundanțe pentru ca ulterior proiectantul unui sistem să fie suficient documentat.

După cum rezultă din cele prezentate, nu există o clasificare de redundanțe universal valabilă iar acelea care există nu sunt nici pe departe complete. În acest context, în cadrul lucrării de față se dorește realizarea unei clasificări concise, cât mai cuprinzătoare, a tipurilor de redundanță, cu precădere în domeniul de aplicabilitate al sistemelor automate.

### 2.1.2 Clasificarea redundanțelor în funcție de nivelul la care se aplică

Un prim criteriu de clasificare a redundanțelor este după nivelul la care se aplică (componentă, modul, sistem). Întrucât această clasificare nu prezintă un interes deosebit prezentarea se limitează la o trecere în revistă a conceptului de bază.

În figura 2.1.3a se prezintă o structură serie (neredundantă) formată din patru componente. Figurile b, c și d ilustrează diferite tipuri de structuri redundante asociate structurii din figura a, respectiv, aplicarea structurii redundante la nivel de sistem, de modul (grupuri de câte două componente) și pentru fiecare componentă în parte.

Se constată [11] că redundanța la nivel de componentă (fig. 2.1.3d) aplicată corespunzător implică cel mai mare grad de fiabilitate (acest lucru ar putea însemna ca fiecare componentă să posede redundanță). Trebuie însă luat în considerare faptul că nu se va putea niciodată ca fiecare componentă să fie prevăzută cu o structură redundanță. În consecință, redundanța se va implementa în general la nivel de modul, dar concluzia acestui studiu este că nivelul de aplicare ar trebui să fie cel mai de jos posibil.

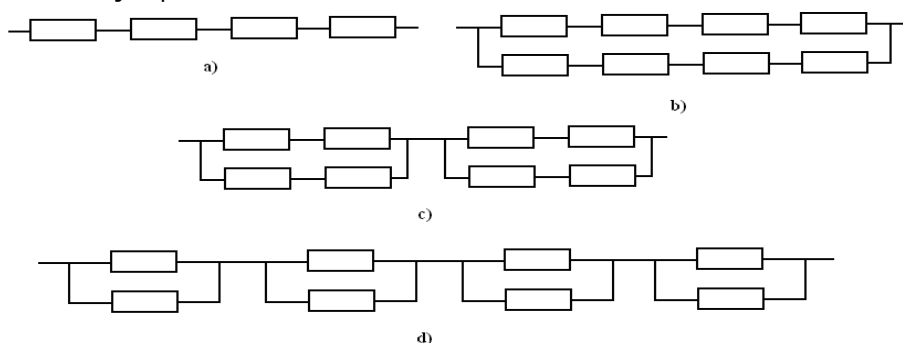


Fig. 2.1.3 Structuri prevăzute cu redundanță la diferite nivele

### 2.1.3 Clasificarea redundanțelor în funcție de reacția sistemului la defect

Un alt criteriu important de clasificare a structurilor redundante îl reprezintă reacția sistemului la defect. Această clasificare este ilustrată în figura 2.1.4.

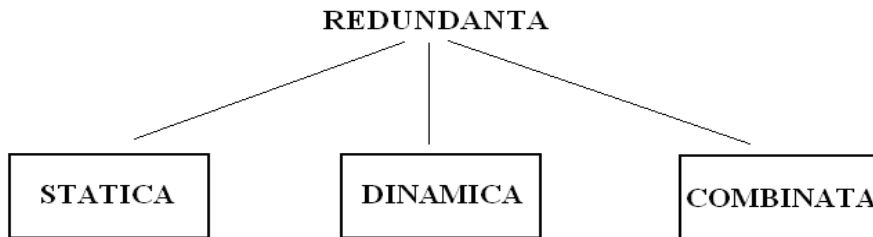


Fig. 2.1.4 Clasificarea redundanțelor în funcție de reacția sistemului la defect

Redundanța dinamică se definește printr-o reacție completă a sistemului la defect. În cazul redundanței dinamice, reacția sistemului la defect cuprinde detecția, diagnoza, izolarea defectului, repararea sistemului (de obicei prin reconfigurare), fiind posibilă repararea sistemului online, în timpul funcționării. Astfel, sistemul se aduce în stare de 100% funcționare.

Structura prevăzută cu redundanță de tip static are o reacție mai redusă din partea sistemului. Ea are caracteristicile unei structuri redundante, dar nu include reparația online a sistemului, respectiv izolarea defectului.

Bineînțeles redundanța de tip dinamic implică o structură mai complicată și mai costisitoare dar mentenanța pentru astfel de sisteme este mai ușor de efectuat și mai puțin costisitoare.

În cazul în care un sistem este prevăzut atât cu redundanță de tip static cât și cu redundanță dinamică se spune că redundanța sistemului este combinată.

În cele ce urmează pentru fiecare tip de redundanță prezentat se va preciza dacă se încadrează în grupa redundanțelor dinamice sau statice.

### 2.1.4 Clasificarea redundanțelor în funcție de zonă și de modalitatea de realizare

#### 2.1.4.1 Prezentarea sintetică a clasificării

Criteriile după care se realizează clasificarea structurilor redundante în cadrul acestui paragraf sunt:

- i) în funcție de sursa defectului (nivelul 1 al clasificării),
- ii) în funcție de modalitatea de realizare a redundanței (nivelul 2 al clasificării).

În figura 2.1.5 se prezintă primul nivel al acestei clasificări. Se disting patru mari categorii și anume: redundanțe pentru defecțiunile care apar în structura hardware, în structura software a sistemului, în faza de transmisie și recepție de date (transmisie, tele-operare, tele-mentenanță, etc.), respectiv redundanța care contracarează defecte apărute la nivelul utilizării și producerii sistemului de către elementul uman.

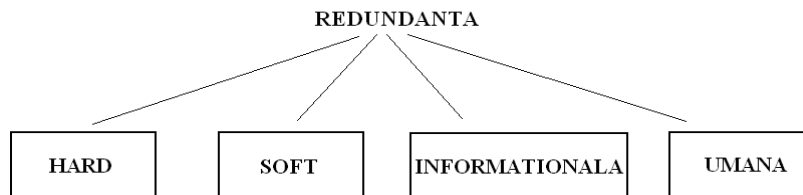


Fig. 2.1.5 Clasificarea redundanțelor în funcție de sursa defectului

Figura 2.1.6, se prezintă ambele nivele ale al clasificării pentru o mai bună localizare a clasificării redundanțelor în funcție de modalitatea de realizare (nivelul 2 al clasificării).

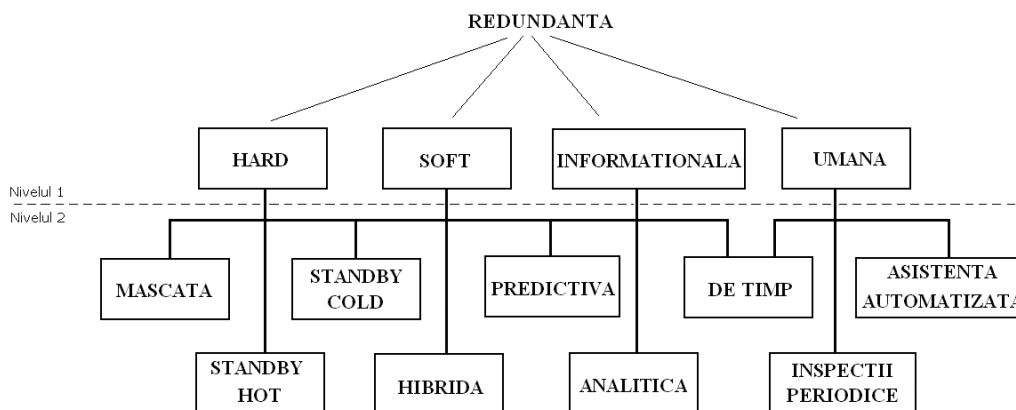


Fig. 2.1.6 Clasificarea redundanțelor

Bineînțeles, alegerea tipului de redundanță se face în funcție de necesitățile fiecărei aplicații. Sistemele automate prezintă un grad mare de complexitate și în consecință oricare ramură a clasificării din fig. 2.1.6 este aplicabilă. În prima fază se distinge zona unde este necesară o structură redundanță pentru mărirea fiabilității și apoi se optează pentru un tip sau altul.

În continuare se prezintă mai în profunzime elementele clasificării, iar pentru fiecare avantajele și dezavantajele caracteristice utilizării lor. Există numeroase motive care impun folosirea unui anumit tip de redundanță. Totul depinde de performanțele așteptate de la sistem, de specificul aplicației, strâns corelate cu costul suplimentar pe care-l implică folosirea structurii respective.

#### 2.1.4.2 Aspecte privind tipurile de redundanță

Înainte de a începe prezentarea tipurilor de redundanță se reamintește că în cadrul subcapitolului 1.3 s-a definit defectul și faptul că funcționarea sigură a unui sistem înseamnă asigurarea că acesta se încadrează în limitele de specificație și atinge performanțele dorite.

În cadrul subcapitolului 1.2 s-au enumerat metode de evaluare ale rețelelor de fiabilitate raportate la structurile redundante. Raportarea la structurile prevăzute cu redundanță este esențială deoarece trebuie făcută neapărat diferența între tip de structură și tip de structură redundanță. Această diferență se desprinde din următorul exemplu.

Structura paralel (fig. 2.1.7) reprezintă punctul de plecare pentru structura redundantă dar ea nu reprezintă automat redundanță.

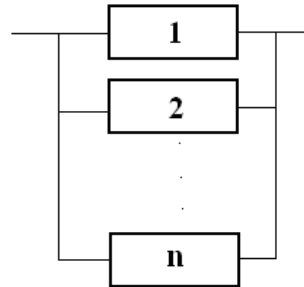


Fig. 2.1.7 Structura paralel cu n entități

Specificul structurii paralele este că în orice moment toate componentele sunt active (fie ele componente hard, programe soft sau transmisii de date). Modalitățile de comportare ale structurii paralele sunt determinate de scopul utilizării acestora. Structura paralel va deveni o structură redundantă în momentul în care există posibilitatea preluării sarcinilor furnizate de entitatea defectă de către o altă entitate. Din acest motiv elementele structurii paralele trebuie să realizeze aceeași funcție a sistemului și să se poată realiza acțiunile necesare detectării și înlocuirii elementului defect.

În principiu, o structură paralel nu prezintă implicit aceste caracteristici: componentele structurii paralele nu trebuie să preia neapărat aceeași funcție în sistem, ci să servească pentru detecția defectelor și nu pentru înlocuirea elementului defect. În consecință, trebuie avută în vedere distingerea celor două moduri de percepere a structurii paralele. În continuare studiul se axează pe structurile redundante și din acest motiv interesează tipologia unei structuri redundante dacă se cunoaște care element s-a defectat, precum și dacă structura poate să identifice atât defectele temporare, cât și cele permanente.

După cum s-a precizat, structura paralel este exemplul tipic pentru care elementele componente sunt în permanență active. Structurile cu redundanță activă au în permanență componentele în stare de funcționare, furnizând astfel un neajuns al acestor structuri. Neajunsul constă în consumul unei cantități de putere în plus prin funcționarea în permanență a tuturor componentelor. Tipul de redundanță activă nu se poate folosi în cazul în care un element menținut în stare de funcționare poate provoca un defect periclitant. Pe de altă parte, redundanța pasivă presupune elemente de rezervă care intră în funcționare doar în cazul în care apare un defect.

Tot relativ la tipurile de redundanță, se introduce noțiunea de redundanță omogenă. Prin redundanța omogenă se înțelege structura redundantă ale cărei componente sunt identice. Dacă aceste componente nu sunt identice (realizate după principii fizice diferite) se vorbește de o redundanță bazată pe diversitate.

### 2.1.4.3 Redundanța mascată

Redundanța mascată este denumită în mod frecvent redundanța paralel cu vot majoritar. În acest caz, sistemul redundant conține  $n$  componente, dar pentru realizarea funcției sistemului sunt necesare doar  $m$  componente funcționale. Astfel, dacă apar defecte la restul de  $n-m$  componente acestea vor fi mascate, iar sistemul va funcționa cu un număr de  $m$  componente (fig. 2.1.8).



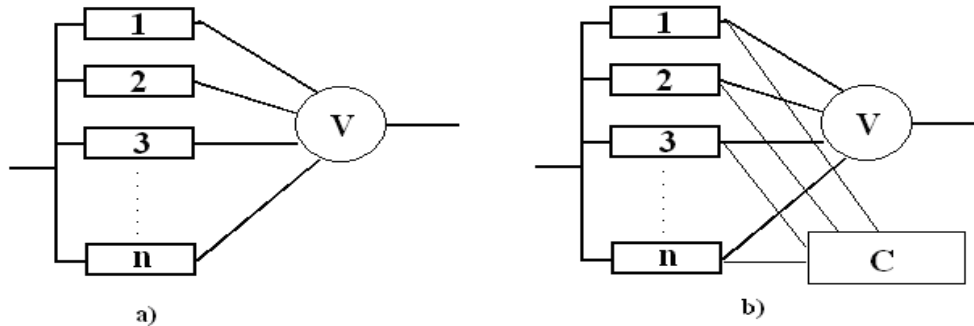


Fig. 2.1.8 Redundanță mascată

În figurile de mai sus elementul de votare V semnaleză dacă există  $m$  componente din cele  $n$  care funcționează corect. V primește semnale de la cele  $n$  elemente, apoi numără câte semnale dintre primele  $m$  sunt corespunzătoare, iar dacă numărul este mai mic decât  $m$  analizează și semnalul  $m+1$ ,  $m+2$  ș.a.m.d., până la  $n$ . Dacă numărul de semnale acceptate este cel puțin egal cu  $m$  atunci sistemul se consideră funcțional, iar dacă acest număr este mai mic decât  $m$  atunci sistemul este defect.

Acest tip de redundanță se încadrează în grupul redundanțelor statice. Nu există posibilitatea înlocuirii online a elementelor defecte, mai mult este posibil să nici nu se evidențieze elementul defect (fig. 2.1.8a). Pentru a „demasca” elementul defect se introduce un comparator  $C$ <sup>2)</sup> (fig. 2.1.8b).

Pentru a crește fiabilitatea elementului de votare V se predeve adeseori și acesta cu o structură redundanță ca în figura 2.1.9 (sistem cu trei elemente și element de votare redundanță).

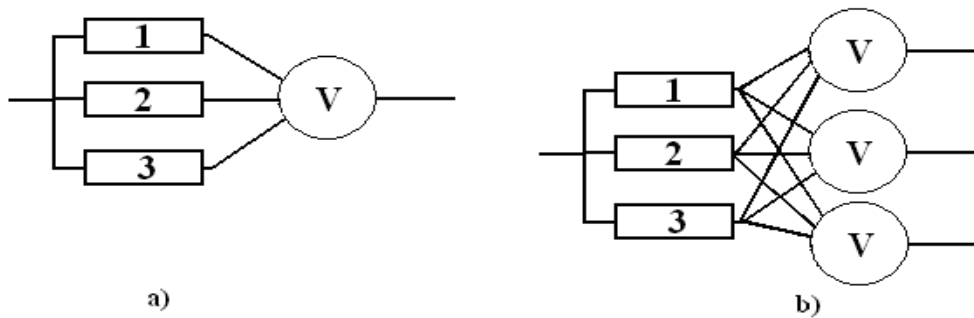


Fig. 2.1.9 Redundanță mascată de ordin trei

#### 2.1.4.4 Redundanța standby hot și standby cold

Pentru început se specifică semnificația simbolizărilor folosite în cadrul figurilor din acest paragraf:

- linia continuă semnifică interacțiunea funcțională dintre două componente

<sup>2)</sup> Compară ieșiri de la entități defecte

- linia punctată semnifică interacțiunea de comandă dintre două componente

**Redundanța standby hot** este ilustrată în figura 2.1.10.

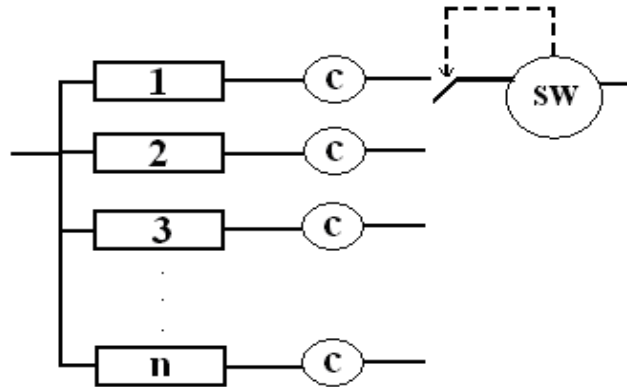


Fig. 2.1.10 Redundanța standby hot

Ea reprezintă o structură paralelă cu un comparator<sup>3)</sup> pe fiecare ramură și un comutator. În cazul în care în urma comparației valoarea de ieșire diferă de valoarea limită prescrisă setată în comparator, deci apare un defect, realizarea funcției se comută pe ramura următoare care este de asemenea activă.

Pentru structura redundanță standby hot este valabilă formula (10) din subcapitolul 1.2. Redundanța standby hot are avantajul de a identifica în permanență elementul defect.

De regulă, apartenența redundanței standby hot la grupul de redundanțe de tip dinamic este conferită de posibilitatea de reparare a elementului defect după ce s-a realizat comutarea (de exemplu posibilitatea deconectării elementului defect în vederea înlocuirii). Dacă elementele 1, 2, ..., n nu sunt reparabile sau dacă acțiunea de reparare nu se poate realiza atunci redundanța se încadrează în grupul de redundanțe de tip static.

**Redundanța standby cold** este ilustrată în figura 2.1.11:

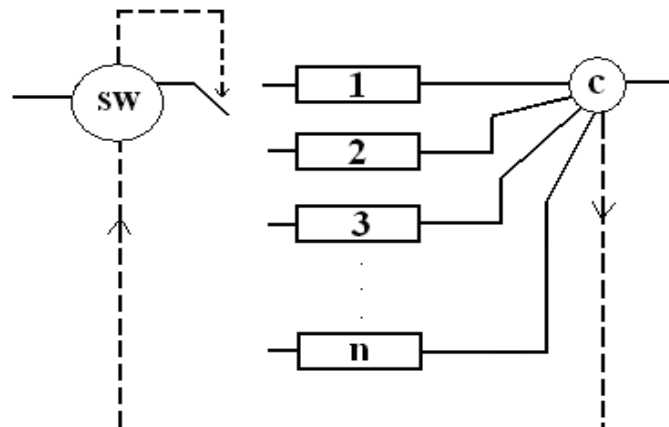


Fig. 2.1.11 Redundanța standby cold

<sup>3)</sup> Compară ieșirea unei entități cu limite prestabilite

În acest caz, redundanța are avantajul că la un moment dat este activ un singur element al schemei. Un alt avantaj din punctul de vedere al costului îl constituie faptul că se lucrează cu un singur comparator <sup>4)</sup>. Dar acest lucru este și un dezavantaj deoarece dacă acest comparator se defectează, rezultă implicit defectarea întregii structuri. Alt dezavantaj al redundanței standby cold este faptul că apar mai multe legături (conexiuni) decât în cazul redundanței active și mascate, apar de asemenea comutări, iar din acest punct de vedere siguranța are de suferit (conexiunile în cadrul unui sistem reprezintă puncte cu o fiabilitate scăzută).

Caracterizarea din punctul de vedere al apartenenței la tipurile de structuri redundante dinamice sau statice este oarecum similară cu cea de la structura redundanță standby hot, în următorul sens: dacă există posibilitatea de reparare a elementului defect după ce s-a realizat comutarea (de exemplu posibilitatea deconectării elementului defect în vederea înlocuirii) atunci redundanța va avea un caracter dinamic, iar dacă această operație nu este posibilă atunci redundanța are un caracter static.

**Redundanța standby propriu-zisă** se încadrează în rândul redundanțelor hibride dar se descrie în cadrul acestui paragraf. Structura este ilustrată în figura 2.1.12. Redundanța standby propriu-zisă facilitează izolarea completă a elementului defect.

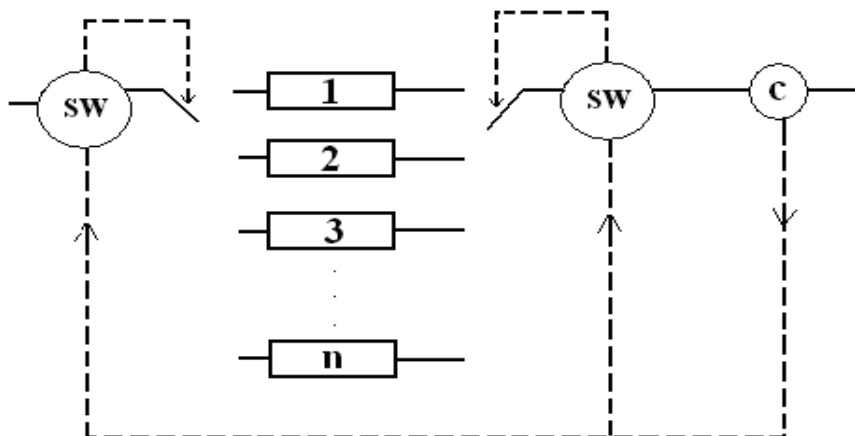


Fig. 2.1.12 Redundanța standby hibridă

Caracterul dinamic este cu atât mai pronunțat cu cât, în caz de defecțiune, se poate recurge la mentenanță on-condition și se poate înlocui elementul defect. În cazul redundanței standby propriu-zisă sistemul prezintă redundanță în orice moment datorită posibilității de intervenție în timpul funcționării.

#### 2.1.4.5 Redundanța predictivă

Există situații în care defecțiunea poate conduce la periclitatea întregului sistem și al mediului înconjurător (defecțiuni la limita de tolerare). De aceea, de multe ori este foarte important ca defectul să nu se producă. În acest context, posibilitatea de predicție a defectelor este una din cele mai importante caracteristici de protecție a unui sistem.

<sup>4</sup> Are aceeași funcție ca și în cazul redundanței standby hot

Redundanța predictivă constă tocmai în posibilitatea unui sistem de a se proteja împotriva defectelor înainte ca ele să se producă [68], [80]. Acest lucru se realizează prin diferite metode de prognoză.

Două metode utilizabile constau în:

- folosirea filtrelor sau observatoarelor, în particular a filtrelor Kalman
- folosirea statisticilor pe bază de șiruri de valori memorate

Filtrele și observatoarele reprezintă structuri sistemice dinamice, care permit sesizarea tendinței de evoluție a sistemului prin intermediul mărimilor de stare ale acestuia, respectiv predicția stării funcționale următoare a sistemului [122]. Ele pot fi implementate hardware sau software și oferă o modalitate de a detecta posibilitatea de a se produce defectul înainte cu perioada de timp  $T_p$  (perioada de timp corespunzătoare fiecărei pași de incrementare- pași de predicție). Durata  $T_p$  trebuie să fie suficient de lungă pentru ca să se poată activa serviciul de mentenanță.

Predicția se poate realiza și cu memorarea de șiruri de valori ale mărimilor caracteristice ale stării sistemului. Bineînțeles că dimensiunile șirurilor sunt setate dinainte (astfel încât folosind principiul „uitării” celor mai vechi valori, dimensiunile să fie suficiente pentru a putea urmări evoluția valorilor variabilelor caracteristice și de a putea prezice stările ulterioare).

De obicei, defecțiunile (defecte propriu-zise precum și ieșirea din plaja de funcționare corectă) se produc treptat ca urmare a uzurii și astfel elementele în curs de defectare vor furniza semnale care se apropie de valorile limită memorate. Dacă se memorează în timp discret și periodic aceste valori se poate determina numărul de pași până la defectare.

Procedura este următoarea: folosind valori absolute sau diferențe se analizează și se observă dacă valorile se îndreaptă spre valoarea limită prescrisă (memorată) și considerând gradientul constant sau estimat după un alt algoritm, se determină prin calcul peste câte perioade de timp se va produce defectul. În urma calculelor va rezulta că sistemul se va defecta peste un număr de  $k$  pași. Dacă acest număr  $k$  este mai mare decât numărul permis memorat de pași până la defectare  $x$  atunci se continuă execuția. Dacă  $k$  este mai mic sau egal decât numărul permis de pași  $x$ , atunci sistemul va considera că se află într-o stare critică și va comuta realizarea funcției pe o altă ramură în cazul în care structura este prevăzută cu redundanță sau se va solicita serviciul de mentenanță.

Pentru realizarea predicției se poate destina un procesor de semnal conectat la ieșirea elementului analizat. Procesorul de semnal se va comporta ca un regulator de diagnoză (bazat pe tabele) care va avea stocate în memorie valorile limită, numărul de pași  $x$  permis până la defectare, precum și șirul memorat al valorilor curente.

În momentul în care zona de memorie destinată memorării se umple, se procedează la „uitarea” unor valori, prin translatarea valorilor (în fig. 2.1.13 la stânga) astfel încât primul element (cel mai devreme introdus) se pierde.

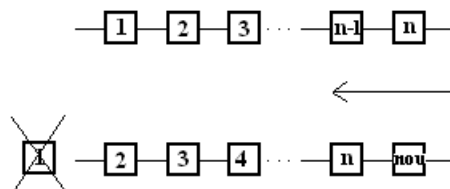


Fig. 2.1.13 Evoluția șirului memorat

Pentru realizarea operației de predicție, șirul valorilor memorate poate fi utilizat în diverse moduri. Astfel, în continuare se prezintă ca exemplu implementarea secvenței care conduce la decizie pe baza unei metode de predicție cu ajutorul unui șir bazat doar pe memorarea ultimelor două valori.

În pseudocod apar următoarele notații:

a – șirul valorilor memorate

a[i] – valoarea din șirul valorilor memorate aflat pe poziția i (valoare curentă sau ultima poziție)

Dif – variația valorii furnizate la ieșirea elementului supervizat față de pasul anterior

k – numărul de pași până la defectare (valoarea estimată)

x – numărul minim de pași admiși până la defectare (valoare prescrisă, setată)

limită – valoarea limită prescrisă ce mărginește plaja admisibilă de evoluție a ieșirii componentei

Interv – intervalul dintre valoarea limită și valoarea curentă

Compară a[i] cu a[i-1]

Dif = abs (a[i] - a[i-1]) ; calculează cu cât s-a apropiat elementul de limita de defectare

Interv<sub>1</sub> = abs (limită<sub>1</sub> - a[i]) ; intervalul rămas până la valorile limită

Interv<sub>2</sub> = abs (limită<sub>2</sub> - a[i])

Interv = min (Interv<sub>1</sub>, Interv<sub>2</sub>) ; determină cea mai apropiată valoare limită

k = Interv / Dif ; calculează numărul de pași până la defectare

Dacă k < x atunci stare critică ; compară k cu numărul x memorat de pași limită

După determinarea apariției unui defect în urma predicției se trece la preluarea funcției realizate de elementul defect de către un alt element. În funcție de posibilitatea de realizare a reparării sau izolării elementului defect după reconfigurare se poate spune dacă structura redundantă este sau nu dinamică.

#### 2.1.4.6 Redundanța de timp

Un tip aparte de redundanță este redundanța de timp. Acest fel de redundanță nu intră în tiparele primului nivel al clasificării din figura 2.1.6.

Lucrări precum [99] se referă la redundanța de timp dar într-un mod incomplet și fără a o clasifica corespunzător. În opinia autorilor, redundanța de timp apare ca o ramură separată pe lângă redundanța hard, soft sau informațională. Redundanța de timp nu se poate clasifica astfel întrucât ea poate fi o componentă a fiecărui tip de redundanță din cadrul clasificării realizate în figura 2.1.5: hard, soft, informațională, umană.

În esență, prin redundanța de timp se înțelege: repetiția acțiunii în vederea identificării defecțiunii și corectării acesteia și se poate aplica în diverse situații, indiferent de zona în care apare defectul și de tipul acestuia. Atât defecte de tipul hard, soft, informaționale sau umane pot fi prevenite sau identificate cu ajutorul redundanței de timp.

Forma cea mai simplă de redundanță de timp este ilustrată în figura 2.1.14.

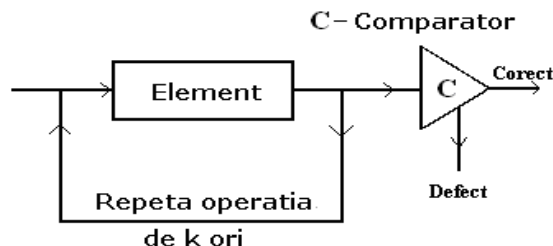


Fig. 2.1.14 Redundanță de timp simplă (repetarea necondiționată de sesizarea defectului)

Figura 2.1.14 ilustrează repetiția necondiționată de sesizarea defectului. Operația se execută de  $k$  ori, iar comparatorul ia o decizie cu privire la apariția sau nu a unei defecțiuni la nivelul elementului (se compară ultimele 2, respectiv  $k$  valori furnizate la ieșire). Bineînțeles printr-o comparație simplă se pot determina doar defecțiunile temporare, dar există posibilități de determinare și a celor permanente descrise mai jos.

Dată fiind definiția redundanței dinamice prezentată în paragraful 2.1.3, integrarea redundanței de timp în forma ei cea mai simplă în această categorie necesită o discuție:

- i) Dacă ne situăm în cazul transmisiei de date de la o singură sursă, iar în urma identificării defectului funcția se realizează printr-una din cele  $k$  informații analizate (mai precis se identifică un defect temporar și se remediază situația) atunci redundanța de timp prezintă un caracter dinamic.
- ii) Dacă se identifică un defect prin procedeul de mai sus, dar funcția se realizează printr-o altă componentă a structurii redundante (ex. informația este preluată de la o altă sursă) atunci caracterul dinamic sau static depinde de modalitatea de implementare a structurii redundante.

Folosirea redundanței de timp în această formă aduce cu sine avantaje și dezavantaje. Unul dintre avantaje ar fi acela că nu este necesară memorarea unor valori prescrise pentru a putea depista defectul. Pornind de la prezumpția că la început, elementul funcționează corect, această primă valoare va reprezenta etalonul de comparație. Acesta este un avantaj deoarece se economisește atât spațiul necesar memorării valorilor prescrise, cât și procedurile de acces și de comparare mai complexe. Un alt avantaj este ușurința cu care se realizează diagnoza, practic repetiția execuției. Sistemul este văzut ca un „black box” și se lucrează doar cu mărimile de intrare și ieșire.

Există însă și dezavantaje care pun semnul de întrebare la folosirea sau nu a redundanței de timp. Unul dintre ele, esențial, este timpul de execuție. Practic, redundanța de timp implică multiplicarea timpului de execuție (dublarea, în cazul unei singure repetiții).

Un alt dezavantaj îl reprezintă repetiția obligatorie a operației. Acesta condiționează amplasarea redundanței de timp. Cauza o reprezintă riscul mare la care este supus sistemul în cazul în care componenta care este prevăzută cu acest tip de redundanță poate produce un defect periclitant. Este posibil că repetând execuția în cazul în care componenta cade într-un defect periclitant, se periclitează siguranța sistemului, precum și a mediului înconjurător.

În ceea ce privește comparatorul și reținerea valorii de dinaintea repetiției, practic se poate folosi un filtru de întârziere (delay) iar ieșirea acestuia se compară cu valoarea furnizată de repetiția execuției. Un exemplu de implementare a filtrului

de întârziere în limbajul de asamblare al unui procesor de semnal (TMS320T31-50) arată în felul următor [62]:

- filtrul de întârziere are funcția de transfer  $H(z) = z^{-1}$  (figura 2.1.15) și realizează operația  $y[t]=u[t-1]$  sau  $y = u_{\text{anterior}}$  (ieșirea este intrarea anterioară).

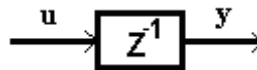


Fig. 2.1.15 Filtrul de întârziere

Codul care realizează implementarea filtrului este:

```

ldf  @u_curent,R7
ldf  R7,R6      ; se stochează intrarea în registrul R6
ldf  @u_anterior,R7 ; în registrul R7 se reține intrarea anterioară
stf  R6,@u_anterior ; intrarea curentă va reprezintă intrarea anterioară
stf  R7,@y_anterior ; ieșirea curentă va reprezenta intrarea anterioară

```

Dacă se efectuează doar o singură repetiție se vor observa doar defecte despre care nu se cunoaște nimic din punctul de vedere al caracterului permanent sau temporar. Există diferite metode prin care se poate determina dacă defectele sunt permanente. Una dintre acestea constă în executarea unui număr suficient de repetiții suplimentare pentru ca să se poată urmări evoluția ieșirii.

Figura 2.1.16 ilustrează o schemă de principiu care se bazează pe repetiția execuției până când valoarea furnizată este corespunzătoare. Principiul este repetiția operației condiționată de sesizarea defectului. Pentru a nu intra într-un ciclu infinit se fixează un număr de maxim  $k$  repetiții. Astfel se realizează o redundanță în timp care poate determina și defectele permanente. Dezavantajul acestei structuri este acela că trebuie memorate valori prescrise de comparație.

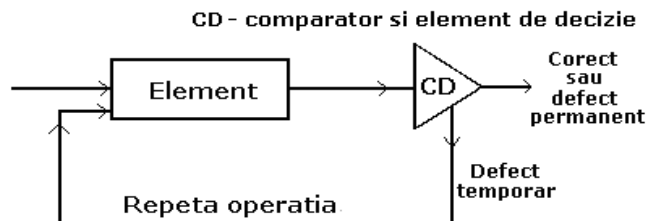


Fig. 2.1.16 Exemplu de redundanță de timp (repetiție condiționată de sesizarea defectului)

Acest tip de schemă poate reprezenta de asemenea, o ramură din cadrul tipului de redundanță umană. Repetiția execuției acțiunii umane până când aceasta se încadrează în limitele specificate reprezintă o soluție curentă de protecție împotriva defectelor ce apar ca urmare a acțiunii omului.

Relativ la încadrarea redundanței de timp în cadrul redundanțelor cu caracter dinamic, trebuie ca după detecția defectului permanent (în urma operațiilor efectuate în figurile 2.1.14 sau 2.1.16), funcția sistemului să fie preluată de către o altă componentă, iar în urma reconfigurării să se poată efectua reparația online a elementului defect (ex. scheme mai complexe care se bazează pe o comutare de tip standby propriu-zisă).

Redundanța de timp se poate combina cu un alt tip de redundanță rezultând astfel o redundanță hibridă (paragraful 2.1.4.7).

O altă aplicație tipică pentru redundanța de timp o reprezintă execuția în paralel a două procese (fie ele soft, hard, informaționale) dar cu o întârziere de o unitate de timp. În acest caz, unul dintre procese reprezintă etalonul de comparație. Întregul ansamblu este prevăzut cu redundanță activă, vezi paragraful 2.1.4.7.

### 2.1.4.7 Redundanța hibridă

Cel mai mare grad de complexitate îl prezintă redundanța hibridă. Aceasta înglobează mai multe tipuri de redundanță în vederea obținerii unei disponibilități ridicate și a unei funcționări sigure. Nu se poate atribui dinainte caracterul dinamic sau static structurii deoarece el depinde de combinația din care este compusă structura redundanță hibridă respectivă.

Abordarea redundanței hibride în acest paragraf se va face urmărind câteva exemple.

Primul exemplu este ilustrat în figura 2.1.17. Este vorba despre o structură complexă compusă din:

- o redundanță standby propriu-zisă cu posibilitate de izolare a elementului defect. În fiecare pas de discretizare a timpului ieșirea în valoarea absolută este comparată cu valoarea absolută a valorii limită memorate. În cazul în care ieșirea depășește valoarea limită, se dă comandă la comutatoarele care cuplează componenta următoare din structura standby. În acest fel se pot identifica defectele bruște (vezi subcapitolul 1.3 și paragraful 4.3.2).
- o redundanță predictivă care se bazează pe memorarea unui șir de valori consecutive ale mărimii  $y$ . Apoi se face predicția numărului de pași  $k$  până la defectare.

Redundanța din figură este o redundanță de tip dinamic.

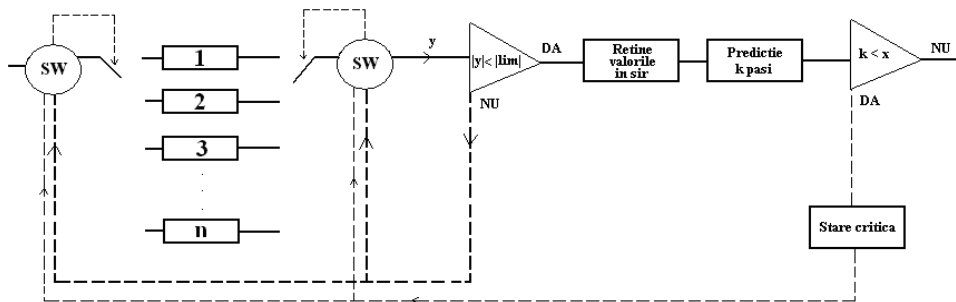


Fig. 2.1.17 Redundanță hibridă, exemplul 1

Ca un al doilea exemplu, încadrabil în grupul de redundanțe soft, se consideră o redundanță hibridă compusă din:

- o redundanță standby cold ale cărei elemente sunt două programe soft a căror realizare a ținut cont de principiul diversității echipamentului sau au fost implementate prin algoritmi diferiți (astfel se elimină defectele neobservate în testare sau proiectare); ambele programe sunt active în orice moment.
- o redundanță de timp care se bazează pe faptul că cele două programe rulează decalat cu o unitate de timp; astfel, în momentul în care programul 1 din fig. 2.1.18 execută segmentul de program  $n$ , programul 2 execută segmentul de program  $n-1$ .



În momentul în care segmentul  $n$  din programul 1 se defectează se comută pe segmentul 2 unde se repetă practic execuția părții  $n$ .

Atribuirea caracterului dinamic sau static al structurii redundante necesită o discuție. Redundanța este în principiu de tip dinamic deoarece sistemul se poate reface online, în procent maxim, întrucât defecțiunile software se pot remedia în timpul funcționării sistemului. Însă, structura redundantă din acest exemplu poate să aibă un caracter static dacă resursele hardware și software nu permit reîncărcarea și repunerea în funcțiune online a programului software reparat.

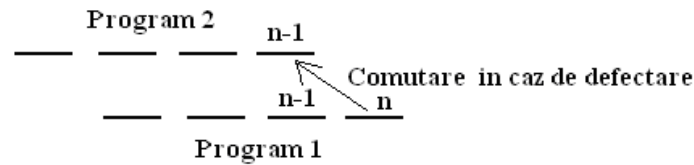


Fig. 2.1.18 Redundanță hibridă, exemplul 2

Al treilea exemplu este ilustrat în figura 2.1.19. El se referă la o redundanță hibridă din clasa redundanțelor informaționale.

Structurile redundante care intră în componența sistemului redundant sunt:

- redundanța de timp prin care în caz de eroare se repetă transmisia de date;
- redundanța standby hibridă care în caz de defectare a unui canal de transmisie comută pe canalul 2; un singur canal este funcțional la un moment dat și se permite reparația în timpul execuției.

Testarea validității informației transmise în cazul structurii curente este realizată într-o modalitate tipică redundanței informaționale. Nu există o informație prememorată cu care să se compare data transmisă (dacă ar exista nu ar avea sens transmisia). De obicei, se folosesc diferite metode de testare a validității datelor transmise [58], [99]. Una dintre metodele de testare a validității informației este verificarea parității biților care compun informația. Această metodă se bazează pe suplimentarea dimensiunii informației transmise cu un bit. Acest bit va conține paritatea datei. Astfel la recepție se verifică paritatea biților datei transmise și dacă aceasta nu coincide cu valoarea bitului suplimentar înseamnă că a apărut un defect.

Exemplu:

- informația transmisă - 1010111\_1\_ - ultimul este bitul de paritate
- la recepție dacă respectiva dată este 1010110\_1\_ - se observă că paritatea informației este 0 iar bitul de paritate este 1, astfel a apărut un defect.

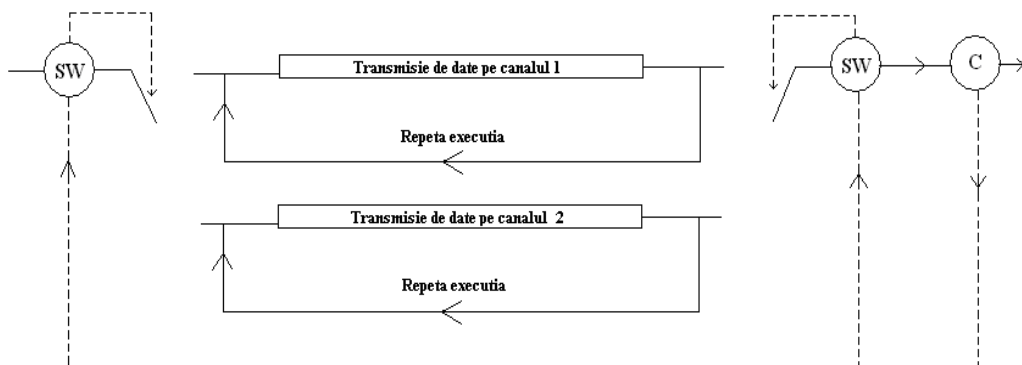


Fig. 2.1.19 Redundanță hibridă, exemplul 3

Exemplul 3 este un tip de redundanță dinamică deoarece redundanța standby propriu-zisă este o parte componentă a sistemului și astfel devine posibilă repararea online a entității defecte după ce aceasta a fost detectată și izolată.

#### **2.1.4.8 Redundanța analitică**

Redundanța analitică este tipul de redundanță care se bazează pe un model matematic al sistemului sau pe dependențe corelaționale între diverse stări ale sistemului. Structurile cu redundanță analitică sunt intens studiate și aplicate atât în sisteme automate, în general, [78], [100] cât și în robotică, în particular [74], [81].

Aplicarea redundanței analitice presupune o cunoaștere apriorică a comportării sistemului. În prima situație, modelul matematic al procesului furnizează împreună cu sistemul real o valoare reziduală din care se determină apariția unui defect. În a doua situație, dependențele corelaționale între diversele stări ale sistemului furnizează informații legate de posibila defectare a acestuia.

#### **2.1.4.9 Inspecții periodice**

Un tip de redundanță practicat la nivelul „element uman” este reprezentat de metoda bine cunoscută a inspecțiilor periodice. Aceasta constă în inspecția periodică efectuată de către personal bine calificat cu scopul de a corecta eventualele defecte apărute la nivelul elementului uman.

Inspețiile periodice surprind o structură redundantă care poate fi percepută și implementată în mai multe moduri, unde elementul uman reprezintă sursa defectului:

- în urma identificării defectului, elementul uman defect este înlocuit
- în urma identificării defectului elementul uman este reinstruit sau corectat

#### **2.1.4.10 Redundanță umană prin asistență automatizată**

La o primă vedere interacțiunea dintre elementul uman și sistemul tehnic în cazul procesului de supervizare și control, este concentrată în principal înspre detecția și repararea defectelor apărute în timpul funcționării. Acest fapt pare plauzibil din moment ce munca în aceste condiții implică o presiune suplimentară de timp asupra personalului care operează și riscul unor mari pierderi în cazul defectării sistemului sau în cazul erorii umane. Obiectivele cercetărilor [39] sunt îndreptate spre stardarde mai ridicate de fiabilitate și siguranță în funcționare.

Mentenanța, pe de altă parte, este privită cu destulă superficialitate, în special în cazul în care se folosește tipul de mentenanță „la defectare” (vezi paragraful 1.4.1). Mentenanța „la defectare” se adoptă în ipoteza că momentul în care un sistem se oprește, adică se află în starea de „shutdown”, nu implică o presiune stringentă de timp asupra elementului uman în vederea supervizării și a controlului sistemului. În realitate problema se pune alfel și anume, timpul înseamnă bani. Astfel timpul care îl petrece un sistem în starea „shutdown” reprezintă pierderi imense de unități monetare pe oră, deci personalul trebuie să acționeze în consecință.

O altă problemă apare atunci când elementul uman efectuează diagnoza sau reparațiile asupra sistemului tehnic, iar acestea, datorită modului în care se

efectuează, se soldează cu defecte ulterioare. De aceea, procedurile de diagnoză a sistemului în procesul de mentenanță, precum și instruirea personalului a fost subiect al numeroase investigații de-a lungul timpului.

În ceea ce privește metodele de dezvoltare a interfețelor om-mașină, trebuie remarcate metode de tip „multimedia”, „realitate virtuală” și „realitatea sporită de calculator” (RSC) care, după [39], par să prezinte în viitor posibilități evidente de progres în ceea ce privește diagnoza și mentenanța sistemelor tehnice. Dintre acestea, RSC este cea mai promițătoare. Aceste tehnici sunt relativ noi și sunt de interes în tele-operare, diagnoză și mentenanță și instruirea personalului. Datorită gradului mare de automatizare sau pur și simplu datorită mărimii și complexității sistemului, procesul de supervizare și control implică o operare indirectă a elementului uman. În plus, sistemele moderne sunt robotizate, adică execută în condiții normale acțiuni repetitive predefinite, iar de cele mai multe ori operează în medii bine cunoscute.

În cazul sistemelor robotizate simple care efectuează acțiuni dinainte planificate și bine studiate, sarcina operatorilor este doar de menținere a procesului într-o stare stabilă. De cele mai multe ori acest lucru se realizează prin observarea sistemului într-o manieră indirectă, cu ajutorul unor informații abstracte provenite de la proces prin instrumente specifice și vizualizarea acestora pe niste ecrane în încăperile de control. Rareori operatorul are nevoie de imagini reale despre sistem și în aceste cazuri pur și simplu se uită din încăperea de control sau imaginile despre proces îi sunt furnizate pe niște ecrane cu ajutorul camerelor de luat vederi. Astfel, pentru aceste tipuri de sisteme în condiții de operare normale, nu este mare nevoie de o reprezentare reală a comportării acestuia pe baza „realității virtuale”.

Operarea sistemelor tehnice este intens analizată și studiată, un exemplu specific fiind roboții care evoluează în medii cu hazard (vezi paragraful 4.2.1), executând task-uri care nu pot fi prezise și planificate (sau acțiunile lor sunt dinainte stabilite doar parțial, într-o mică măsură). Aceste tipuri de sisteme sunt mobile și trebuie controlate de la distanță. Modul de operare necesită impresia unei observări directe a procesului și de asemenea informații despre mediul înconjurător care se poate schimba. Astfel o vedere externă este foarte importantă, iar aceasta este de multe ori îngreunată de distanța dintre operator și proces. În consecință elementului uman trebuie să îi fie furnizate imagini hibride despre sistem formate din „vizualizări externe” a sistemului, fie ele reale sau bazate pe „realitatea virtuală”, sporite de informații adiționale, constituind un exemplu tipic de tehnică RSC.

În cazul mentenanței, contrar supervizării și controlului, de cele mai multe ori aparențele fizice a sistemului dau cele mai multe informații despre posibilele defecte. Dar pentru operator, imaginea fizică externă contează doar atunci când defectele nu sunt ascunse sub o suprafață intactă. Acest caz reprezintă din nou o problemă tipică RSC (fig. 2.1.20), elementul uman având nevoie de imagini reale din interior (ex. vizualizare cu raze X), sporite de informații legate de procedurile pe care trebuie să le efectueze, deci de imagini hibride ca și cele din cazul teleoperării.

Se pot distinge două tipuri de instruire a personalului:

- instruire pentru supervizarea și controlul sistemului
- instruire pentru mentenanță

În primul caz comportarea reală a procesului este simulată la fel ca și în cazul real.

În al doilea caz faza în care sistemul este în reparare este importantă. Astfel personalul trebuie instruit pe sistemul real cu ajutorul tehnicilor RSC. Dacă sistemul este prea mare sau este încă în construcție, instruirea se va face cu ajutorul unor machete sau a simulărilor cu ajutorul tehnicilor „realității virtuale”.

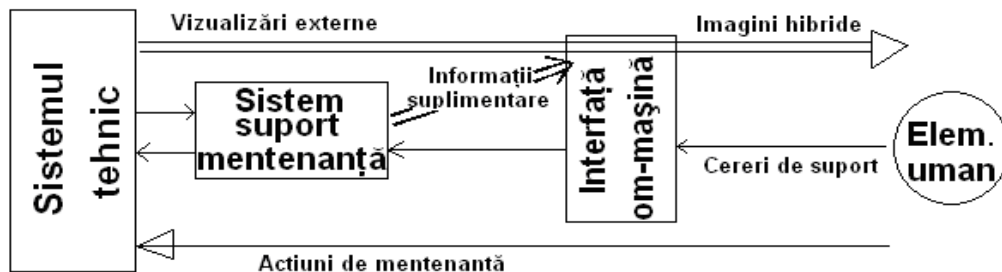


Fig. 2.1.20 Mentenanța cu ajutorul tehnicii RSC

De asemenea un element foarte important în mentenanță pentru personal îl reprezintă instrumentul sau ajutorul suplimentar de căutare în documentația sistemului tehnic. Având de-a face cu o documentație voluminoasă, operatorul nu mai este în stare sub presiunea timpului să găsească informațiile dorite într-un interval scurt. De aceea în primul rând se recomandă folosirea elementelor „multimedia” la realizarea documentației (conexiuni între text și imagini) și un sistem de navigare rapidă și transparentă.

Opinia generală cu privire la argumentarea necesității tehnicilor RSC pentru instruire și mentenanță este cea prezentată în [39]:

- cunoștințele acumulate de o persoană de-a lungul timpului nu pot fi reamintite destul de repede în caz de stres (mentenanță sub presiunea timpului)
- imaginile hibride furnizate prin RSC elimină și probabilele interpretări greșite sau diferite (de la persoană la persoană) ale documentației sau nerecunoașterea elementelor din sistem dacă sunt altfel decât în figurile din documentație
- pe parcursul timpului documentația care a fost livrată împreună cu produsul este modificată.
- materialul tipărit pentru documentație este dificil de manevrat perioade lungi de timp, se uzează

#### 2.1.4.11 Principii constructive de implementare ale structurilor redundante

În procesul de realizare a unei structuri redundante trebuie să se țină cont de principiul diversității echipamentului. În cazul unei structuri redundante este indicat să se folosească echipamente diferite pentru fiecare ramură. În acest fel se elimină defectele de mod comun care pot să apară la un moment dat în cadrul structurilor de același fel (fie ele hard sau soft).

Acest principiu în cadrul structurilor hardware implică folosirea de componente diferite pentru realizarea elementelor de pe fiecare ramură redundantă. În structurile software principiul înseamnă diversificarea modalității de realizare a programelor implementate, folosirea de algoritmi diferiți.

Pe cât posibil, pentru a mări gradul de siguranță, se urmărește ca în cazul redundanței duble pe o ramură elementul să fie realizat strict hardware, iar pe cealaltă ramură să existe o implementare software a aceluiași element.

Un alt principiu care trebuie avut în vedere în cazul în care se dorește funcționarea sigură a sistemului și în cazul unor dezastre este poziționarea componentelor care realizează redundanța în locații diferite (principiu realizabil doar pentru unele sisteme).

### **2.1.5 Aspecte privind utilizarea structurilor redundante în cazuri critice de dezastru și atacuri voite**

Redundanța contra dezastrelor este foarte importantă în protecția unui sistem. Dezastrele sunt fenomene imprevizibile care dacă se produc pot cauza defectarea gravă (de multe ori iremediabilă) a sistemului. De asemenea defectarea sistemului în astfel de situații conduce la punerea în pericol a mediului înconjurător și implicit a elementului uman.

Redundanța contra dezastrelor constă în existența unei structuri de închidere totală a sistemului, respectiv de trecere în stare de funcționare sigură cu minim de resurse utilizate sau de comutare spre o altă structură provizorie care să realizeze funcția sistemului pe timp de avarie. În momentul producerii unui dezastru sistemul de închidere trebuie să se poată activa și să se poată trece în starea sigură de funcționare de unde să se poată efectua diferite analize care să determine posibilitatea repunerii în funcțiune totală.

Redundanța contra atacurilor voite este un tip special de redundanță care trebuie perceput și aplicat în conformitate cu tipologia atacurilor la integritatea sistemului. Când se vorbește despre atacuri voite se au în vedere în primul rând atacurile care vizează sistemele software și transmisiile de date. Redundanța standard care se dezvoltă în vederea obținerii unui sistem tolerant la defecte nu poate rezolva întotdeauna problema atacurilor voite din două motive:

- dacă se folosesc sisteme identice, acestea prezintă aceleași tipuri de vulnerabilități, deci nu pot rezista la același tip de intruziune
- elementele cotate cu o probabilitate mică de defectare prezintă cu totul altă rată de defectare în cazul când atacurile se provoacă intenționat.

În [129] se propune redundanța dinamică diversificată care cuprinde arhitectură diversificată, funcții și implementare diversificată.

În cazul atacurilor voite există un anumit tipar după care se acționează. Atacatorul se ghidează după următorii pași:

- Încercarea – este practic căutarea de erori în sistem și analiza posibilității de intrare
- Explorarea – odată intrat, atacatorul explorează sistemul și resursele acestuia
- Exploatarea – în momentul în care intrusul găsește în sistem ceva util, face orice pentru a-l obține și se ajunge până la distrugerea sistemului.

Protecția contra unor astfel de atacuri urmărește de asemenea niște pași bine definiți:

- Rezistența – înseamnă protecția împotriva posibilității de acces în sistem. Rezistența este greu de realizat pentru că tehnicile atacatorilor sunt tot mai dezvoltate și pentru că într-o rețea (de ex. internet în cazul tele-operării sau tele-mentenanței) pot avea acces din ce în ce mai mulți utilizatori (cunoștințele de bază pentru a intra în rețea sunt deja la îndemâna oricui) și astfel atacul poate veni de la oricine.
- Detecția și evaluarea – se referă la detecția activității de atac asupra securității sistemului precum și la evaluarea stricăciunilor produse până

în acest moment. Această etapă este una pregătitoare înainte de a declanșa răspunsul sistemului.

- Reparația și comutarea – reparația sistemului se realizează doar în cazul în care defectele sunt minore, iar comutarea sistemului pe un alt sistem standby se produce în momentul în care stricăciunile sunt mari.
- Evoluție și adaptare – înseamnă îmbunătățirea siguranței în funcționare a sistemului în vederea unor atacuri ulterioare. Acest lucru se realizează prin memorarea modului de atac care s-a produs și adaptarea sistemului ca în viitor să fie stabil în cazul unor astfel de atacuri.

## 2.2 Studiul unei structuri redundante hibride – structura m-n-k

Există o mare varietate de structuri redundante iar clasificările existente, precum și cea propusă în subcapitolul 2.1 lasă loc liber combinării diferitelor variante. Astfel, considerăm că avantajele structurilor m out of n și standby pot fi reunite într-o structură îmbunătățită de tipul m-n-k (fig. 2.2.1) care face obiectul acestui subcapitol.

### 2.2.1 Redundanța hibridă m-n-k

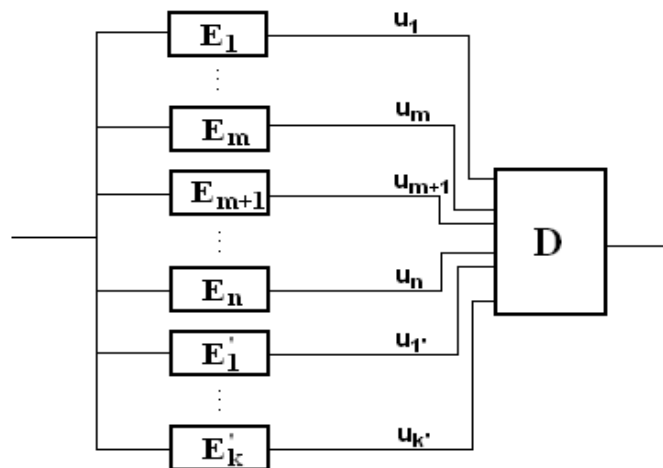


Fig. 2.2.1 Reprezentare schematică a redundanței hibride m-n-k

Structura conține  $n + k$  elemente notate cu  $E_1, \dots, E_n, E_1', \dots, E_k'$ . Funcția cerută de sistem este realizată de  $m$  elemente. În permanență, blocul decizional  $D$  analizează un număr de  $p$  elemente active ( $m \leq p \leq n$ ),  $n$  fiind numărul maxim de elemente active la un moment dat. Un număr de maximum alte  $k$  elemente de rezervă se află în regim de standby. Atâta timp cât  $k > 0$ , se analizează  $p = n$  elemente. În lipsa elementelor de rezervă, dar cel puțin în condițiile existenței a minimum  $m$  elemente funcționale,  $p$  va fi egal cu numărul tuturor elementelor funcționale. Principiul este sintetizat prin figura 2.2.2.

Sistemul se poate afla într-unul din următoarele regimuri de funcționare: normal, anormal acceptat și anormal neacceptat. Regimul normal înseamnă că  $p=n$  și că există cel puțin un element de rezervă. Regimul anormal acceptat apare dacă  $p < n$  dar totodată  $p \geq m$ , iar regimul anormal neacceptat este atunci când  $p < m$ .

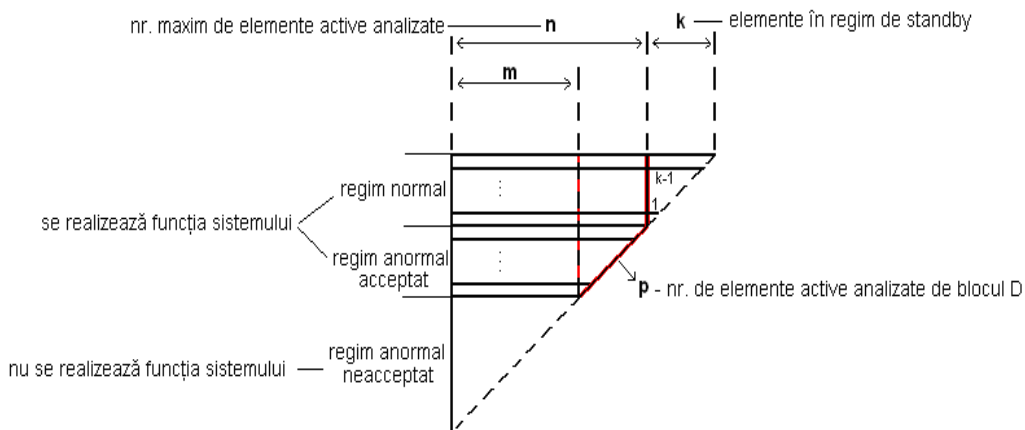


Fig. 2.2.2 Schema de funcționare a redundanței hibride m-n-k

Se observă că redundanța hibridă m-n-k prezintă următoarele avantaje care o plasează pe o poziție evident superioară structurilor m out of n sau standby: cunoașterea în permanență a elementelor defecte, posibilitatea activării entităților, posibilitatea dezactivării entităților în vederea reparației, gradul ridicat de fiabilitate etc. Din punctul de vedere al reacției sistemului la defecțiuni, redundanța hibridă m-n-k este de tip dinamic.

### 2.2.1.1 Variante de implementare a schemei de principiu

Figura 2.2.1 reprezintă o schemă de fiabilitate a structurii redundante m-n-k. Mărimea  $u_i$  reprezintă semnalul provenit de pe ramura elementului  $E_i$  și analizat de blocul decizional D. În figura 2.2.3 este reprezentat separat un singur element al structurii. Blocul D acționează asupra entității în funcție de varianta de implementare a structurii, cu ajutorul întrerupătoarelor 1 și 2.

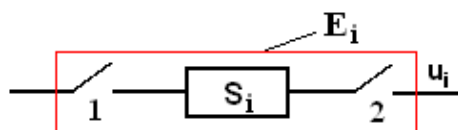


Fig. 2.2.3 Reprezentarea unui element al structurii redundante

Definim procesul de activare stânga ca fiind situația în care întrerupătorul 1 este închis și ca urmare elementul  $S_i$  este conectat la sursa de semnal, iar procesul de activare dreapta atunci când întrerupătorul 2 este închis și ca urmare blocul D preia semnalul furnizat de către  $S_i$ . Procesului de dezactivare înseamnă că întrerupătoarele se deschid și semnalele sunt întrerupte la intrarea sau la ieșirea elementului, în funcție de partea pe care se acționează. Atunci când se vorbește

despre activare, respectiv dezactivare stânga sau dreapta, acțiunea are în vedere doar întrerupătorul de pe acea parte.

Spunem că structura este *hot* atunci când întrerupătorul 1 este în permanență închis, este *cold* dacă întrerupătorul 2 este în permanență închis, sau *totală* în cazul în care ambele întrerupătoare există și pot fi în oricare din cele două stări. Implementarea structurii redundante constă practic în implementarea blocului decizional D, în concordanță cu una din cele trei variante de mai sus. Subcapitolul tratează situația cea mai dificilă, varianta *totală*.

În continuare, se detaliază modul de acțiune al blocului de decizie în raport cu entitățile sistemului. Blocul de decizie fiind implementat în întregime software (pe DSP, vezi subcapitolul 2.2.3), trebuie să conțină module de program care să trateze fiecare regim de funcționare în parte.

Blocul D analizează la un moment dat un număr  $p$  ( $m \leq p \leq n$ ) de semnale  $u_i$  ( $i=1;p$ ), determină dacă există vre-un defect și identifică elementele  $E_i$  defecte. Operația de identificare este esențială, permițând cunoașterea entității defecte, în vreme ce prin operația de determinare se află doar dacă există defecte, fără să nominalizeze respectivele elemente. Dacă sistemul se află în regimul de funcționare normal atunci blocul D activează stânga și dreapta următoarea entitate aflată în rezervă. Dacă sistemul se află în regim de funcționare normal sau anormal acceptat atunci entitățile defecte se dezactivează în vederea înlocuirii acestora.

Ideea implementată se axează pe compararea ieșirilor  $u_i$  ale elementelor între ele la fiecare moment de timp (pas de eșantionare).

Notăm pentru un pas de eșantionare:

$E_d$  -elementele care se defectează ( $d=1;q$ ),

$E_g$  -elementele în stare bună de funcționare ( $g=1;r$ ).

Evident  $q + r = p$ , iar modalitatea de abordare implementată se bazează pe principiul majorității, care înseamnă că totdeauna  $r \geq q$ .

Compararea entităților se face cu ajutorul unui factor ajustabil  $f$  (plaja permisă) astfel:

Dacă  $u_i \in [(1-f) \cdot u_j, (1+f) \cdot u_j]$  și  $u_j \in [(1-f) \cdot u_i, (1+f) \cdot u_i]$  atunci elementele  $E_i$  și  $E_j$  se află în același grup. Notând cu  $G_g$  - grupul majoritar de entități, rezultă că:

-  $E_g \in G_g$ ,

-  $E_d \notin G_g$ .

Un exemplu care ilustrează modalitatea de implementare a procedurii de analiză a elementelor  $E_1$  și  $E_2$  este descris prin următoarea secvență de pseudocod.

*Dacă ((absolut ( $u_1-u_2$ )  $\leq f \cdot u_1$ ) și ( absolut( $u_1-u_2$ )  $\leq f \cdot u_2$ )) atunci corect*

Dacă în urma analizei apare una din următoarele două situații speciale:

- nu se conturează clar  $G_g$  ( $\overline{G_g} = \overline{G_d}$ ),

-  $\forall E_j \in G_g, \exists E_i \in G_g$  astfel încât  $u_i \in [(1-f) \cdot u_j, (1+f) \cdot u_j]$  și  $u_j \in [(1-f) \cdot u_i, (1+f) \cdot u_i]$ , dar  $\exists E_j \in G_g, \exists E_x \in G_g$  astfel încât  $u_x \notin [(1-f) \cdot u_j, (1+f) \cdot u_j]$  și  $u_j \notin [(1-f) \cdot u_x, (1+f) \cdot u_x]$ , atunci este evident că au apărut defecte,

decizia se va lua cu ajutorul mediei din pasul precedent a ieșirilor entităților funcționale.

Constituirea grupului  $G_g$  reprezintă operația de selecție. În mometul în care  $G_g$  este definit urmează operația de utilizare. Utilizarea grupului  $G_g$  reprezintă



operația de realizare a funcției. Cele  $m$  elemente din grupul  $G_g$  care realizează funcția se vor alege cât mai aproape de media tuturor elementelor funcționale.

Notăm  $G_g'$  - grupul celor  $m$  elemente care realizează funcția. Astfel, cu ajutorul elementelor grupului  $G_g'$ , ieșirea furnizată ( $y$ ) de structura redundată în regimul de funcționare normal sau anormal acceptat va fi:

$$y = \text{Media curentă} = \frac{\sum_{g=1}^r u_g}{r}, \text{ iar pentru regimul anormal neacceptat } y=0.$$

### 2.2.1.2 Utilizarea structurii redundante m-n-k pentru realizarea unui bloc de măsurare

Structura redundată hibridă m-n-k poate fi utilizată, în particular, pentru realizarea unui bloc de măsurare. În acest caz elementele pot fi traductoare sau senzori. Organigrama din fig. 2.2.4 ilustrează operațiile efectuate în cadrul unui pas de eșantionare (fără filtrarea ieșirii, traductoarele fiind considerate deja calibrate) în cazul implementării unei astfel de structuri redundante hibride 2-3-1 (figura 2.2.4a).

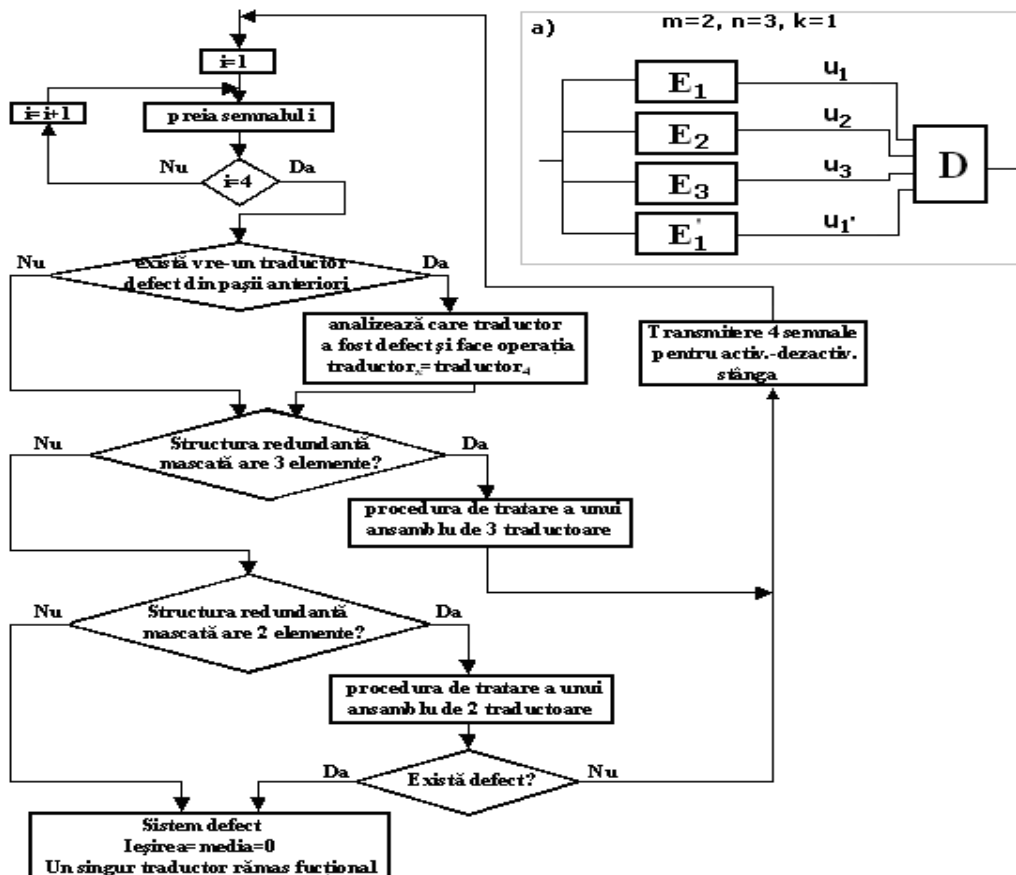


Fig. 2.2.4 Schema logică pentru redundanța hibridă 2-3-1

Ieșirea poate fi filtrată opțional cu un filtru de ordinul doi. Evident, ca o consecință a faptului că elementele care intră în structura redundantă nu sunt identice din diverse motive, va fi necesară prevederea unei operații de calibrare. O detaliere a operațiilor se face în continuare în cadrul paragrafului 2.2.2.

### 2.2.2 Implementarea redundanței hibride m-n-k pe DSP

În urma prezentării redundanței, acest subcapitol detaliază implementarea practică a structurii redundante hibride m-n-k din paragraful 2.2.1 pe DSP, implementare prezentată de autor în [63]. Structura particulară cu redundanță hibridă 2-3-1 este folosită ca bloc de măsurare.

În prima parte a paragrafului se prezintă arhitectura generală utilizată în vederea implementării, precum și modalitatea de interconectare a dispozitivelor fizice. Apoi se analizează posibilitatea de preluare a semnalelor de la traductoarele care intră în componența structurii, pentru a putea fi analizate. Paragrafele următoare ilustrează modalitatea de a transmite atât semnalul principal de ieșire a structurii redundante, cât și a celorlalte patru semnale de ieșire. Algoritm propriu-zis care constituie implementarea blocului decizional se prezintă în detaliu. Traductoarele care intră în componența structurii vor fi calibrate cu ajutorul unei amplificări de compatibilizare calculată la început, iar ieșirea este filtrată cu un filtru digital de ordinul doi. Finalul subcapitolului ilustrează rezultatele obținute.

#### 2.2.2.1 Arhitectura generală

Structura hardware folosită la implementare se compune din (Anexa 1 prezintă elementele teoretice ajutătoare legate de următoarele module fizice): modulul DSK (care include procesorul de semnal TMS320C31-50), multiplexorul analogic MAX307, numărătorul CD4040BE, respectiv demultiplexorul analogic CD4051BE. Figura 2.2.5 ilustrează modalitatea de interconectare a acestora.

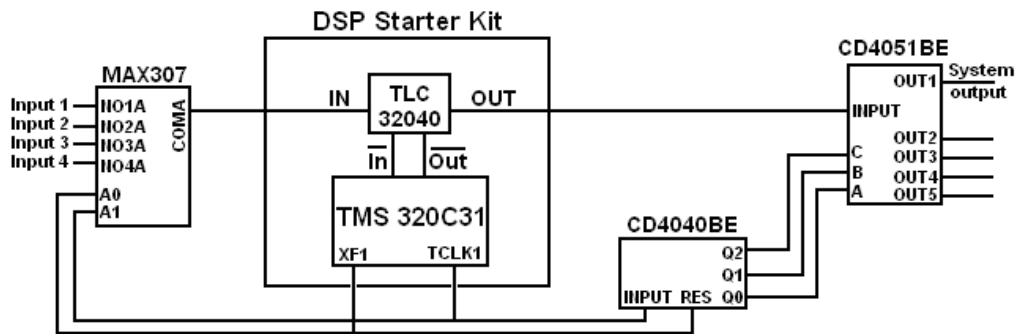


Fig. 2.2.5 Arhitectura generală a sistemului

Procesorul de semnal poate folosi un semnal de tact intern, dar este indicată utilizarea unui semnal de tact extern generat de un controller situat la un nivel superior în cadrul reglării, pentru a funcționa în corelație cu acesta. Un semnal de tact extern va fi obținut cu ajutorul temporizatorului 0 al procesorului de semnal și anume prin pinul TCLK0. Pentru a face ca temporizatorul să utilizeze pinul TCLK0 în vederea atingerii scopului menționat se configurează registrul de control global al

său astfel ca bitul CLKSCR să fie 0 și FUNC să fie 1. Astfel valoarea care se va încărca în registrul de control global este C1h:

```
LDI 0xC1, R0
LDI R0, @T0_ctrl
```

Semnalul analogic de la multiplexor se convertește în semnal numeric de către interfața analogic-numerică TLC32040, procesul invers regăsindu-se la ieșire. Activarea multiplexorului, a demultiplexorului și a interfeței analogic-numerică este realizată de către DSP, legăturile nefiind ilustrate în figură (pinul XF0 activează interfața analogic-numerică și multiplexorul, iar pinul XF1 demultiplexorul).

### 2.2.2.2 Achiziția semnalelor

Achiziția semnalelor se realizează cu ajutorul multiplexorului MAX307, prin controlul direct al procesorului de semnal TMS320C31-50. Algoritmul de achiziție a semnalelor urmărește selecția semnalelor prin configurarea pinului liber al procesorului de semnal XF1 și a ieșirii temporizatorului 1 (TCLK1), apoi procesul de preluare semnalelor realizată într-o porțiune de cod.

Pinul TCLK1 al temporizatorului 1 și pinul liber XF1 se conectează la pinii de selecție ai multiplexorului, A1, respectiv A0. Configurând în mod corespunzător acești pini se pot obține toate posibilitățile de selecție a celor 4 semnale de intrare. Evoluția semnalelor de selecție este ilustrată în figura 2.2.6.

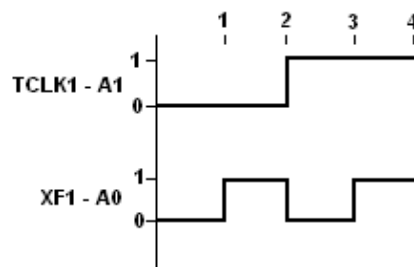


Fig. 2.2.6 Tranziția semnalelor de selecție

Pinul XF1 poate lua valoarea „0”, respectiv „1” setând registrul IOF în mod corespunzător, astfel:

```
LDI 38, IOF ; XF1 este setat pe valoarea „0”
LDI 102, IOF ; XF1 este setat pe valoarea „1”
```

Diferența între cazurile în care TCLK1 furnizează valoarea „0” sau „1” o face doar setarea bitului de inversare INV al registrului de control global al temporizatorului 1. Astfel, liniile de cod care realizează setările sunt:

```
LDI 0x3C1, R0
LDI R0, @T1_ctrl ; pinul TCLK1 este setat pe valoarea „1”
LDI 0x7C1, R0
LDI R0, @T1_ctrl ; pinul TCLK1 este setat pe valoarea „0”
```

Concepând un algoritm corespunzător pentru pinii de selecție ai multiplexorului, se pot prelua cele 4 semnale de intrare. Astfel, evoluția din figura 2.2.6 este obținută prin implementarea schemei logice din figura următoare:

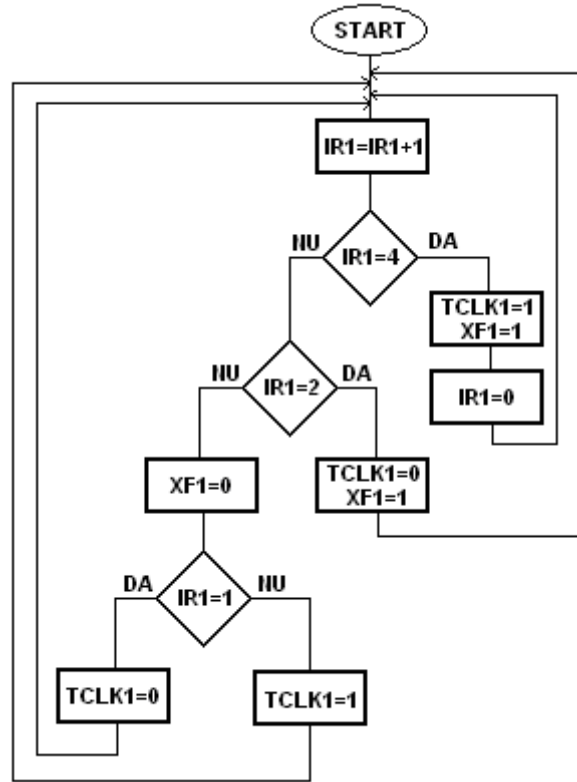


Fig. 2.2.7 Schema logică pentru preluarea semnalelor

Pentru exemplificarea modului de implementare se prezintă porțiunea de program care realizează activarea ultimei intrări (NO4A - A1=1, A0=0):

```

ADDI  1, IR1      ; Incrementează contorul
CMPI  IR1,R4     ; Compară starea contorului cu a registrului
BNE   DOI        ; Dacă diferă se sare la eticheta specificată
LDI   102,IOF    ; Setează pe XF1 = 1
LDI   0,IR1      ; Contorul se reinițializează la 0
LDI   0x7C1,R0   ; Se pornește temporizatorul 1 astfel
STI   R0,@T1_ctrl ; Încât la pinul de ieșire este TCLK1=0
b     FIN        ; Se face saltul la final
  
```

### 2.2.2.3 Transmiterea ieșirilor

După cum s-a specificat în capitolul anterior sistemul redundant trebuie să fie capabil să furnizeze la final, pe lângă ieșirea propriu-zisă a sistemului și semnale care să poată dezactiva sau activa traductoarele structurii redundante (în total sunt 5 ieșiri). Deoarece Starter Kit-ul are doar o ieșire, în vederea atingerii scopului se apelează la demultiplexorul CD4051BE. Ieșirea analogică a sistemului se conectează la pinul de intrare al acestuia. Soluția creează totodată o nouă problemă legată de numărul de biți (trei) de selecție necesari pentru a obține 5 ieșiri. Situația a fost

rezolvată prin utilizarea numărătorului CD4040BE, ale cărui prime trei ieșiri (Q1, Q2, Q3) sunt conectate la pinii de selecție ai demultiplexorului (C, B, A).

Numărătorul numără pe frontul căzător al semnalului conectat la pinul INPUT al său. Aici s-a legat pinul de ieșire al temporizatorului 1 (TCLK1). Procesul consideră o numărătoare până la 5 după care se resetează la 0. Resetarea se face cu ajutorul pinului XF1 conectat la pinul de RESET. Numărătorul se resetează la valoarea „1”.

#### 2.2.2.4 Algoritm propriu-zis

Algoritmul propriu-zis îi corespunde schema logică prezentată în fig. 2.2.4. Primul pas este inițializarea factorului ajustabil  $f$  (exemplu  $f=1/100$ ). În cele ce urmează se descrie algoritmul implementat de la momentul în care semnalele sunt preluate până în momentul transmiterii ieșirii, respectiv a semnalelor de activare-dezactivare a traductoarelor. Întâi, dacă se analizează puțin cazul general, se reamintește ideea că tendința sistemului este de a păstra regimul normal ( $p=n$ ) cât se poate de mult. S-ar putea concepe sisteme care să nu activeze elemente aflate în rezervă decât în momentul defectării mai multor traductoare, situație care complică excesiv metodele de programare ajungându-se la timpi de execuție mai mari și în special la necesitatea unui spațiu de memorie mult mai mare pentru a stoca informațiile.

Maniera de implementare oferă o posibilitate de extindere a redundanței hibride 2-3-1 la tipuri de complexitate superioară, precum și o structurare și o ierarhizare clară a subrutinelor. Astfel, programul conține o procedură generală divizată în trei părți (fig. 2.2.4), două dintre ele tratând ansambluri de două, respectiv trei traductoare active, iar ultima, situația în care sistemul este defect. Extinzând raționamentul, pentru o redundanță hibridă 2-3-k n-ar mai fi necesare decât simple proceduri de rețușare, corpul principal ar rămâne același. Astfel, în exemplul curent procedura de tratare a unui ansamblu de 3 traductoare se apelează pe tot parcursul regimului normal de funcționare.

După principiul prezentat în paragraful 2.2.1, mai întâi se analizează informațiile primite de la traductoarele active. Dacă un traductor se defectează, se acționează în funcție de regimul de funcționare curent, cu ajutorul demultiplexorului.

Complicații apar dacă în cazul analizei a trei traductoare active se ajunge la a doua situație specială descrisă în cap. 2.2.1. În acest caz este clar că un traductor este defect (presupunem traductorul 3 sau 1) și se recurge la determinarea celor două funcționale cu ajutorul mediei din pasul precedent. Astfel, dintre cele două traductoare se alege „cel bun” ca fiind cel mai apropiat de medie.

```

LDF    @medie,R4    ; se încarcă media precedentă în R4
SUBF   R4,R1        ; se calculează diferența trad1 - medie
SUBF   R4,R3        ; se calculează diferența trad3 - medie
ABSF   R1,R1        ;
ABSF   R3,R3        ; valorile absolute ale diferenței
CMPF   R3,R1        ; compară cele două variante
BGE    C1           ; salt la C1 dacă traductorul 1 este defect
LDF    12,R0        ; traductorul 3 este defect
STF    R0,@a        ; reține o valoare specifică într-o locație
B      FA123        ; salt la finalul subrutinei
C1     LDF    23,R0 ; idem dacă traductorul 1 este defect

```

STF R0,@a  
B FA123

În permanență semnalul furnizat la ieșirea sistemului va fi media semnalelor din grupul  $G_g$  extras după definirea grupului de bază  $G_g$ .

În momentul în care rămân doar două traductoare funcționale se intră automat în prima situație specială descrisă în cap. 2.2.1, iar în acest caz dacă este necesar se decide cu ajutorul mediei din pasul precedent care traductor este defect, iar în următorul pas întregul sistem se dezactivează.

### 2.2.2.5 Amplificarea de compatibilizare

Elementele care intră în componența structurii redundante nu sunt identice, iar chiar dacă se consideră după catalog identice, rareori produsele care ies din fabrică au o evoluție perfect similară. Acest fapt poate să determine ca algoritmul structurii redundante să conducă la decizii eronate, deoarece el se bazează pe sesizarea de deviații corecte ale ieșirilor elementelor.

În prima etapă este necesară o compatibilizare a ieșirilor traductoarelor. Se pornește de la ipoteza că, la început, fiecare element al structurii este funcțional. În prima etapă a funcționării, există opțiunea de a selecta care traductor să reprezinte referința în funcție de care să se facă compatibilizarea. În ipoteza că pentru compatibilizare se acționează multiplicativ, se calculează amplificările de compatibilizare pentru fiecare element al structurii redundante. Aceste amplificări de compatibilizare sunt setate și păstrate pe toată durata funcționării, iar astfel evoluția după algoritmul prezentat nu va fi periclitată.

În locația dată de variabila „which” este setat numărul traductorului după care se va face compatibilizarea. În cazul curent ieșirile traductoarelor vor fi citite în primii patru regiștrii, iar în R5 va fi încărcată valoarea variabilei „which”. Prin comparații succesive se alege care din regiștrii va reprezenta numărătorul amplificărilor de compatibilizare, și se încarcă în registrul R5. Următorul pas este calculul amplificărilor de compatibilizare, iar ca exemplu se dă amplificarea „k1” pentru traductorul 1:

LDF R1, R0 ;Încarcă valoarea traductorului 1 în reg. R0  
CALL INV ;Se apelează subrutina de inversare  
MPYF R5, R0 ;Înmulțește cu valoarea anterior stocată în R5  
STF R0, @K1 ;Se stochează rezultatul în K1

### 2.2.2.6 Filtrarea ieșirii

Filtrarea ieșirii se va face cu ajutorul unui filtru numeric de ordinul 2. Pentru a filtra semnalele de frecvență mai mare se va utiliza un filtru trece-jos Butterworth [25] cu frecvența de tăiere ajustabilă. Forma unui astfel de filtru este:

$$y = a_1y_1 + a_2y_2 + b_0u + b_1u_1 + b_2u_2$$

unde:

y – ieșirea filtrată

u – intrarea în filtru (ieșirea nefiltrată)

$y_1, y_2, u_1, u_2$  – ieșirile filtrate și nefiltrate din pașii anteriori (momentele de timp t-1 și t-2)

Având ordinul  $n=2$  al filtrului și frecvența de tăiere se pot determina coeficienții  $a$  și  $b$  foarte simplu utilizând de exemplu mediul Matlab, și anume instrucțiunea  $[b, a] = \text{butter}(n, w_n)$ .  $w_n$  trebuie să fie sub formă de procent din frecvența Nyquist.

### 2.2.2.7 Rezultate obținute

Rezultatele studiului realizat în cadrul paragrafului 2.2 s-au diseminat prin lucrarea [63].

În cele ce urmează se prezintă un exemplu de funcționare a structurii redundante. Structura implementată pe DSP primește informațiile de la traductoare sub forma unor tensiuni (aplicabilitatea structurii redundante nu depinde de tipul traductoarelor utilizate). În acest context se generează semnalul de intrare ilustrat în figura 2.2.8, care reprezintă o evoluție în intervalul 0-12 V. Pentru a evidenția eficiența implementării structurii redundante, semnalul de intrare se consideră variabil în timp, și ușor perturbat.

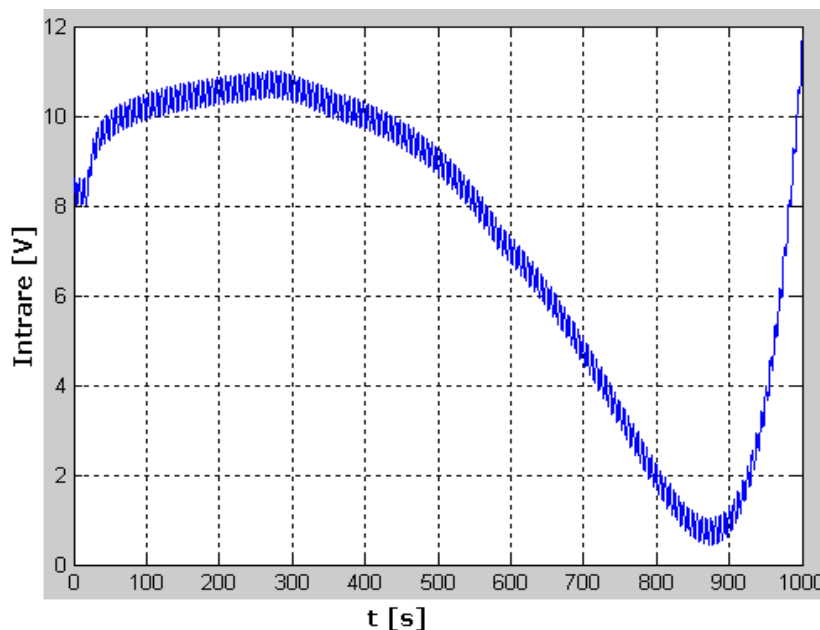


Fig. 2.2.8 Semnalul de intrare

Scenariul se prezintă în felul următor: inițial primele trei traductoare sunt active, iar primul care se defectează este traductorul 1 după 200s de funcționare. Acesta se dezactivează ca urmare a defectului și se activează traductorul 4. Următorul care se defectează este chiar noul intrat în funcțiune, traductorul 4 la momentul  $t=500s$ . Din acest moment analiza cuprinde doar elementele 1 și 3. După o perioadă de timp ( $t=800s$ ) traductorul 3 iese din funcțiune ca urmare a unui defect, rămânând activ un doar traductorul 2. Din acest moment un singur element este activ, condiție necorespunzătoare, iar sistemul se dezactivează. Evoluția celor patru traductoare este ilustrată în figura 2.2.9, iar ieșirea sistemului este prezentată în figura 2.2.10.

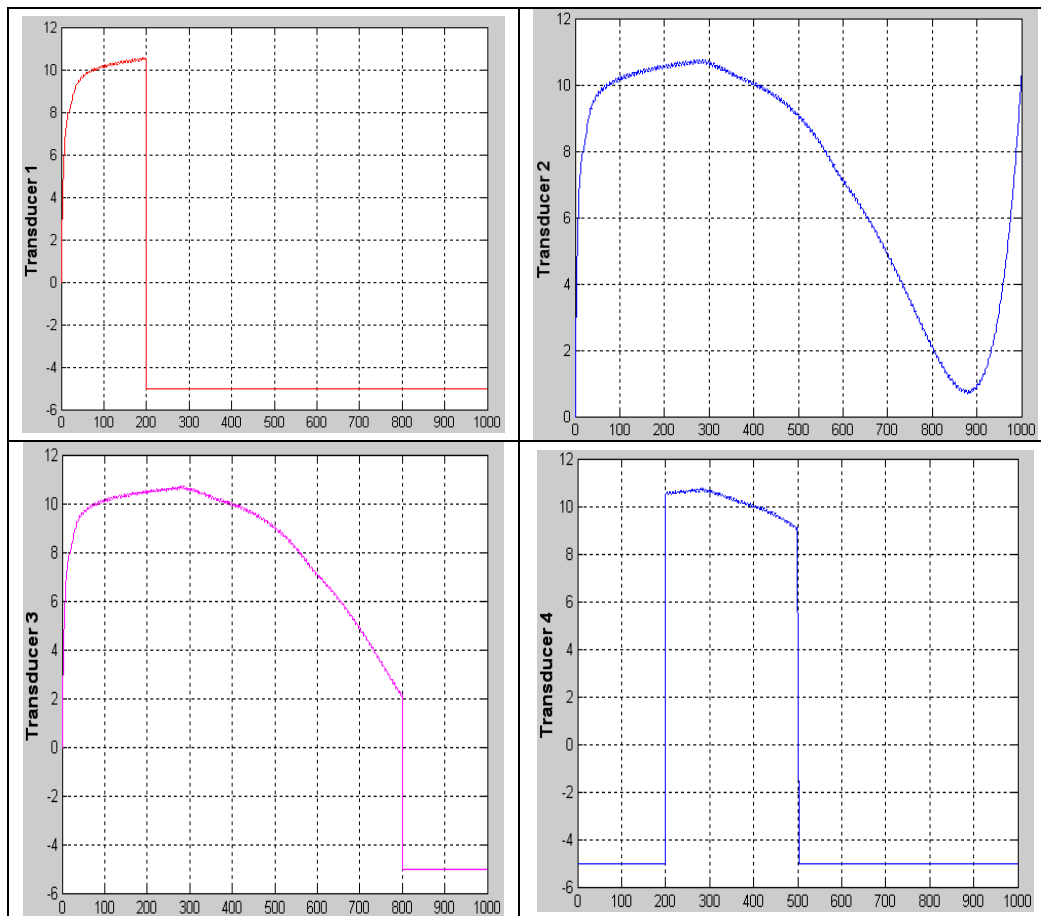


Fig. 2.2.9 Evoluția celor patru traductoare

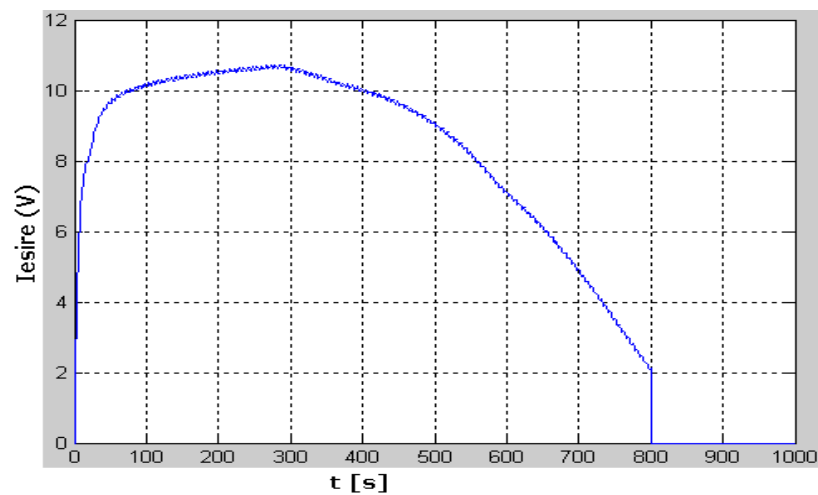


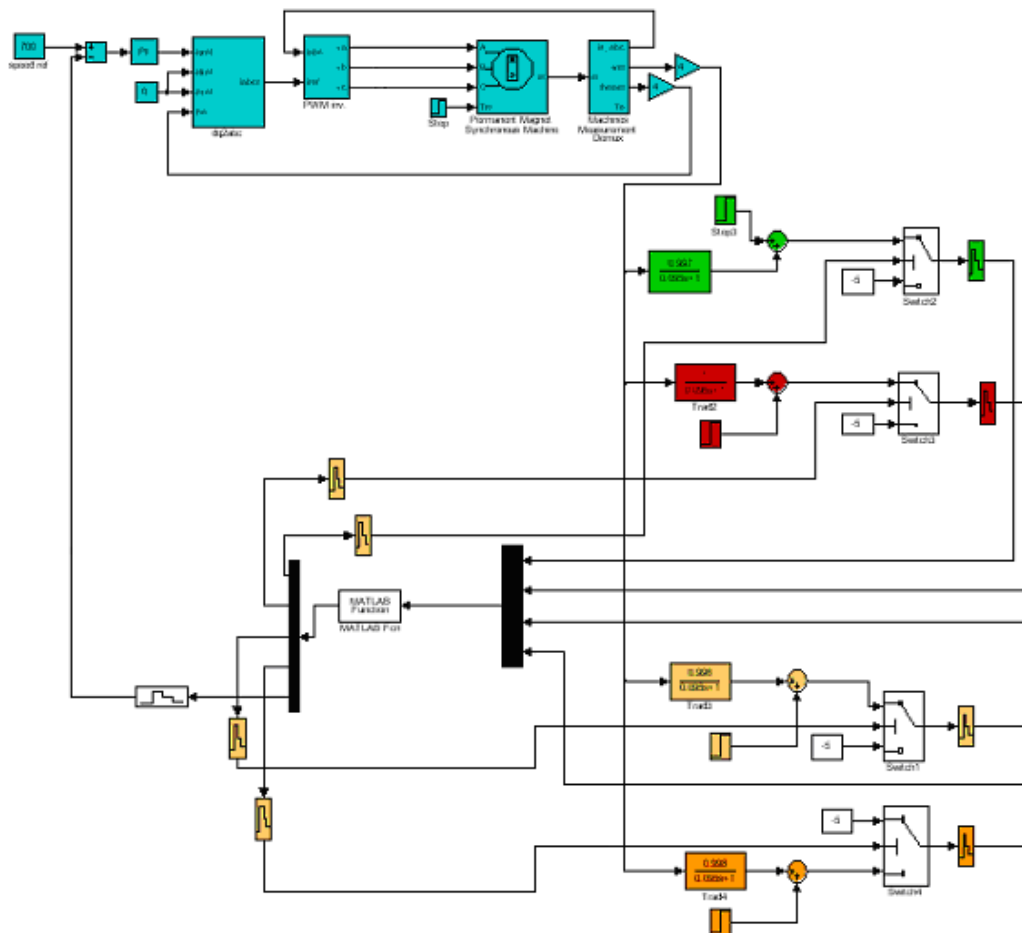
Fig. 2.2.10 Ieșirea sistemului



### 2.2.3 Utilizarea structurii cu redundanță hibridă 2-3-1 în cadrul sistemului de conducere al unei mașini electrice

În principiu, redundanța hibridă m-n-k este aplicabilă pentru orice sistem. În acest paragraf se ilustrează utilizarea structurii cu redundanță hibridă 2-3-1 (vezi paragrafele 2.2.1, 2.2.2) pentru o mașină electrică și anume un motor sincron cu magneți permanenți (MSMP).

Structura redundanță se utilizează pentru blocul de măsurare a turației sistemului de reglare a turației motorului. Elementele structurii redundante sunt traductoare de viteză și au ca intrare viteza unghiulară. Viteza unghiulară prescrisă este de 700 rad/s, iar sistemul redundanț este amplasat pe calea de reacție a buclei de reglare. Modelul întregului sistem este ilustrat în figura 2.2.11. Autorul sintetizează în acest paragraf studiul prezentat în [64].



2.2.11 Motor sincron cu magneți permanenți prevăzut cu o structură redundanță hibridă 2-3-1.

Traductoarele sunt elemente de tip PT1, iar modelul motorului sincron cu magneți permanenți se evidențiază prin culoarea albastru deschis.

Scenariul considerat este următorul:

- sistemul funcționează cu succes până la un anumit moment de timp ( $t=1s$ ), când se defectează traductorul 3. În acest moment, elementul al treilea este dezactivat total (atât stânga cât și dreapta), iar elementul aflat în rezervă se activează și intră în acțiune. După un interval de timp (la momentul  $t=1,6s$ ) se defectează chiar traductorul 4, activat ultimul. În această situație elementul defect se dezactivează, iar sistemul se bazează doar pe două entități. Deoarece funcția din sistem este realizabilă de 2 entități sistemul continuă să funcționeze. În final traductorul al doilea se defectează ( $t=2,2s$ ) și este dezactivat. Având în vedere că funcția nu se poate realiza cu un singur element funcțional, sistemul se dezactivează.

Evoluția celor patru traductoare este ilustrată în figurile 2.2.12, 2.2.13, 2.2.14 și 2.2.15.

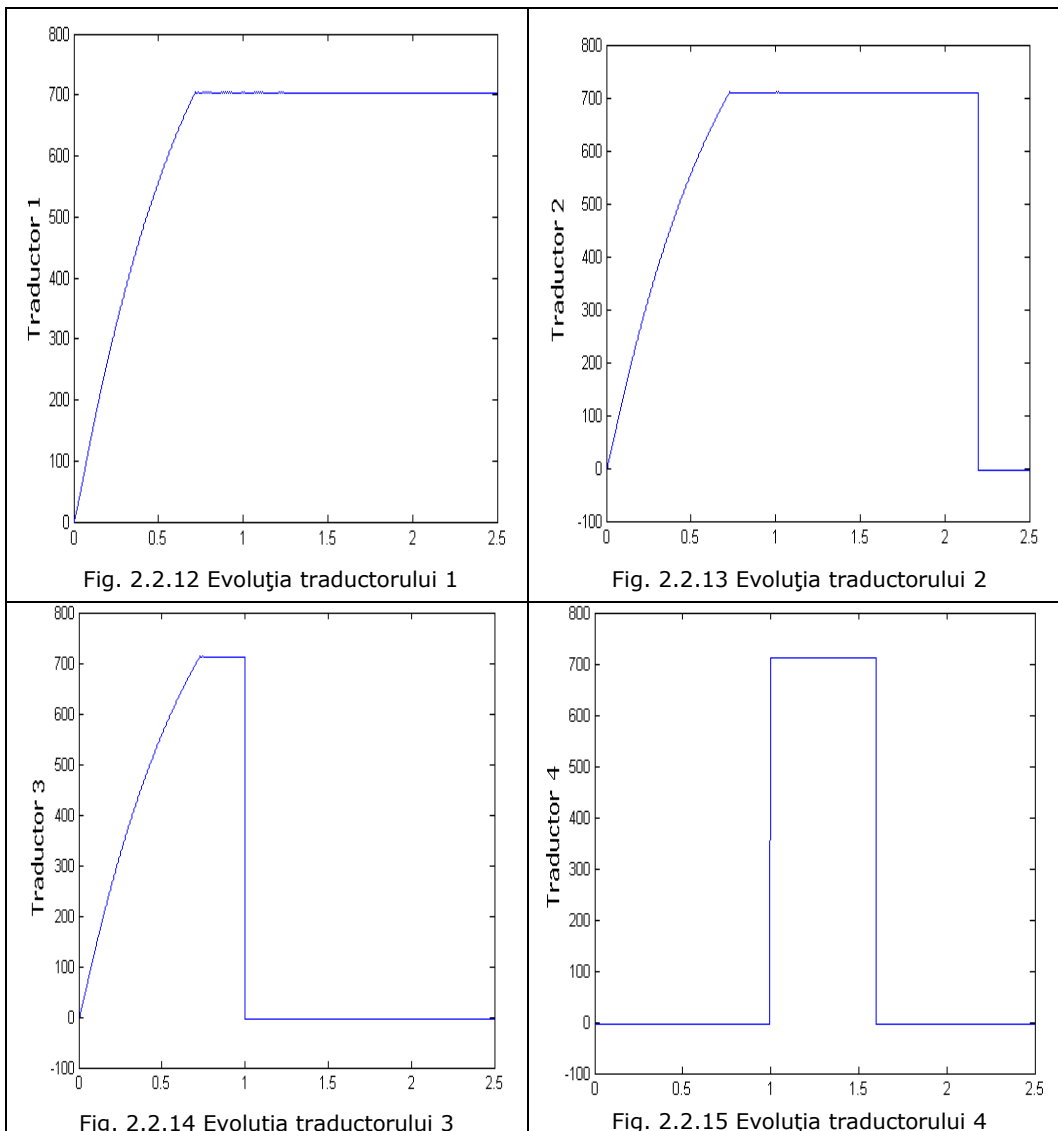


Figura 2.2.16 prezintă graficul ieșirii furnizate de structura cu redundanță hibridă 2-3-1.

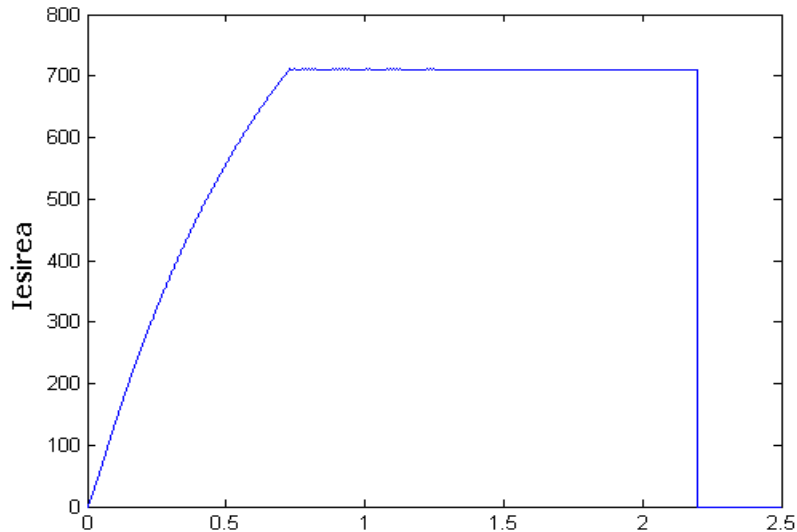


Fig. 2.2.16 Evoluția ieșirii structurii redundante

## 2.3 Studiul structurilor cu redundanță activă omogenă cu k componente reparabile

### 2.3.1 Structura redundantă activă cu k entități reparabile identice, aspecte referitoare la ratele de tranziție

Aplicarea redundanței asupra unui sistem conduce la mărirea fiabilității/disponibilității. Definițiile fiabilității și disponibilității au fost redată în cadrul subcapitolului 1.2, conceptul de fiabilitate folosindu-se preponderent în sisteme nereparabile sau parțial reparabile, iar cel de disponibilitate în cazul sistemelor total reparabile (în acest caz utilizarea termenului de fiabilitate își pierde sensul).

În figura 2.3.1 se ilustrează o structură redundantă omogenă de ordin k (redundanța omogenă este definită în cadrul paragrafului 2.1.4.2). Pentru îndeplinirea funcției din sistem este necesară o singură entitate (componentă).

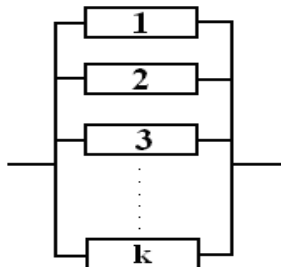


Fig. 2.3.1 Structură redundantă de ordin k

Pentru a calcula și a studia cu cea mai multă exactitate fiabilitatea, respectiv disponibilitatea unei structuri este necesar să se cunoască ratele de defectare și de reparare ale componentelor. Presupunem că cele  $k$  componente sunt identice și că se consideră fiecare componentă în parte. Fie  $\lambda$  rata de defectare individuală și  $\mu$  rata de reparare individuală.

Sistemul privit ca întreg are pe de altă parte  $k+1$  stări și anume: starea de funcționare totală, starea în care o componentă este defectă, starea în care două componente sunt defecte, ș.a.m.d. În structura cu redundanță de ordin  $k$ , singura stare în care sistemul nu este disponibil este cea cu toate componentele defecte. Tranziția sistemului privit ca întreg dintr-o stare în alta se face corespunzător ratelor de defectare și de reparare specifice.

Subiectul acestui paragraf este de a prezenta relații între cele  $k$  rate de defectare și de reparare ale sistemului privit ca întreg (care caracterizează tranzițiile între stări succesive) și intensitățile de defectare și de reparare individuale ale componentelor.

Dinamica fiabilității, disponibilității sau a altor atribute de calitate a sistemelor se poate studia folosind modelele dinamice în timp discret sau în timp continuu, cunoscute sub denumirea de modele Markov [47], [58], [73], [14]. În abordarea corectă a unei probleme trebuie distinse două etape, cea de construcție a modelului și cea de utilizare a modelului. Construcția se bazează, pe de-o parte, pe alcătuirea unui model conceptual care redă calitativ procesele care au loc, iar pe de altă parte, pe transpunerea acestui model conceptual într-un model matematic. La rândul ei, utilizarea se bazează pe încadrarea corectă a modelului matematic în conceptele teoriei sistemelor, supunerea acestuia unor eventuale transformări și folosirea efectivă a modelului pentru rezolvarea problemei care interesează, această ultimă fază incluzând coroborarea calculului cu interpretarea rezultatelor.

În cadrul paragrafului 1.7 s-a precizat faptul că diverse studii au fost efectuate în cadrul unui grup de cercetare în cadrul căruia a lucrat și autorul tezei. Studiile din cadrul acestui capitol au fost diseminate prin lucrările [34], [35]. În acest context, în cele ce urmează se face doar o prezentare sintetică a studiilor.

În fig. 2.3.2 este reprezentată structura redundantă cu  $k$  entități reparabile asociată sistemului din figura 2.3.1 pentru modelare Markov. Structura se poate găsi în  $k + 1$  stări notate cu  $\textcircled{0}, \textcircled{1}, \textcircled{2}, \dots, \textcircled{k}$ . S-au notat cu  $\lambda_{i,i+1}$ ,  $i=0, 1, 2, \dots, k-1$ , ratele de defectare corespunzătoare tranzițiilor din starea  $\textcircled{i}$  în starea  $\textcircled{i+1}$  și cu  $\mu_{i+1,i}$  ratele de reparare corespunzătoare tranzițiilor din starea  $\textcircled{i+1}$  în starea  $\textcircled{i}$ . Studiul detaliat face obiectul lucrării [35], paragraful de față reținând doar ipoteza și rezultatul cercetării.

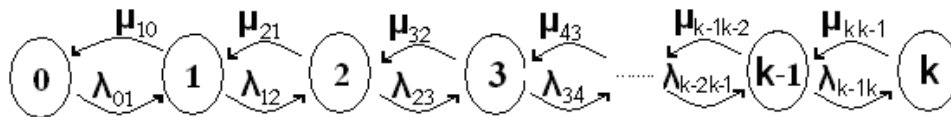


Fig. 2.3.2 Modelul Markov al structurii redundante ilustrată în fig. 2.3.1

Procesele tranzitorii asociate probabilităților celor  $k+1$  stări prezentate în fig. 2.3.2 pot fi descrise de ecuațiile următoare care țin cont de intensitățile de defectare și reparare:

$$\begin{cases} \dot{P}_0(t) = -\lambda_{01}P_0(t) + \mu_{10}P_1(t) \\ \dot{P}_1(t) = \lambda_{01}P_0(t) - (\lambda_{12} + \mu_{10})P_1(t) + \mu_{21}P_2(t) \\ \dot{P}_2(t) = \lambda_{12}P_1(t) - (\mu_{21} + \lambda_{23})P_2(t) + \mu_{32}P_3(t) \\ \dots \\ \dot{P}_i(t) = \lambda_{i-1,i}P_{i-1}(t) - (\mu_{i,i-1} + \lambda_{i,i+1})P_i(t) + \mu_{i+1,i}P_{i+1}(t) \\ \dots \\ \dot{P}_k(t) = \lambda_{k-1,k}P_{k-1}(t) - \mu_{k,k-1}P_k(t) \end{cases} \quad (1)$$

În regim staționar, o entitate se poate găsi în doar două stări,  $\textcircled{0}$  și  $\textcircled{1}$ ,  $\lambda$  și  $\mu$  fiind intensitățile de defectare și reparare individuale. Probabilitățile de regim staționar ale celor două stări se obțin cu formulele [47], [73]:

$$P_{0,\infty} = \frac{\mu}{\lambda + \mu}, \quad P_{1,\infty} = \frac{\lambda}{\lambda + \mu} \quad (2)$$

În urma studiului prezentat în [35], între ratele de defectare și de reparare necunoscute, asociate tranzițiilor din figura 2.3.2 (k rate de defectare  $\lambda_{i,i+1}$ ,  $i=0, 1, \dots, k-1$ , și ratele de reparare  $\mu_{i,i-1}$ ,  $i=1, 2, \dots, k$ ) există următoarele relații de legătură:

$$\begin{cases} \lambda_{01} = \frac{k}{1} \frac{\lambda}{\mu} \mu_{10} \\ \lambda_{12} = \frac{k-1}{2} \frac{\lambda}{\mu} \mu_{21} \\ \dots \\ \lambda_{i,i+1} = \frac{C_k^{i+1}}{C_k^i} \frac{\lambda}{\mu} \mu_{i+1,i} \\ \dots \\ \lambda_{k-1,k} = \frac{1}{k} \frac{\lambda}{\mu} \mu_{k,k-1} \end{cases} \quad (3)$$

Se observă că în egalitățile (3) apar k egalități cu 2k necunoscute. În consecință, un număr de k intensități diferite de  $\lambda$  și  $\mu$  trebuie să se determine experimental. Mărimile alese în acest scop sunt intensitățile de reparare  $\mu_{10}, \mu_{21}, \mu_{32}, \dots, \mu_{k,k-1}$ , mai ușor de monitorizat. Presupunând că valorile lor sunt cunoscute, din relațiile (3) rezultă posibilitatea de a determina ratele de defectare  $\lambda_{01}, \lambda_{12}, \lambda_{23}, \dots, \lambda_{k-1,k}$ .

Modalitatea de determinare experimentală a ratelor de reparare  $\mu_{10}, \mu_{21}, \mu_{32}, \dots, \mu_{k,k-1}$ , pentru un sistem format din k entități reparabile este ilustrată în [35]. Rezultatul se exprimă prin ecuațiile (4) și (5).

$$\mu_{i,i-1} = \frac{1}{MDT_i}, \quad i=1;k \quad (4)$$

$$MDT_i = \frac{1}{N_{DT_i}} \sum_{j=1}^n T_{DT_{ij}} \cdot \quad (5)$$

$\mu_{i,j-1}$  este rata de reparare corespunzătoare tranziției sistemului din starea  $(i)$  în starea  $(j)$ , iar  $MDT_i$  este timpul mediu de defectare corespunzător cazului în care  $i$  entități din cele  $k$  ale sistemului redundant sunt defecte și urmează a se repara una din cele  $i$  fără să se defecteze alta din cele rămase. Timpii  $T_{DT_{ij}}$  se consideră corespunzător acelor intervale de timp în care  $i$  entități din  $k$  sunt defecte.

### 2.3.2 Studiu de caz: un sistem redundant cu 3 entități reparabile

Se consideră un sistem format din trei entități reparabile, căruia îi corespund tranzițiile prezentate în figura de mai jos (Fig. 2.3.3):

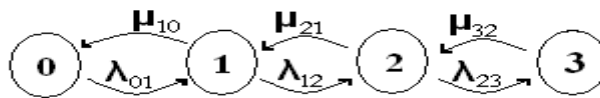


Fig. 2.3.3 Modelul pentru structura redundantă cu trei entități reparabile

Procesele tranzitorii asociate probabilităților celor patru stări prezentate în fig. 2.3.3 pot fi descrise de ecuațiile următoare care țin cont de intensitățile de defectare și reparare corespunzătoare:

$$\begin{aligned} P_0(t + \Delta t) &= P_0(t)(1 - \lambda_{01}\Delta t) + P_1(t)\mu_{10}\Delta t \\ P_1(t + \Delta t) &= P_0(t)\lambda_{01}\Delta t + P_1(t)(1 - \mu_{10}\Delta t) (1 - \lambda_{12}\Delta t) + P_2(t)\mu_{21}\Delta t \\ P_2(t + \Delta t) &= P_1(t)\lambda_{12}\Delta t + P_2(t)(1 - \mu_{21}\Delta t) (1 - \lambda_{23}\Delta t) + P_3(t)\mu_{32}\Delta t \\ P_3(t + \Delta t) &= P_2(t)\lambda_{23}\Delta t + P_3(t)(1 - \mu_{32}\Delta t) \end{aligned} \quad (6)$$

Prin trecere la limită pentru  $\Delta t \rightarrow 0$  obținem:

$$\begin{cases} \dot{P}_0(t) = -\lambda_{01}P_0(t) + \mu_{10}P_1(t) \\ \dot{P}_1(t) = \lambda_{01}P_0(t) - (\lambda_{12} + \mu_{10})P_1(t) + \mu_{21}P_2(t) \\ \dot{P}_2(t) = \lambda_{12}P_1(t) - (\mu_{21} + \lambda_{23})P_2(t) + \mu_{32}P_3(t) \\ \dot{P}_3(t) = \lambda_{23}P_2(t) - \mu_{32}P_3(t) \end{cases} \quad (7)$$

Adunând membru cu membru aceste relații se obține:

$$\dot{P}_0(t) + \dot{P}_1(t) + \dot{P}_2(t) + \dot{P}_3(t) = 0 \quad (8)$$

Întrucât existența ansamblului este o certitudine din relația (8) rezultă:

$$P_0(t) + P_1(t) + P_2(t) + P_3(t) = 1 \quad (9)$$

În regim staționar ecuațiile (7) corespunzătoare sistemului devin:

$$\begin{cases} -\lambda_{01}P_{0,\infty} + \mu_{10}P_{1,\infty} = 0 \\ \lambda_{01}P_{0,\infty} - (\lambda_{12} + \mu_{10})P_{1,\infty} + \mu_{21}P_{2,\infty} = 0 \\ \lambda_{12}P_{1,\infty} - (\mu_{21} + \lambda_{23})P_{2,\infty} + \mu_{32}P_{3,\infty} = 0 \\ \lambda_{23}P_{2,\infty} - \mu_{32}P_{3,\infty} = 0 \end{cases} \quad (10)$$

Totodată, din relația (9) rezultă:

$$P_{0,\infty} + P_{1,\infty} + P_{2,\infty} + P_{3,\infty} = 1 \quad (11)$$

Determinarea celor patru valori necunoscute de regim staționar  $P_{0,\infty}$ ,  $P_{1,\infty}$ ,  $P_{2,\infty}$  respectiv  $P_{3,\infty}$  se realizează rezolvând sistemul de ecuații format din (10) și (11), valorile obținute fiind:

$$\begin{cases} P_{1,\infty} = P_{0,\infty} \frac{\lambda_{01}}{\mu_{10}} \\ P_{2,\infty} = P_{0,\infty} \frac{\lambda_{01}\lambda_{12}}{\mu_{10}\mu_{21}} \\ P_{3,\infty} = P_{0,\infty} \frac{\lambda_{01}\lambda_{12}\lambda_{23}}{\mu_{10}\mu_{21}\mu_{32}} \\ P_{0,\infty} + P_{1,\infty} + P_{2,\infty} + P_{3,\infty} = 1 \end{cases} \quad (12)$$

Determinarea valorilor de regim staționar  $P_{0,\infty}$ ,  $P_{1,\infty}$ ,  $P_{2,\infty}$  respectiv  $P_{3,\infty}$  se realizează rezolvând (12), valorile obținute fiind:

$$\begin{cases} P_{0,\infty} = \frac{\mu_{10}\mu_{21}\mu_{32}}{\mu_{10}\mu_{21}\mu_{32} + \lambda_{01}\lambda_{12}\mu_{32} + \lambda_{01}\mu_{21}\mu_{32} + \lambda_{01}\lambda_{12}\lambda_{23}} \\ P_{1,\infty} = \frac{\lambda_{01}\mu_{21}\mu_{32}}{\mu_{10}\mu_{21}\mu_{32} + \lambda_{01}\lambda_{12}\mu_{32} + \lambda_{01}\mu_{21}\mu_{32} + \lambda_{01}\lambda_{12}\lambda_{23}} \\ P_{2,\infty} = \frac{\lambda_{01}\lambda_{12}\mu_{32}}{\mu_{10}\mu_{21}\mu_{32} + \lambda_{01}\lambda_{12}\mu_{32} + \lambda_{01}\mu_{21}\mu_{32} + \lambda_{01}\lambda_{12}\lambda_{23}} \\ P_{3,\infty} = \frac{\lambda_{01}\lambda_{12}\lambda_{23}}{\mu_{10}\mu_{21}\mu_{32} + \lambda_{01}\lambda_{12}\mu_{32} + \lambda_{01}\mu_{21}\mu_{32} + \lambda_{01}\lambda_{12}\lambda_{23}} \end{cases} \quad (13)$$

În relația (13) se fac următoarele notații:

$$\begin{cases} \mu_{10}\mu_{21}\mu_{32} = \tilde{\mu} \\ \lambda_{01}\lambda_{12}\lambda_{23} = \tilde{\lambda} \\ \lambda_{01}\mu_{21}\mu_{32} = \tilde{\lambda}\tilde{\mu} \\ \lambda_{01}\lambda_{12}\mu_{32} = \tilde{\lambda}\tilde{\lambda}\tilde{\mu} \end{cases} \quad (14)$$

După cum apare și în paragraful 2.3.1,  $\lambda$  și  $\mu$  sunt intensitățile de defectare și reparare individuale (din cazul unei singure entități) corespunzătoare cazului în care sistemul se poate găsi doar în două stări,  $\textcircled{0}$  și  $\textcircled{1}$ , cu probabilitățile descrise de relația (2).

În ceea ce privește ansamblul de entități, întrucât evoluția fiecărei entități este independentă de a celorlalte entități din ansamblul de 3, sunt valabile următoarele observații:

- ✓ Probabilitatea ca sistemul să fie în starea  $\textcircled{0}$  este egală cu produsul probabilităților ca fiecare din entități să fie în starea  $\textcircled{0}$ .

$$P_{0\infty} = \tilde{P}_{0\infty} \tilde{P}_{0\infty} \tilde{P}_{0\infty} = \frac{\mu^3}{(\lambda + \mu)^3} \quad (15.1)$$

- ✓ Probabilitatea ca sistemul să fie în starea  $\textcircled{1}$  rezultă din probabilitatea ca oricare dintre cele trei entități ale sistemului să fie în starea  $\textcircled{1}$  și celelalte două în starea  $\textcircled{0}$ .

$$P_{1\infty} = 3\tilde{P}_{0\infty} \tilde{P}_{0\infty} \tilde{P}_{1\infty} = 3 \frac{\mu^2 \lambda}{(\lambda + \mu)^3} \quad (15.2)$$

- ✓ Probabilitatea ca sistemul să fie în starea  $\textcircled{2}$  rezultă din probabilitatea ca oricare dintre cele trei entități ale sistemului să fie în starea  $\textcircled{0}$  și celelalte două în starea  $\textcircled{1}$ .

$$P_{2\infty} = 3\tilde{P}_{1\infty} \tilde{P}_{1\infty} \tilde{P}_{0\infty} = 3 \frac{\mu \cdot \lambda^2}{(\lambda + \mu)^3} \quad (15.3)$$

- ✓ probabilitatea ca sistemul să fie în starea  $\textcircled{3}$  rezultă din probabilitatea toate cele trei entități ale sistemului să fie în starea  $\textcircled{1}$ .

$$P_{3\infty} = \tilde{P}_{1\infty} \tilde{P}_{1\infty} \tilde{P}_{1\infty} = \frac{\lambda^3}{(\lambda + \mu)^3} \quad (15.4)$$

În urma efectuării notațiilor (14), din relațiile (13) și (15) rezultă sistemul de ecuații:

$$\left\{ \begin{array}{l} \frac{\tilde{\mu}}{\tilde{\mu} + \tilde{\lambda} + \lambda\tilde{\mu}\tilde{\mu} + \lambda\tilde{\lambda}\tilde{\mu}} = \frac{\mu^3}{(\lambda + \mu)^3} \\ \frac{\lambda\tilde{\mu}\tilde{\mu}}{\tilde{\mu} + \tilde{\lambda} + \lambda\tilde{\mu}\tilde{\mu} + \lambda\tilde{\lambda}\tilde{\mu}} = \frac{3\lambda\mu^2}{(\lambda + \mu)^3} \\ \frac{\lambda\tilde{\lambda}\tilde{\mu}}{\tilde{\mu} + \tilde{\lambda} + \lambda\tilde{\mu}\tilde{\mu} + \lambda\tilde{\lambda}\tilde{\mu}} = \frac{3\lambda^2\mu}{(\lambda + \mu)^3} \\ \frac{\tilde{\lambda}}{\tilde{\mu} + \tilde{\lambda} + \lambda\tilde{\mu}\tilde{\mu} + \lambda\tilde{\lambda}\tilde{\mu}} = \frac{\lambda^3}{(\lambda + \mu)^3} \end{array} \right. \quad (16)$$

Efectuând calculele adecvate se ajunge la sistemul:



$$\begin{cases} \frac{\tilde{\lambda}}{\tilde{\mu}} = \left(\frac{\lambda}{\mu}\right)^3 \\ \frac{\tilde{\lambda}\tilde{\mu}\tilde{\mu}}{\tilde{\mu}} = \frac{3\lambda}{\mu} \\ \frac{\tilde{\lambda}\tilde{\lambda}\tilde{\mu}}{\tilde{\lambda}} = \frac{3\mu}{\lambda} \end{cases} \quad (17)$$

Folosind relațiile (17), detaliate din (14), în final se obține:

$$\begin{cases} \frac{\lambda_{01}}{\mu_{10}} = \frac{3\lambda}{\mu} \\ \frac{\lambda_{12}}{\mu_{21}} = \frac{\lambda}{\mu} \\ \frac{\lambda_{23}}{\mu_{32}} = \frac{\lambda}{3\mu} \end{cases} \quad (18)$$

Datorită faptului că există trei ecuații cu șase necunoscute se determină experimental după metoda din [35], particularizată pentru o structură cu 3 entități reparabile, următoarele intensități de reparare:

- $$\mu_{32} = \frac{1}{MDT_3} \quad (19)$$

unde  $MDT_3$  este timpul mediu de defectare corespunzător cazului în care toate cele trei entități ale sistemului redundant sunt defecte și poate fi determinat cu relația:

$$MDT_3 = \frac{1}{N_{DT_3}} \sum_{j=1}^n T_{DT_3,j} \quad (20)$$

Timpii  $T_{DT_3,j}$  se consideră corespunzător acelor intervale de timp în care toate cele trei entități sunt defecte.

- $$\mu_{21} = \frac{1}{MDT_2} \quad (21)$$

unde  $MDT_2$  este timpul mediu de defectare corespunzător cazului în care două entități ale sistemului redundant sunt defecte și urmează a se repara una din cele două fără să se defecteze ce-a de-a treia, respectiv poate fi determinat cu relația:

$$MDT_2 = \frac{1}{N_{DT_2}} \sum_{j=1}^n T_{DT_2,j} \quad (22)$$

Timpii  $T_{DT_2,j}$  se consideră corespunzător acelor intervale de timp în care două entități din trei sunt defecte.

$$\bullet \quad \mu_{10} = \frac{1}{MDT_1} \quad (23)$$

unde  $MDT_1$  este timpul mediu în care una dintre entități este defectă și urmează să fie reparată fără a se defecta pe parcurs nici una din cele 2 entități și poate fi determinat utilizând relația

$$MDT_1 = \frac{1}{N_{DT_1}} \sum_{j=1}^n T_{DT_1j} \quad (24)$$

unde timpii  $T_{DT_1j}$  se consideră corespunzător acelor intervale de timp în care una dintre entități este defectă urmată de trecerea în starea în care toate trei entitățile sunt reparate.

În cazul în care s-au determinat intensitățile de reparare prin relațiile (19)-(24), valorile intensităților de defectare se pot calcula utilizând relațiile:

$$\begin{cases} \lambda_{23} = \frac{\lambda}{3\mu} \mu_{32} \\ \lambda_{12} = \frac{\lambda}{\mu} \mu_{21} \\ \lambda_{01} = \frac{3\lambda}{\mu} \mu_{10} \end{cases} \quad (25)$$

Studiul se continuă cu partea aplicativă bazată pe analize de disponibilități în cadrul subcapitolului 3.2.

### 2.3.3 Calculul ratei de reparare a unui ansamblu serie

În fiabilitate, structura serie este una cu un risc mare de defectare, respectiv cu o siguranță scăzută în funcționare. O structură serie neprevăzută cu subansambluri redundante trebuie cel puțin să fie constituită din componente cu un grad mare de fiabilitate pentru a prezenta un anumit grad de siguranță. Specificul structurii serie este acela că orice defectare a unei componente implică defectarea întregului sistem.

În literatura de specialitate (Ex. [47], [73]), pentru rata de defectare a unui ansamblu cu  $n$  componente, cu ratele de defectare  $\lambda_i$ , conectate în serie, se demonstrează formula:

$$\lambda_r = \sum_{i=1}^n \lambda_i \quad (26)$$

Din punctul de vedere al estimării disponibilității este importantă și stabilirea unei formule de calcul a ratei de reparare  $\mu_r$  a unei structuri de  $n$  componente conectate într-o schemă de fiabilitate de tip serie în funcție de intensitățile de defectare și de reparare a elementelor din ansamblu. Pentru aceasta s-a folosit modelarea Markov.

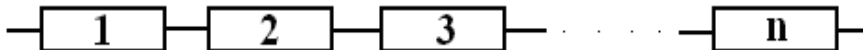


Fig. 2.3.4 Exemplu de structură serie cu  $n$  componente

Sistemului ilustrat în figura 2.3.4, în care  $\textcircled{i}$  corespunde stării în care i din cele n elemente ale structurii sunt în fază de reparare, îi corespunde modelul următor:

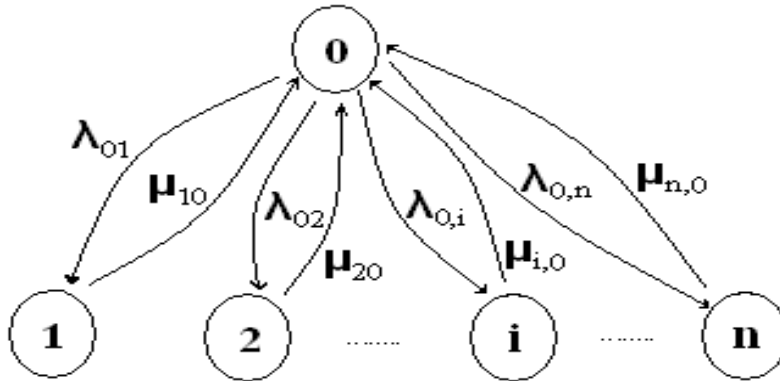


Fig. 2.3.5 Modelul Markov asociat schemei serie

Procesele tranzitorii asociate probabilităților celor n+1 stări ( $\textcircled{0}$ ,  $\textcircled{1}$ , ...,  $\textcircled{n}$ ) prezentate în fig. 2.3.5 pot fi descrise de ecuațiile următoare care țin cont de intensitățile de defectare și reparare:

$$\begin{cases} P_0(t+\Delta t) = P_0(t) \sum_{i=1}^n (1-\lambda_{0,i}\Delta t) + \sum_{i=1}^n P_i(t) \mu_{i,0}\Delta t \\ P_i(t+\Delta t) = P_0(t) \lambda_{0,i}\Delta t + P_i(t) (1-\mu_{i,0}\Delta t), \quad i=1;n \end{cases} \quad (27)$$

Prin trecere la limită pentru  $\Delta t \rightarrow 0$  obținem:

$$\begin{cases} \dot{P}_0(t) = -P_0(t) \sum_{i=1}^n \lambda_{0,i} + \sum_{i=1}^n P_i(t) \mu_{i,0} \\ \dot{P}_i(t) = \lambda_{0,i} P_0(t) - \mu_{i,0} P_i(t), \quad i=1;n \end{cases} \quad (28)$$

Se poate observa că suma relațiilor (28) este:  $\dot{P}_0(t) + \dot{P}_1(t) + \dot{P}_2(t) + \dots + \dot{P}_n(t) = 0$ . Deoarece sistemul se află cu certitudine într-una din cele n+1 stări rezultă:

$$P_0(t) + P_1(t) + P_2(t) + \dots + P_n(t) = 1. \quad (29)$$

În regim staționar relațiile (28) și (29) conduc la sistemul:

$$\begin{cases} -P_{0,\infty} \sum_{i=1}^n \lambda_{0,i} + \sum_{i=1}^n P_{i,\infty} \mu_{i,0} = 0 \\ \lambda_{0,i} P_{0,\infty} - \mu_{i,0} P_{i,\infty} = 0, \quad i=1;n \\ P_{0,\infty} + P_{1,\infty} + P_{2,\infty} + \dots + P_{n,\infty} = 1 \end{cases} \quad (30)$$

Soluțiile sistemului de ecuații (30) sunt:

$$\begin{cases} P_{0,\infty} = \frac{1}{1 + \sum_{i=1}^n \frac{\lambda_{0,i}}{\mu_{i,0}}} \\ P_{i,\infty} = \frac{\lambda_{0,i}}{\mu_{i,0}} P_{0,\infty}, \quad i = 1; n \end{cases} \quad (31)$$

Sistemul din figura 2.3.4 este în stare de funcționare când toate componentele sunt în stare de funcționare, iar starea de defectare înseamnă defectarea oricăreia dintre cele  $n$  elemente ale ansamblului serie. Ca urmare, disponibilitatea ( $A_s$ ) sistemului este:

$$A_s = P_{0,\infty} = \frac{1}{1 + \sum_{i=1}^n \frac{\lambda_{0,i}}{\mu_{i,0}}}, \quad (32)$$

$$\text{iar non-disponibilitatea } U = \sum_{i=1}^n P_{i,\infty} = P_{0,\infty} \sum_{i=1}^n \frac{\lambda_{0,i}}{\mu_{i,0}}. \quad (32')$$

În același timp, disponibilitatea ansamblului serie este [47], [73], [49]:

$$A_s = \frac{\mu_r}{\mu_r + \lambda_r} \quad (33)$$

Astfel, din relațiile (32) și (33) rezultă egalitatea:

$$\frac{1}{1 + \sum_{i=1}^n \frac{\lambda_{0,i}}{\mu_{i,0}}} = \frac{\mu_r}{\mu_r + \lambda_r} \quad (34)$$

iar în urma efectuării de calcule rezultă:

$$\mu_r = \frac{\lambda_r \prod_{i=1}^n \mu_{i,0}}{\sum_{i=1}^n \prod_{j=1, j \neq i}^n \lambda_{0,i} \mu_{j,0}} \quad (35)$$

Împărțim cu  $\prod_{i=1}^n \mu_{i,0}$  și utilizând (26), relația (35) devine:

$$\mu_r = \frac{\sum_{i=1}^n \lambda_{0,i}}{\sum_{i=1}^n \frac{\lambda_{0,i}}{\mu_{i,0}}} \quad (36)$$

Relația (36) reprezintă expresia ratei de reparare a unui ansamblu de  $n$  componente în serie.

## 2.4 Tranziții simultane în cazul structurilor reparabile cu redundanță activă

Multitudinea situațiilor reale face ca în literatura de specialitate să figureze analize de disponibilitate pentru un număr redus de situații [47], [58], [73], [14], [49], [90]. Bunăoară, nu se prezintă situațiile în care pot să apară tranziții simultane a două sau mai multe componente ale unui sistem din starea de defectare în starea de reparare și invers. Admiterea acestei ipoteze implică posibilitatea considerării unor tranziții complexe, diferite de cele analizate.

Dacă se consideră cazul unui sistem redundant de ordin  $k$  ( $k$  entități reparabile), ilustrat în fig. 2.3.2 din capitolul 2.3, ratele de tranziție dintr-o stare în alta se prezintă în felul următor:

- $\lambda_{i,i+1}$ ,  $i=0, 1, 2, \dots, k-1$ , ratele de defectare corespunzătoare tranzițiilor din starea  $\textcircled{i}$  în starea  $\textcircled{i+1}$
- $\mu_{i+1,i}$  ratele de reparare corespunzătoare tranzițiilor din starea  $\textcircled{i+1}$  în starea  $\textcircled{i}$ .

Dacă se admite însă faptul că pot să apară la un moment dat tranziții simultane a două sau mai multe componente din starea de funcționare în starea de defectare și invers, situația se poate schimba considerabil. Figura 2.4.1 surprinde această situație:

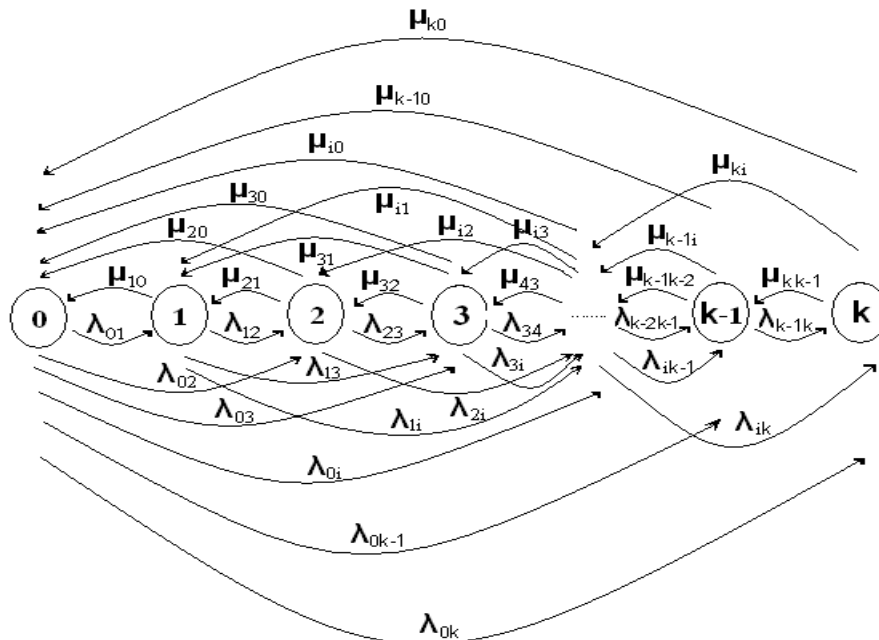


Fig. 2.4.1 Modelul Markov al structurii redundante admitând tranziții simultane

Astfel, în afara ratelor definite mai sus, trebuie avute în vedere și următoarele intensități de defectare, respectiv reparare corespunzătoare tranziției mai multor componente dintr-o stare în alta:

- $\lambda_{i,j}$ ,  $i=0;k-2, j=2;k$ , unde  $j \geq i+2$ , reprezintă ratele de defectare simultane corespunzătoare tranzițiilor sistemului redundant din starea  $\textcircled{i}$  în starea  $\textcircled{j}$
- $\mu_{i,j}$ ,  $i=k;2, j=k-2;0$ , unde  $j \leq i-2$ , reprezintă ratele de reparare simultane corespunzătoare tranzițiilor sistemului redundant din starea  $\textcircled{i}$  în starea  $\textcircled{j}$

Studiul disponibilității trebuie să țină cont și de tranzițiile simultane între stările unui sistem. Obiectivul acestui subcapitol îl reprezintă furnizarea de relații de interdependență între ratele de defectare și de reparare considerate, oferind posibilitatea calculului unor rate în ipoteza altora cunoscute. Modalitatea de lucru este ilustrată printr-un studiu efectuat asupra unei structuri cu două entități reparabile.

Situația unei structuri redundante cu două entități reparabile în cadrul căreia se admit tranziții simultane ale componentelor dintr-o stare în alta a fost studiată de autor, în colaborare, în lucrarea [34]. Paragraful curent prezintă punctul de plecare al studiului și ilustrează rezultatul obținut, fără a mai detalia cercetarea propriu-zisă.

Modelul este mai complex decât cel analizat în fig. 2.3.3 din paragraful 2.3.2, în sensul că apare posibilitatea de defectare sau de reparare a două componente simultan. În cazul unui sistem redundant cu două entități reparabile, cele trei stări posibile ale sistemului sunt identice cu cele prezentate în paragraful anterior, diferența o reprezintă ipoteza admiterii unor tranziții de pas dublu, de trecere din starea  $\textcircled{0}$  în starea  $\textcircled{2}$  și invers, de trecere din starea  $\textcircled{2}$  în starea  $\textcircled{0}$ . Fie  $\lambda_{02}$  și  $\mu_{20}$  ratele de defectare, respectiv reparare simultană asociate celor două tipuri de tranziții, respectiv probabilitățile de tranziție  $P_0(t)\lambda_{02}\Delta t$  și  $P_2(t)\mu_{20}\Delta t$  ale sistemului din starea  $\textcircled{0}$  în  $\textcircled{2}$  și din starea  $\textcircled{2}$  în starea  $\textcircled{0}$ . Structura este ilustrată în fig. 2.4.2 :

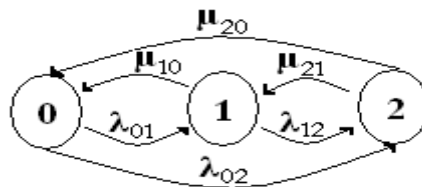


Fig. 2.4.2 Structura redundanță de ordin 2 în care se admit tranziții simultane

Punctul de plecare al modelării este reprezentat de descrierea proceselor tranzitorii asociate probabilităților celor trei stări:

$$\begin{aligned}
 P_0(t + \Delta t) &= P_0(t) (1 - \lambda_{01}\Delta t) (1 - \lambda_{02}\Delta t) + P_1(t)\mu_{10}\Delta t + P_2(t)\mu_{20}\Delta t \\
 P_1(t + \Delta t) &= P_0(t)\lambda_{01}\Delta t + P_1(t) (1 - \mu_{10}\Delta t) (1 - \lambda_{12}\Delta t) + P_2(t)\mu_{21}\Delta t \\
 P_2(t + \Delta t) &= P_0(t)\lambda_{02}\Delta t + P_1(t) \lambda_{12}\Delta t + (1 - \mu_{20}\Delta t)(1 - \mu_{21}\Delta t) P_2(t)
 \end{aligned}
 \tag{37}$$

Dinamica tranzițiilor, utilizată apoi pentru implementare în vederea unui studiu de disponibilitate realizat în cadrul subcapitolului 3.3, este descrisă prin ecuațiile:

$$\begin{cases} \dot{P}_0(t) = (-\lambda_{01} - \lambda_{02})P_0(t) + \mu_{10}P_1(t) + \mu_{20}P_2(t) \\ \dot{P}_1(t) = \lambda_{01}P_0(t) - (\lambda_{12} + \mu_{10})P_1(t) + \mu_{21}P_2(t) \\ \dot{P}_2(t) = \lambda_{02}P_0(t) + \lambda_{12}P_1(t) - (\mu_{21} + \mu_{20})P_2(t) \end{cases} \quad (38)$$

Rezultatul studiului este reprezentat de sistemul (39), care furnizează interdependența dintre ratele de tranziție simultane, ratele de tranziție normale între stările sistemului, respectiv ratele de tranziție individuale ale entităților:

$$\begin{cases} \frac{\lambda_{01}\mu_{21} + \lambda_{02}\mu_{21} + \lambda_{01}\mu_{20}}{\lambda_{12}\lambda_{01} + \lambda_{12}\lambda_{02} + \lambda_{02}\mu_{10}} = \frac{2\lambda}{\mu} \\ \frac{\mu_{10}\mu_{21} + \mu_{10}\mu_{20} + \lambda_{12}\mu_{20}}{\lambda_{01}\mu_{21} + \lambda_{02}\mu_{21} + \lambda_{01}\mu_{20}} = \frac{\lambda}{2\mu} \end{cases} \quad (39)$$

Sistemul (39) prezintă două ecuații și șase necunoscute. În concluzie, din cele șase necunoscute, patru se determină experimental și anume ratele de reparare, precum și rata de defectare simultană ( $\lambda_{02}$ ,  $\mu_{10}$ ,  $\mu_{20}$ ,  $\mu_{21}$ ).

### 3. Studiul unor aspecte referitoare la reglarea și disponibilitatea unei centrale electrice geotermale

După cum s-a precizat în subcapitolul 1.7, unul dintre sistemele care a furnizat posibilitatea aplicării studiilor teoretice întreprinse l-a reprezentat centrala electrică geotermală din cadrul Universității Oradea. Autorul tezei a făcut parte din colectivul care s-a preocupat de evaluarea disponibilității diferitelor subsisteme ale centralei și de imaginarea unor soluții de creșterea a disponibilității. Principalele rezultate au fost valorificate prin lucrările [34], [35], [45], [65], [67].

În acest context, în cadrul capitolului se prezintă mai întâi (subcapitolul 3.1) elemente legate de centrala electrică geotermală, urmate, în subcapitolul 3.2, de implementarea dezvoltărilor teoretice referitoare la evaluarea disponibilității, în ipoteza considerării tipurilor de tranziții expuse în cadrul subcapitolului 2.3. În subcapitolul 3.3 se prezintă un studiu de caz în care se aplică cele descrise în cadrul subcapitolului 2.4. În următorul subcapitol, 3.4, se prezintă elemente legate de regulatoarele interpolative, pentru ca ulterior, în 3.5 și 3.6, să se descrie o soluție de reglare a temperaturii bazată pe interpolare, soluție prin care se asigură creșterea disponibilității în cadrul blocului vaporizatoare. Ultimul subcapitol, 3.7, prezintă un modul mobil de detecție și diagnoză a defectelor pentru sistemele automate cu aplicarea asupra centralei electrice geotermale bazat pe identificarea unor corelații caracteristice diferitelor tipuri de defecțiuni.

#### 3.1 Centrala electrică geotermală

Centrala electrică geotermală realizează conversia energiei termice a apei geotermale în energie electrică utilizând bioxidul de carbon ca agent de lucru. În figura 3.1.1 apare sub formă de machetă o imagine generală a centralei [2].

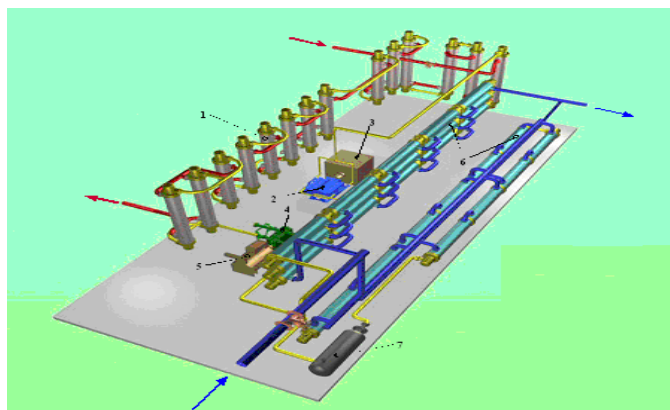


Fig. 3.1.1 Privire generală asupra centralei electrice geotermale



Părțile componente principale ale centralei electrice geotermale sunt: blocul vaporizatoare, grupul turbină-generator, blocul condensatoare, vasul tampon și pompa de CO<sub>2</sub> lichid (vezi fig. 3.1.2). Întreaga evoluție este concepută pentru a realiza transferul de căldură în cadrul unui ciclu termodinamic dat. Liniile întrerupte din stânga și dreapta figurii corespund fluxurilor de apă caldă, respectiv rece din sistem. Linia îngroșată reprezintă circuitul de CO<sub>2</sub> din cadrul centralei electrice geotermale. În blocul vaporizatoare, CO<sub>2</sub> preia energia de la apa caldă și o cedează în grupul turbină-generator pentru a produce energie electrică și în blocul condensatoare pentru a obține parametrii termodinamici doriți pentru reluarea ciclului.

În fig. 3.1.2 și dezvoltările ulterioare se utilizează următoarele notații (parametrii echipamentelor se găsesc în [79]):

RB<sub>1</sub> – valva de acces (robinet) a apei calde în vaporizatoare; prin intermediul ei se controlează debitul de apă caldă  $q_{ac}$ ;  $\theta_{ac}$  - temperatura apei calde la ieșirea din vaporizatoare;

RB<sub>3</sub> – valva de acces (robinet) a apei reci în condensatoare; servește pentru controlul debitului de apă rece  $q_{ar}$ ;  $\theta_{ar}$  - temperatura apei reci la ieșirea din condensatoare;

$\theta_1$  - temperatura CO<sub>2</sub> la ieșirea din vaporizatoare; TT<sub>1</sub> – senzorul care măsoară  $\theta_1$  și furnizează buclei de reglare a temperaturii  $\theta_1$  temperatura de pe reacție  $\theta_{1r}$ ;

$\theta_3$  - temperatura CO<sub>2</sub> la ieșirea din condensatoare; TT<sub>3</sub> – senzorul care măsoară  $\theta_3$  și furnizează buclei de reglare a temperaturii  $\theta_3$  temperatura de pe reacție  $\theta_{3r}$ ;

$h_1, h_3$  – deschiderile valvelor RB<sub>1</sub> și RB<sub>3</sub>;

$u_1, u_3$  – semnalele de comandă pentru motoarele valvelor RB<sub>1</sub>, respectiv RB<sub>3</sub>, furnizate de reglatoarele din cadrul buclelor de reglare ale blocurilor vaporizatoare, respectiv condensatoare;

$\theta_2$  - temperatura CO<sub>2</sub> la intrarea în blocul condensatoare;

$q_{CO_2, x-y}$  - debitul de CO<sub>2</sub> între două subsamblă consecutive x și y;

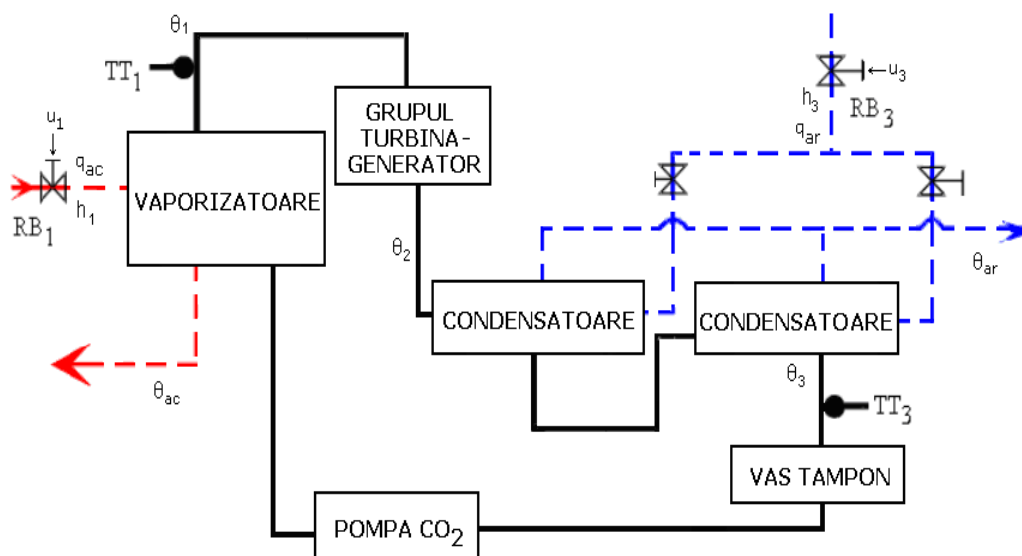


Fig. 3.1.2 Structura centralei electrice geotermale

Pentru a asigura ciclul termodinamic al CO<sub>2</sub>, respectiv pentru a obține valorile nominale  $\theta_{1nom}$  și  $\theta_{3nom}$ ,  $\theta_1$  și  $\theta_3$  sunt reglate în bucle închise separate. Întregul proces este subordonat unei comenzi de nivel superior al CO<sub>2</sub> realizate de pompă.

### 3.2 Implementarea pe un caz real a dezvoltărilor teoretice referitoare la structurile cu redundanță activă cu k componente reparabile identice

În cadrul acestui paragraf, studiul teoretic descris în cadrul subcapitolului 2.3 se aplică practic în cazul real al sistemului de control al ciclului termodinamic al centralei electrice geotermale prezentată în subcapitolul 3.1.

Pentru studierea sistemului de control al ciclului termodinamic, din punctul de vedere al fiabilității, respectiv disponibilității, într-o primă etapă, s-au realizat modelele de fiabilitate/disponibilitate corespunzătoare. Astfel, în figura 3.2.1 este ilustrată, într-o primă etapă, schema de fiabilitate a sistemului la nivel de blocuri componente (subsisteme). Subsistemele diferitelor bucle de reglare sunt conduse de un automat programabil (subsistemul de conducere).

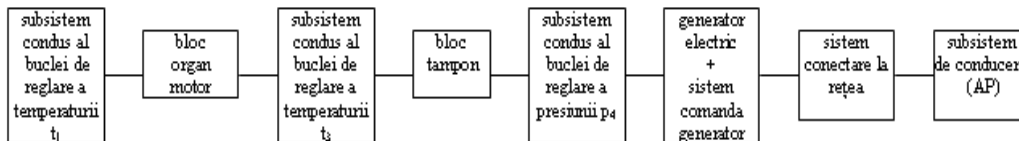


Fig. 3.2.1 Schema de analiză a fiabilității sistemului de control al ciclului termodinamic

Într-o a doua etapă, s-a obținut schema de fiabilitate din figura 3.2.2 a subsistemului condus al buclei de reglare a temperaturii  $t_1$ .

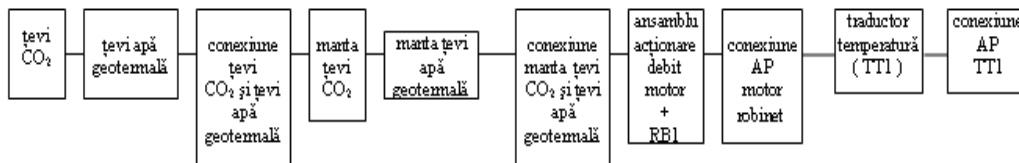


Fig. 3.2.2 Schema de fiabilitate a subsistemului condus al buclei de reglare a temperaturii  $t_1$

Într-o a treia etapă, în cadrul buclei de reglare a temperaturii au fost identificate, din considerente funcționale, două puncte critice: ansamblul de acționare debit motor+RB<sub>1</sub> și subsistemul traductor de temperatură TT<sub>1</sub>. În vederea asigurării unei fiabilități/disponibilități mai ridicate a subsistemului prezentat, în lucrarea [35] s-a propus, ca soluție, schema cu componentele redundante din figura 3.2.3, în care se folosește pentru primul punct critic o structură dublu redundantă paralel, iar pentru al doilea punct critic o structură triplu redundantă paralel.

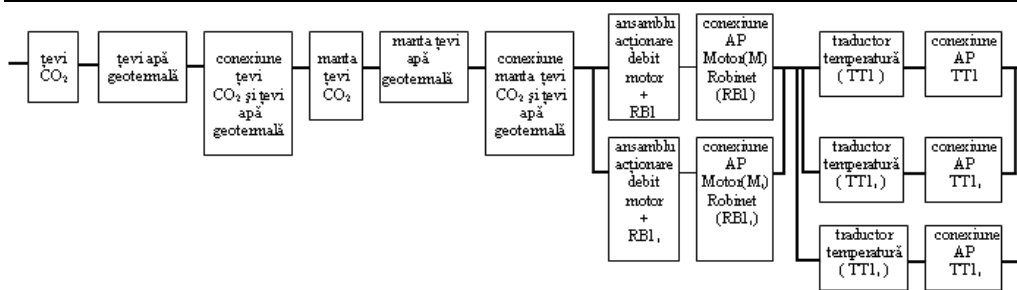


Fig. 3.2.3 Schema de fiabilitate cu redundanță a subsistemului condus al buclei de reglare a temperaturii  $t_1$

Pentru a compara disponibilitatea structurii inițiale cu disponibilitatea structurii propuse, adică a schemei din figura 3.2.2, cu cea a schemei din figura 3.2.3, se folosește, potrivit subcapitolului 2.3, modelarea Markov. Intensitățile de defectare și de reparare ale tuturor componentelor precum și intensitățile de reparare determinate experimental pentru structurile care prezintă redundanță au valorile din Tabelul 3.1. În cadrul tabelului cu ajutorul indicelui  $n$  se indică tranzițiile din starea  $n+1$  în starea  $n$  (indicii ratelor de reparare iau valori de la 0 la 2), așa cum este ilustrat în cadrul figurii 2.3.2 din paragraful 2.3.1, iar cu ajutorul indicelui  $k$  entitatea.

Tabelul 3.1		$\lambda_k$	$\mu_k$	$\mu_{n+1n-k}$
K	Entitate	(ore <sup>-1</sup> )	(ore <sup>-1</sup> )	(ore <sup>-1</sup> )
1	țevi CO <sub>2</sub>	0.0002283	0.0415	
2	țevi apă geotermală	0.000225	0.0416	
3	conexiune țevi CO <sub>2</sub> - apă geotermală	0.000215	0.05	
4	manta țevi CO <sub>2</sub>	0.000182	0.0415	
5	manta țevi apă geotermală	0.00018	0.0416	
6	conexiune manta țevi CO <sub>2</sub> - apă geotermală	0.000175	0.05	
7	RB <sub>1</sub>	0.0001543	0.05	$\mu_{10-7} = 0.0470,$ $\mu_{21-7} = 0.04715,$
	motor RB <sub>1</sub>	0.000057	0.0416	
	conexiune AP - motor RB <sub>1</sub>	0.00021	0.05	
8	traductor de temperatură TT <sub>1</sub>	0.00012	0.05	$\mu_{10-8} = 0.0454,$ $\mu_{21-8} = 0.0455,$ $\mu_{32-8} = 0.0456$
	Conexiune AP - TT <sub>1</sub>	0.000156	0.0416	

După cum se observă în figura 3.2.3 structurile cu redundanță triplă și cu redundanță dublă conțin componente conectate într-o rețea serie (Ex. Traductorul de temperatură TT<sub>1</sub> și conexiunea PLC-TT<sub>1</sub>). Pentru a putea realiza simularea (în vederea obținerii evoluției disponibilității sistemului prevăzut cu structuri redundante), respectiv calculul disponibilității sistemului cu structuri redundante este necesar să se calculeze ratele de defectare și de reparare ale grupurilor de componente conectate în serie. Astfel, trebuie să se calculeze  $\mu_7$  și  $\lambda_7$ , respectiv  $\mu_8$  și  $\lambda_8$ , cunoscându-se ratele de defectare și de reparare ale fiecărui element din

conexiunile serie,  $\mu_{7i}$  și  $\lambda_{7i}$ ,  $\mu_{8j}$  și  $\lambda_{8j}$ , unde  $i=1;3$ ,  $j=1;2$ . Folosind formulele (26) și (36) obținute în paragraful 2.3.3 se obțin relațiile:

$$\lambda_7 = \lambda_{71} + \lambda_{72} + \lambda_{73}, \quad \lambda_8 = \lambda_{81} + \lambda_{82} \quad (1)$$

$$\frac{\lambda_{71} + \lambda_{72} + \lambda_{73}}{\mu_7} = \frac{\lambda_{71}}{\mu_{71}} + \frac{\lambda_{72}}{\mu_{72}} + \frac{\lambda_{73}}{\mu_{73}}, \quad \frac{\lambda_{81} + \lambda_{82}}{\mu_8} = \frac{\lambda_{81}}{\mu_{81}} + \frac{\lambda_{82}}{\mu_{82}} \quad (2)$$

Urmează determinarea ratelor de defectare  $\lambda_{01-7}$ ,  $\lambda_{12-7}$ , respectiv  $\lambda_{01-8}$ ,  $\lambda_{12-8}$ ,  $\lambda_{23-8}$ . Astfel, subsistemul 7 din cadrul tabelului 3.1 prezintă o redundanță dublă, iar pentru a calcula ratele de defectare aferente se aplică relațiile (3) din cadrul paragrafului 2.3.1 după cum urmează:

$$\lambda_{12-7} = \frac{\lambda_7}{2\mu_7} \mu_{21-7}, \quad \lambda_{01-7} = \frac{2\lambda_7}{\mu_7} \mu_{10-7} \quad (3)$$

În cazul structurii triplu redundante (structura 8 din tabelul 3.1), pentru determinarea ratelor de defectare  $\lambda_{01-8}$ ,  $\lambda_{12-8}$ ,  $\lambda_{23-8}$  se utilizează relația (25) din cadrul paragrafului 2.3.2. Astfel:

$$\lambda_{23-8} = \frac{\lambda_8}{3\mu_8} \mu_{32-8}, \quad \lambda_{12-8} = \frac{\lambda_8}{\mu_8} \mu_{21-8}, \quad \lambda_{01-8} = \frac{3\lambda_8}{\mu_8} \mu_{10-8} \quad (4)$$

Se fac următoarele notații:  $A_i$ ,  $i = 1, 2, \dots, 6$ , - disponibilitățile elementelor 1, ..., 6, menționate în tabelul 3.1, diponibilitatea ansamblului 7 fără redundanță -  $A_7$ , respectiv cu redundanță dublă -  $A_7'$ , iar disponibilitatea ansamblului 8 fără redundanță -  $A_8$ , respectiv prezentând redundanță triplă -  $A_8'$ .

Studiul prin modele Markov înseamnă în esență, așa cum s-a prezentat în cadrul subcapitolului 1.2, calculul probabilităților ca sistemul să se găsească într-o anumită stare, ținând seama de tranzițiile din sistem dintr-o stare în alta. Aceste probabilități pot fi calculate cu ajutorul ratelor de defectare și de reparare din tabelul 3.1, respectiv cu ajutorul relațiilor (1), (2), (3), (4). Disponibilitatea unei structuri seriale rezultă în final însumând probabilitățile stărilor de succes ale sistemului.

În regim staționar, calculul disponibilității se poate realiza în cazul de față algebric. Disponibilitatea este însă o funcție de timp care, chiar pentru structurile simple, nu mai poate fi determinată analitic ci doar numeric, prin simulare. În acest context, în continuare, se implementează structura prevăzută cu redundanțe, respectiv fără redundanțe în vederea simulării. Implementarea se va realiza în conformitate cu dinamica tranzițiilor (ex. pentru subansamblul triplu redundant se vor utiliza formulele (7) din paragraful 2.3.2). Simulările au ca scop prezentarea comparativă a evoluției disponibilităților pentru cele două situații. În cadrul implementării se face uz de formulele (3) și (4).

În fig. 3.2.4 se prezintă schema de implementare (modelul Simulink) a sistemului prevăzut cu redundanța dublă și triplă în punctele critice (se notează cu  $A_r$  disponibilitatea sistemului). De asemenea în fig. 3.2.5 este ilustrată schema de implementare (modelul Simulink) a sistemului fără redundanțe (se notează cu  $A_{non-r}$  disponibilitatea sistemului).

Pentru a înțelege mai bine maniera în care s-a realizat implementarea, în fig. 3.2.4, în cadrul schemei care ilustrează subansamblul triplu redundant, se vor face următoarele notații referitoare la formulele (7) din paragraful 2.3.2:  $P_i'$  - probabilitatea subansamblului de a se afla în starea  $i$ ,  $i=0;3$ . Se menționează că starea inițială a subansamblului este starea  $\textcircled{0}$  în care toate entitățile sunt funcționale, iar disponibilitatea subansamblului triplu redundant ( $A_8'$ ) se calculează

însușind probabilitățile  $P_0'$ ,  $P_1'$  și  $P_2'$ , corespunzătoare stărilor de succes ale sistemului.

Analog, în figura 3.2.5, se notează probabilitățile  $P_j$ ,  $j=0;1$ , corespunzătoare aceluiași subsansamblu care de această dată nu este prevăzut cu redundanță.

În urma efectuării simulărilor s-au obținut câteva rezultate semnificative, prezentate în continuare.

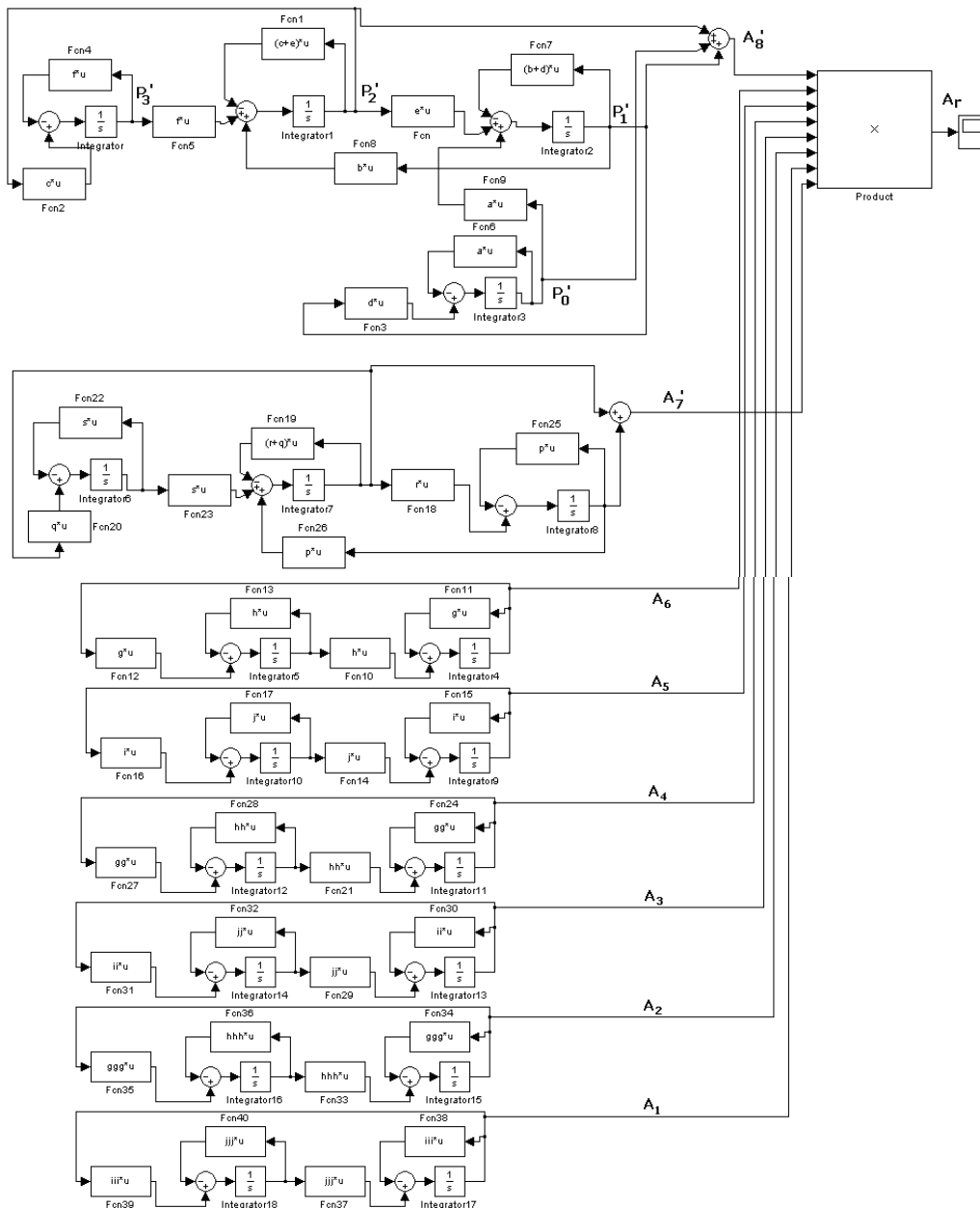


Fig. 3.2.4 Schema de implementare a sistemului prevăzut cu redundanțe duble și triple

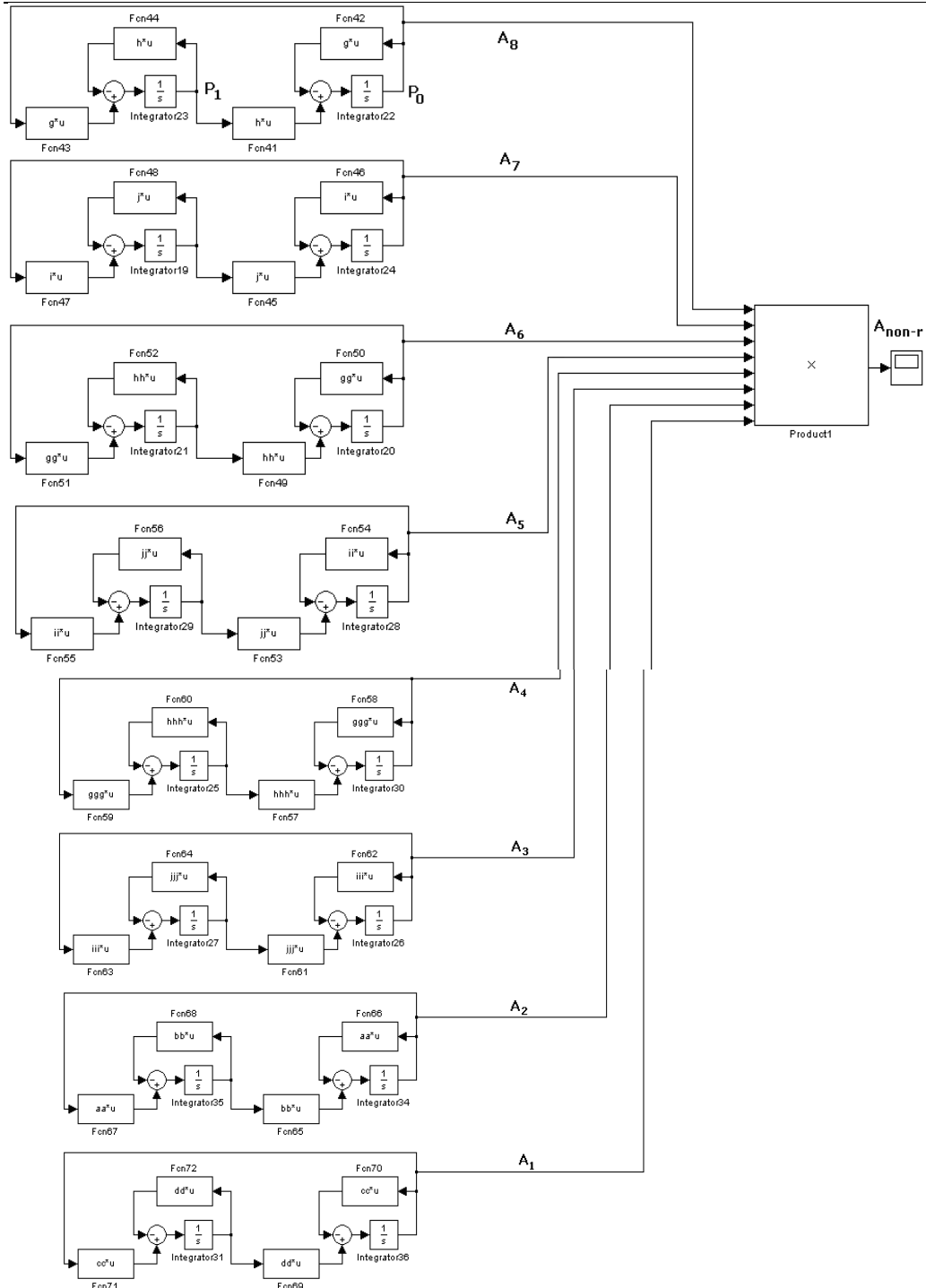


Fig. 3.2.5 Schema de implementare a sistemului fără redundanțe

Într-o primă situație (primul scenariu), s-a considerat primul punct critic, în care se aplică redundanța triplă (ansamblul traductor de temperatură  $TT_1$  și conexiunea PLC- $TT_1$ ). Următoarele două figuri (3.2.6 și 3.2.7) ilustrează comparativ evoluția disponibilității structurii în cazul în care se aplică redundanța triplă și în situația în care nu există redundanță (este vorba despre disponibilitățile  $A_8'$  și  $A_8$  ilustrate în cadrul figurilor 3.2.4 și 3.2.5).

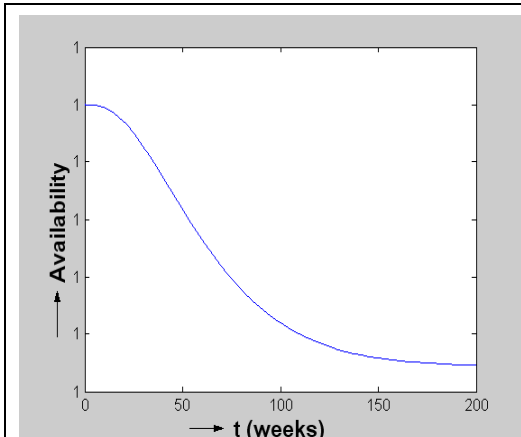


Fig. 3.2.6 Disponibilitatea subansamblului cu redundanță triplă ( $A_8'$ )

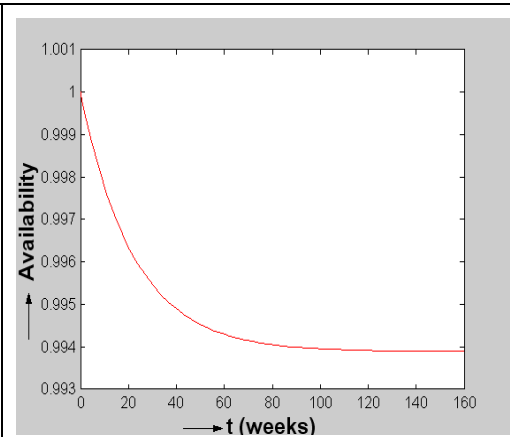


Fig. 3.2.7 Disponibilitatea subansamblului neredundant ( $A_8$ )

Urmărind aceeași situație, este furnizată și non-disponibilitatea în cele două situații (fig. 3.2.8, fig. 3.2.9). Este vorba despre mărimile  $P_3'$  și  $P_1$  reprezentate în figurile 3.2.4, respectiv 3.2.5.

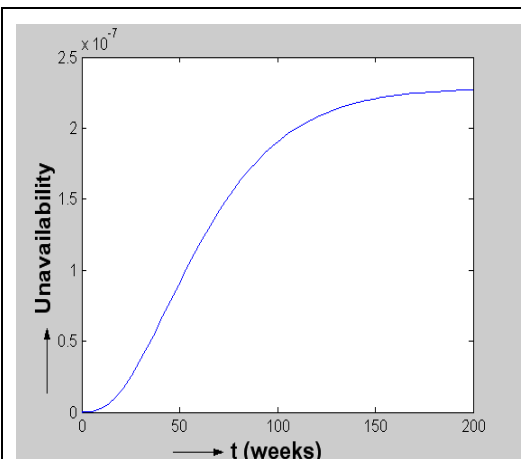


Fig. 3.2.8 Non-disponibilitatea subansamblului cu redundanță triplă

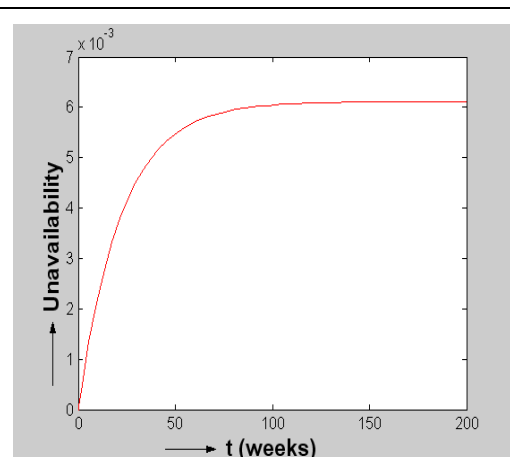
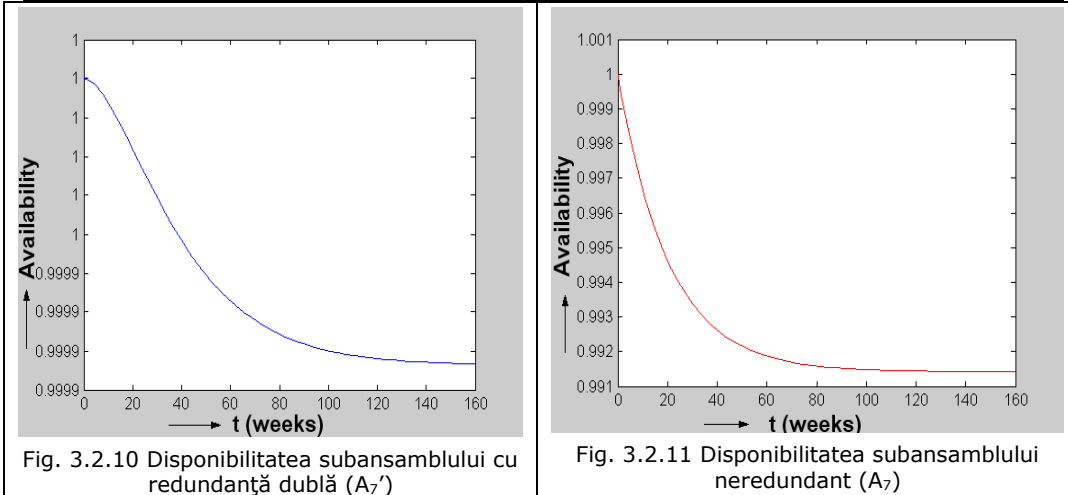


Fig. 3.2.9 Non-disponibilitatea subansamblului neredundant

A doua situație (al doilea scenariu de simulare) aduce în prim plan al doilea punct critic. Următoarele două figuri prezintă disponibilitatea pentru al doilea punct critic în cazul în care acesta este prevăzut cu structura redundanță dublă ( $A_7'$  în figura 3.2.5), precum și situația în care ansamblul nu prezintă redundanță ( $A_7$  în figura 3.2.4).



Ultimul scenariu de simulare realizează prezentarea comparativă a evoluției disponibilității întregului sistem în două situații:

- sistemul este prevăzut cu structuri redundante (schema de simulare 3.2.4),
- sistemul nu este prevăzut cu structuri redundante (schema de simulare 3.2.5).

S-a avut în vedere că disponibilitatea sistemului fără redundanțe din figura 3.2.2 este:

$$A_{\text{non-r}}(t) = A_1(t) \cdot A_2(t) \cdot A_3(t) \cdot A_4(t) \cdot A_5(t) \cdot A_6(t) \cdot A_7(t) \cdot A_8(t), \quad (5)$$

iar disponibilitatea sistemului care prezintă redundanțe din figura 3.2.3 este:

$$A_r(t) = A_1(t) \cdot A_2(t) \cdot A_3(t) \cdot A_4(t) \cdot A_5(t) \cdot A_6(t) \cdot A_7'(t) \cdot A_8'(t). \quad (6)$$

Cu scopul comparării rezultatelor obținute, evoluția disponibilităților ( $A_r$  și  $A_{\text{non-r}}$ ) este redată în figura 3.2.12 pe același grafic:

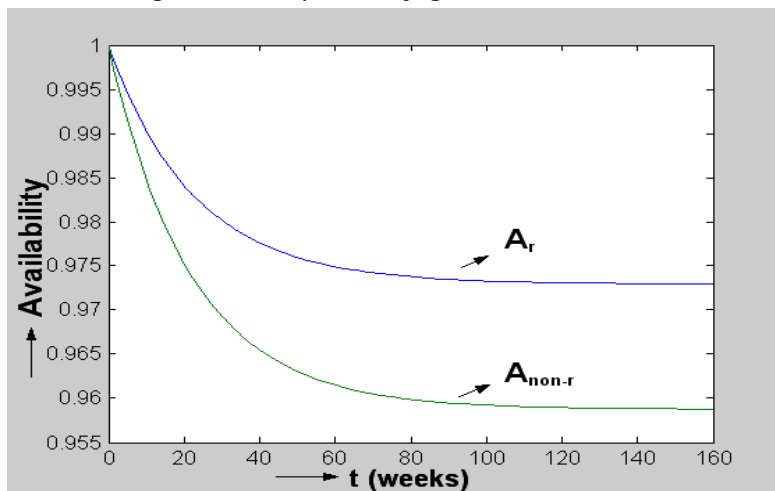


Fig. 3.2.12 Reprezentarea disponibilităților în cazul în care structurile prezintă sau nu redundanță



După cum se observă în figura 3.2.12, sistemul care cuprinde structurile cu redundanță dublă și triplă are o disponibilitate superioară celui simplu, fără redundanțe. Se confirmă practic faptul că disponibilitatea crește în cazul în care un sistem prezintă redundanță. În situația curentă prin aplicarea redundanței în două puncte critice apare o diferență considerabilă, de peste 1,5%.

### 3.3 Studiu de caz pentru dezvoltările teoretice referitoare la tranzițiile simultane în cazul utilizării structurilor reparabile cu redundanță activă

În acest subcapitol se face un studiu pe un caz real, utilizând aspectele teoretice din cadrul subcapitolului 2.4. Situația reală se referă la sistemul automat reprezentat de centrala electrică geotermală (vezi subcapitolul 3.1), mai precis la o structură din cadrul sistemului studiat în subcapitolul 3.2 și anume: sistemul de control al ciclului termodinamic dintr-o centrală electrică geotermală.

În subcapitolul 3.2 s-a analizat sistemul condus al buclei de reglare a temperaturii  $t_1$ .

Pentru a evidenția aportul adus de considerarea tranzițiilor simultane (vezi subcapitolul 2.4) cu privire la disponibilitate, de data aceasta se compară prima structura redundanță (ilustrată în fig. 3.2.2 din cadrul subcapitolului 3.2) cu o structură redundanță pentru care se au în vedere și tranzițiile simultane între stări.

Studiul se mărginește doar la unul dintre punctele critice din cadrul sistemului condus al buclei de reglare a temperaturii  $t_1$ , și anume ansamblul de componente serie: ansamblul acționare debit motor+RB2 ilustrat în fig. 3.3.1.

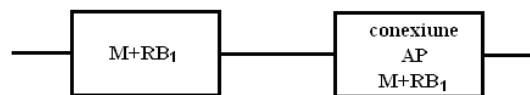


Fig. 3.3.1 Ansamblul serie acționare debit motor+RB<sub>1</sub>

Pentru creșterea disponibilității, subansamblul serie este conceput de manieră redundanță, practic fiind vorba despre o redundanță paralel de ordin 2. Sistemul rezultat este ilustrat în Fig. 3.3.2.

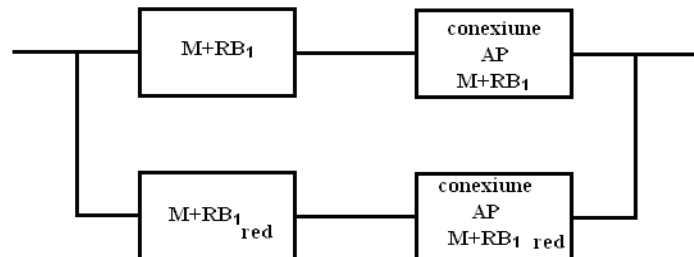


Fig. 3.3.2 Structură redundanță de ordin 2 a ansamblului din fig. 3.3.1

Obiectul studiului îl constituie realizarea unei comparații între:

- 1) situația în care nu se admit tranzițiile simultane dintr-o stare în alta
- 2) situația în care se iau în considerare tranzițiile simultane a două componente din starea de defectare în starea de reparare sau invers.

Valorile pentru care se efectuează simularea sunt prezentate în tabelul 3.2. Ele reprezintă subsamblul 7 al tabelului 3.1 din subcapitolul anterior la care s-au adăugat ratele de tranziție simultană dintr-o stare în alta.

Tabel 3.2	$\lambda$ (hour <sup>-1</sup> )	$\mu$ (hour <sup>-1</sup> )	$\mu_{i+1,i}$ (hour <sup>-1</sup> )
1. RB <sub>1</sub>	0.0001543	0.05	$\mu_{10} = 0.0470,$ $\mu_{21} = 0.04715,$ $\mu_{20} = 0.0029,$ $\lambda_{02} = 0.0000145$
2. motor RB <sub>1</sub>	0.000057	0.0416	
3. conexiune AP-motor RB <sub>1</sub>	0.00021	0.05	

Calcululele fac uz de formulele (26) și (36) din paragraful 2.3.3, care furnizează modalitatea de calcul pentru rata de defectare și de reparare rezultantă pentru un sistem ale cărui componente sunt conectate serie (ansamblul de acționare debit motor și RB2 sunt reprezentă o conexiune serie și se cunosc doar intensitățile de reparare și de defectare individuale). Astfel:

$$\lambda_s = \lambda_1 + \lambda_2 + \lambda_3 \quad (7)$$

$$\mu_s = \frac{\lambda_1 + \lambda_2 + \lambda_3}{\frac{\lambda_1}{\mu_1} + \frac{\lambda_2}{\mu_2} + \frac{\lambda_3}{\mu_3}} \quad (8)$$

Pentru situația în care se admit tranzițiile simultane între stări se continuă studiul prin a se calcula ratele de defectare  $\lambda_{01}$  și  $\lambda_{12}$ , care se obțin cu ajutorul relațiilor (39) din subcapitolul 2.4. Astfel, după ce se rezolvă sistemul de ecuații:

$$\left\{ \begin{array}{l} \lambda_{12} = \frac{\mu_s (\lambda_{01} \cdot \mu_{21} + \lambda_{02} \cdot \mu_{21} + \lambda_{01} \cdot \mu_{20}) - 2 \cdot \lambda_{02} \cdot \mu_{10} \cdot \lambda_s}{2 \cdot \lambda_s \cdot (\lambda_{01} + \lambda_{02})} \\ \mu_{10} \cdot \mu_{21} + \mu_{10} \cdot \mu_{20} + \mu_{20} \cdot \left( \frac{\mu_s (\lambda_{01} \cdot \mu_{21} + \lambda_{02} \cdot \mu_{21} + \lambda_{01} \cdot \mu_{20}) - 2 \cdot \lambda_{02} \cdot \mu_{10} \cdot \lambda_s}{2 \cdot \lambda_s \cdot (\lambda_{01} + \lambda_{02})} \right) = \\ = \frac{\lambda_s}{2 \cdot \mu_s} \cdot (\lambda_{01} \cdot \mu_{21} + \lambda_{02} \cdot \mu_{21} + \lambda_{01} \cdot \mu_{20}) \end{array} \right.$$

se implementează schema de simulare din fig. 3.3.3. Disponibilitatea este o funcție de timp și ca urmare pentru determinarea ei este necesar un studiu numeric, prin simulare.

A doua situație este cea în care nu se admit tranziții simultane a două componente dintr-o stare în alta. Partea teoretică referitoare la această situație este descrisă în subcapitolul 2.3, iar partea practică în subcapitolul 3.2.

Valorile pentru care se face implementarea sunt extrase din tabelul 3.1 (din subcapitolul 3.2), cele corespunzând subsamblului 7:

Tabel 3.3	$\lambda$ (hour <sup>-1</sup> )	$\mu$ (hour <sup>-1</sup> )	$\mu_{i+1,i}$ (hour <sup>-1</sup> )
1. RB <sub>1</sub>	0.0001543	0.05	$\mu_{10} = 0.0470,$ $\mu_{21} = 0.04715.$
2. Motor RB <sub>1</sub>	0.000057	0.0416	
3. conexiune AP - motor RB <sub>1</sub>	0.00021	0.05	

Ratele de defectare și de reparare ale structurii serie se calculează similar celuilalt caz. Ratele de defectare ale sistemului privit ca întreg,  $\lambda_{01}$  și  $\lambda_{12}$ , se calculează cu relațiile (3) din cadrul subcapitolului 2.3.

3.3 Studiu de caz pentru dezvoltările teoretice referitoare la tranzițiile simultane 85  
 în cazul utilizării structurilor reparabile cu redundanță activă

Schema de implementare pentru această situație este ilustrată de asemenea în fig. 3.3.3.

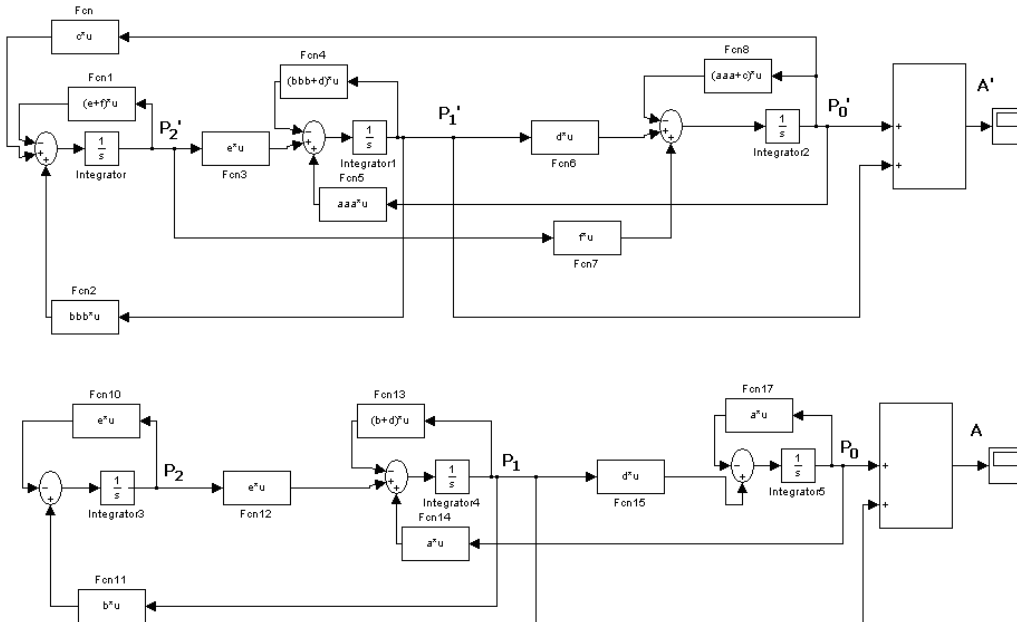


Fig. 3.3.3 Schemele de implementare ale structurii care admite rate de tranziție simultane (sus), respectiv a structurii care nu admite rate de defectare simultane (jos)

Primul set de simulări se efectuează pentru a afla evoluțiile în timp ale probabilităților  $P_0$  (în cazul în care nu se admit tranziții de pas dublu) și  $P_0'$  (pentru cazul când se admit tranzițiile de pas dublu), care reprezintă probabilitățile ca structura redundantă să se afle în starea  $\textcircled{0}$ . Starea  $\textcircled{0}$  înseamnă situația de totală funcționare a structurii, fără nici o componentă defectă. Rezultatele furnizate în ambele situații sunt prezentate în fig. 3.3.4.

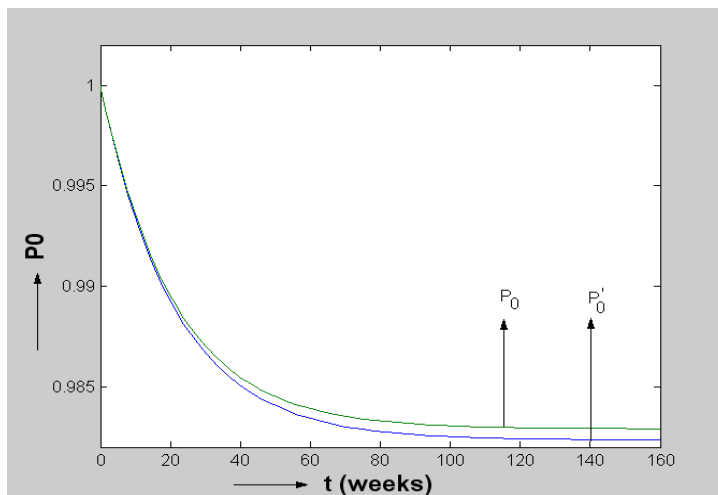


Fig. 3.3.4 Evoluția în timp a probabilității ca sistemul să se afle în starea  $\textcircled{0}$

Următoarele simulări urmăresc probabilitatea ca sistemul să se afle în starea ① (o componentă defectă), respectiv în starea ② (starea de defectare totală a ansamblului redundant). Și de data aceasta se studiază atât evoluția în timp a probabilităților  $P_1$ ,  $P_2$  (în cazul în care nu se admit tranziții de pas dublu) cât și a probabilităților  $P_1'$ ,  $P_2'$  (pentru cazul când se admit tranzițiile de pas dublu). Rezultatele furnizate sunt prezentate comparativ în figurile 3.3.5 și 3.3.6 care surprind faptul că  $P_1'$  și  $P_2'$  au valori mai mari decât  $P_1$  și  $P_2$ . Situația este diferită de cea referitoare la  $P_0$  și  $P_0'$ .

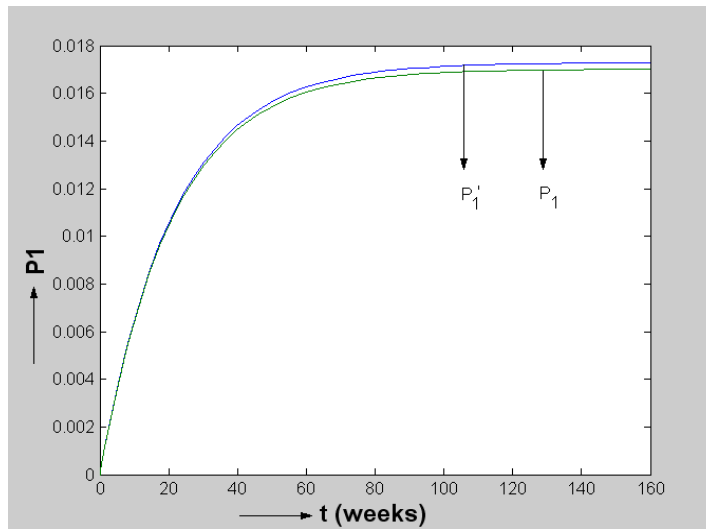


Fig. 3.3.5 Evoluția în timp a probabilității ca sistemul să se afle în starea ①

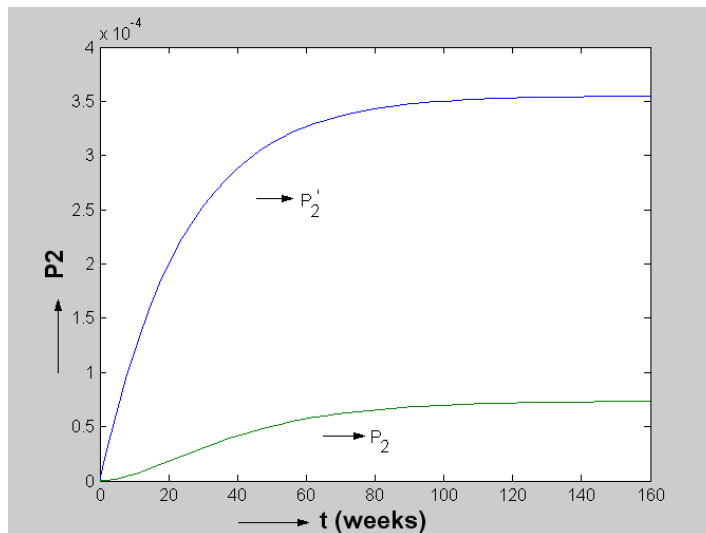


Fig. 3.3.6 Evoluția în timp a probabilității ca sistemul să se afle în starea ②

În final, figura 3.3.7 ilustrează rezultatul final (și global) al studiului care se referă la evoluția disponibilității sistemului în timp. Se prezintă spre comparație:

- Disponibilitatea A – pentru situația în care nu se admit tranziții simultane
- Disponibilitatea A' – pentru cazul în care se admit tranzițiile simultane, de pas dublu

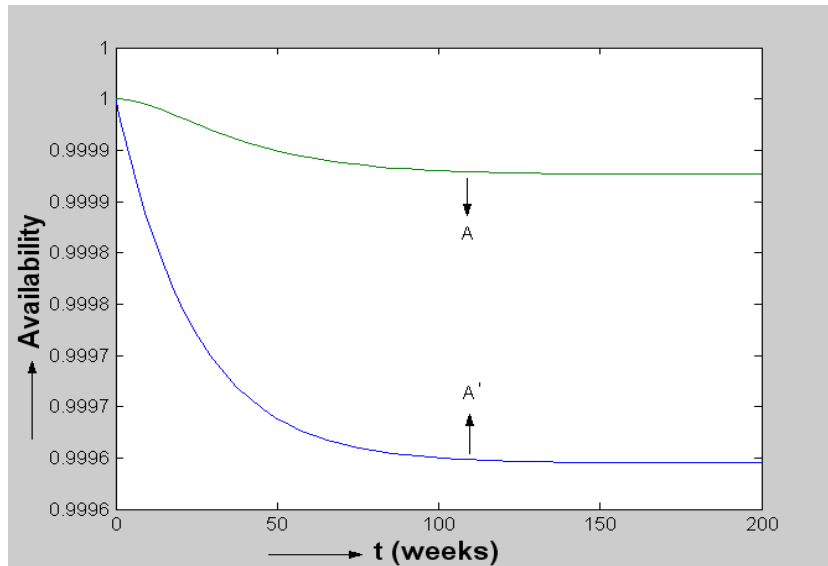


Fig. 3.3.7 Reprezentarea disponibilităților pentru cele două cazuri

Concluzia este aceea că surprinderea tranzițiilor simultane conduce la disponibilități de valori mai mici decât în cazul în care tranzițiile simultane sunt neglijate. În cazul de față diferențele nu sunt însă semnificative.

### 3.4 Regulatele interpolative

Interpolarea reprezintă o metodă de generalizare pentru cazuri reale, diferite de cele observate, a rezultatelor obținute prin observații discrete, bazată pe datele provenite de la cazul real și cel observat. De obicei, generalizarea se realizează prin definirea unor funcții de interpolare. În ingineria reglării, experții utilizează de regulă interpolarea sub forma unor funcții scalare de variabile scalare și sub formă de tabele de corespondență unidimensionale. Independent de metoda de interpolare utilizată, oricărei operații de interpolare îi corespunde un bloc de calcul. În particular, în Matlab/Simulink se folosesc tabelele "look-up". În contextul raționamentului interpolativ se consideră regulatele interpolative implicite și explicite [59], [61].

Principalele tehnici interpolative utilizate în reglajul automat sunt metodele fuzzy, de interpolare propriu-zisă și neurale. Regulatele interpolative implicite includ modelele fuzzy, rețelele neurale și orice combinații ale acestora [125]. În [33], este studiată o variantă de implementare interpolativă a unor structuri de regulate fuzzy și neurale, derivate dintr-un regulator PI, iar ca alternativă se prezintă un regulator interpolativ propriu-zis (prezentat în variantă RIP) care conduce la performanțe de reglare mai bune.

Modelele RIP (Rule based InterPolative) aparțin metodelor de reglare interpolative explicite, implementate în mod obișnuit utilizând tabele de corespondență obținute prin diverse metode. Un regulator interpolativ propriu-zis și denumit în continuare în mod simplu regulator interpolativ, reprezintă un regulator la care mărimea de comandă se obține în funcție de eroarea de reglare sau de mărimi din care aceasta poate fi determinată prin interpolare într-o mulțime de puncte de sprijin reținută tabelar. Tabelele sunt configurate astfel încât sistemul de reglare să obțină performanțele impuse prin proiectare.

În aplicațiile de conducere apare adeseori și necesitatea implementării unei dependențe neliniare  $y = f(u)$  inclusă într-o lege de reglare proiectată. Implementarea acestor dependențe neliniare poate ridica probleme de acordare în aplicația practică. Astfel, o dependență proiectată  $y = f(u)$  ajunge în final, la implementare, să fie modificată sub forma  $y = f'(u)$ . Modificarea poate avea caracter local, parțial sau general, iar efectul poate fi o implementare relativ complicată, chiar și în variantă numerică, a dependenței  $y = f'(u)$ . În acest context apare ca o soluție alternativă implementarea interpolativă prin blocuri lookup-table.

În particular, dependența  $y = f(u)$  poate reprezenta chiar o lege de reglare de tip neinerțial. În fig. 3.4.1 se prezintă simbolizarea unui regulator interpolativ, cu orientarea de la  $u$  la  $y$ , care implementează de o manieră interpolativă o caracteristică de transfer intrare-ieșire  $y=f(u)$ . Este similară reprezentării unui element neliniar.

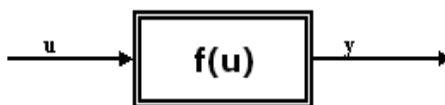


Fig. 3.4.1 Structura simplă a unui regulator interpolativ

Obținerea unei caracteristici de transfer interpolative poate fi privită, totodată, ca o metodă de sinteză care pleacă de la rezultate de proiectare  $y=f(u)$  determinate prin alte metode și care urmărește îmbunătățirea aplicației prin sinteza unei dependențe  $y= f'(u)$  implementată tabelar și folosirea unei metode de interpolare [33]. Gradele de libertate de care dispune proiectantul sunt reprezentate de structura tabelului prin care se implementează dependența respectivă, adică de fiecare punct din tabel și de metoda de interpolare utilizată.

Un punct de sprijin este un  $(n+1)$ -uplu în care primele  $n$  elemente definesc locul de sprijin iar ultimul valoarea de sprijin (în cazul reglatoarelor valoarea de sprijin este valoarea mărimii de comandă). Practic, considerăm că un punct de sprijin  $P_i$  este un punct într-un spațiu  $(n+1)$  dimensional,  $P_i(V_{1i}, V_{2i}, \dots, V_{ni}, R_i)$ , în care coordonatele  $(V_{1i}, V_{2i}, \dots, V_{ni})$  corespund locului de sprijin iar  $R_i$  este valoarea de sprijin asociată locului de sprijin. Mulțimea punctelor de sprijin se notează cu  $P = \{P_i\}$ . Se consideră că  $P$  definește prin puncte o funcție care asociază locurilor de sprijin punctele de sprijin corespunzătoare  $R = f(V_1, V_2, \dots, V_n)$ .

Metoda de interpolare este o metodă de calcul a valorii funcției  $f$  pentru un punct  $(V_1, V_2, \dots, V_n)$  arbitrar folosind punctele de sprijin din tabelul de interpolare. Ea poate fi de diverse tipuri. De regulă metoda de interpolare aleasă este cea liniară.

Fie  $V_{ab}$  o coordonată a locului de sprijin corespunzător punctului de sprijin  $P_i$ . Indicele  $a$  ia valorile  $a \in \{1, 2, \dots, n\}$  și indică rangul coordonatei. Indicele  $b$  reprezintă eticheta coordonatei de rang  $a$  și ia valori naturale  $b \in \{1, 2, \dots, n_b\}$ . În acest caz, pentru valoarea coordonatei de rang  $a$ ,  $V_a$ , a punctului curent  $P(V_1, V_2, \dots, V_n, R)$  este valabilă condiția de apartenență:

$$V_a, i_a \leq V_a < V_a, i_a + 1, \quad a \in \{1, 2, \dots, n\}, i_a \in \{1, 2, \dots, n_b - 1\}, \quad (9)$$

Mulțimea indicilor  $i_a + 1$ ,  $a \in \{1, 2, \dots, n\}$ , asociați punctului  $P$ , notată cu  $\{i_a + 1\}_P$ , definește *locul de sprijin esențial* asociat punctului  $P$  pe baza coordonatelor  $(V_1, V_2, \dots, V_n)$ :  $\{i_a + 1\}_P = (i_1 + 1, i_2 + 1, \dots, i_n + 1)$ . Locurile de sprijin esențiale sunt utilizate în cadrul studiilor prezentate în subcapitolele 3.5 și 4.3.

Pentru exemplificare, în figura următoare se consideră cazul interpolării corespunzătoare unei funcții de trei variabile  $R = f(V_1, V_2, V_3)$ . Aceasta înseamnă că prin interpolare se determină puncte de forma  $P(V_1, V_2, V_3, R)$ , cu valoarea lui  $R$  calculată potrivit unei metode de interpolare alese. Pentru un punct curent vom avea:

$$V_{1,i} \leq V_1 < V_{1,i+1}, \quad V_{2,j} \leq V_2 < V_{2,j+1}, \quad V_{3,k} \leq V_3 < V_{3,k+1} \quad (9')$$

unde  $i, j$  și  $k$  reprezintă etichetele asociate celor trei coordonate  $i \in \{1, 2, \dots, n_1\}$ ,  $j \in \{1, 2, \dots, n_2\}$ ,  $k \in \{1, 2, \dots, n_3\}$ . Locul de sprijin esențial pentru cazul din (9') este mulțimea de etichete  $\{i + 1, j + 1, k + 1\}$ . Lui îi corespunde punctul de sprijin esențial:

$$P_{i+1, j+1, k+1} = P(V_{1,i+1}, V_{2,j+1}, V_{3,k+1}, f(V_{1,i+1}, V_{2,j+1}, V_{3,k+1}))$$

Pentru simplitate, în continuare se operează cu notațiile  $P_{i,j,k}$  și  $R_{i,j,k} = f(V_{1,i}, V_{2,j}, V_{3,k})$ .

Valorile de sprijin se găsesc în tabele unidimensionale, sau multidimensionale. Interpolarea se realizează după setarea tabelului de interpolare potrivit aplicației. Implementări ale unor regulatoare complexe interpolative se găsesc în [62], [94], [37].

### 3.5 Soluție de reglare a temperaturii bazată pe interpolare

Subcapitolul prezintă o soluție de reglare a temperaturii  $CO_2$  la ieșirea din blocul vaporizatoare, parte a centralei electrice geotermale (vezi subcapitolul 3.1). Soluția se bazează pe un regulator tripozițional cu reacție operațională. Dinamica regulatorului este adaptată într-o manieră interpolativă în funcție de variațiile perturbațiilor măsurabile care acționează asupra blocului vaporizatoare.

Subcapitolul începe prin descrierea elementelor legate de regulatorul tripozițional cu reacție operațională, urmată de prezentarea structurii de reglare

existentă în cadrul blocului vaporizatoare evidențiindu-se neajunsurile acesteia. Se prezintă apoi soluția propusă de reglare a temperaturii CO<sub>2</sub> din blocul vaporizatoare, iar în final se ilustrează eficiența structurii propuse care relevă performanțe mai bune atât din punct de vedere dinamic, cât și staționar.

### 3.5.1 Regulator tripozițional cu reacție operațională cu acțiune de tip PI

Elemente legate de reglatoarele interpolative au fost prezentate în cadrul subcapitolului 3.4.

Este bine cunoscut faptul că pentru elementele de execuție cu un grad de inerție ridicat, pentru a obține o precizie de reglare sigură, pentru a reduce gradul de consum de energie, sau pentru a evita uzura, se utilizează blocuri de control nelineare (NB) bazate pe elemente bipoziționale sau tripoziționale. Acestea se consideră a avea un comportament dinamic cvasi-liniar de tipul PD, PI sau PID, conferit de utilizarea elementelor adecvate de corecție operațională locală (CE) și prin combinarea acțiunii acestora cu acțiunea elementelor de execuție (A). Comportarea acestor tip de reglatoare a fost studiată intensiv înainte de 1990 [18], [36], [71], [96], [126]. În figura de mai jos se prezintă schema bloc a unui regulator PI cu filtrare obținut prin combinarea unui element tripozițional cu histerzis, un bloc de reacție operațională de tip PDT<sub>1</sub> cu funcția de transfer

$$H_{CE}(s) = K_{CE} \cdot \frac{T'_{CE} \cdot s + 1}{T_{CE} \cdot s + 1} \quad (10)$$

și un element de execuție de tip integrator (A).

Pentru regulatorul din figura 3.5.1 este valabil un model matematic intrare-ieșire de aproximare de forma:

$$T_F \cdot \dot{h}_1(t) + h_1(t) - h_1(t_0) = K_R \cdot \left[ a(t) + \frac{1}{T_i} \int_{t_0}^t a(\tau) d\tau \right] \quad (11)$$

unde  $K_R$ ,  $T_i$ , și  $T_F$  sunt în funcție de parametrii  $K_A$ ,  $K_{CE}$ ,  $T_{CE}$ ,  $\varepsilon_1$ ,  $u_{10}$  ilustrați în figură. Limitele de integrare  $t_0$  și  $t$  reprezintă momentul inițial și cel curent în utilizarea regulatorului.

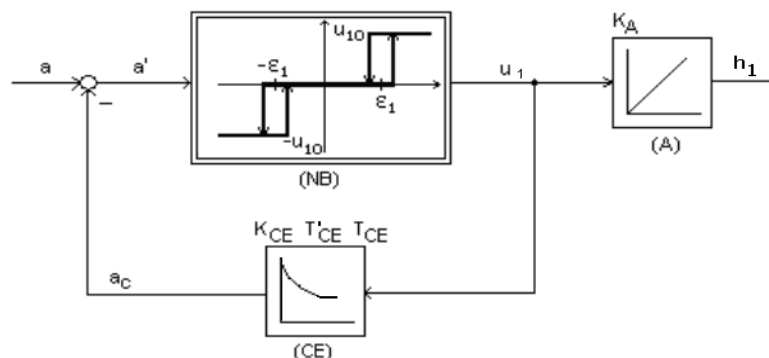


Fig. 3.5.1 Regulator PI cu filtrare realizat cu element tripozițional cu histerzis și reacție operațională



Nivelele nenule  $-u_{10}$  și  $u_{10}$  ale caracteristicii tripoziționalului cu histerzis (centrate în jurul valorilor  $-\varepsilon_1$  și  $\varepsilon_1$ ) asigură creșterea sau descreșterea ieșirii  $m$  a elementului de acționare cu viteză constantă, iar zona de insensibilitate asigură menținerea ieșirii constante în regim staționar. Histerzisul limitează frecvența de comutare între nivelele  $u_{10}$  și  $0$ , respectiv  $0$  și  $-u_{10}$ . Pentru o implementare numerică, limitarea prin histerzis nu este în general necesară atât timp cât procesul de eșantionare acționează prin pasul de eșantionare  $h$  sub forma unei limitări implicite.

### 3.5.2 Structura de reglare a temperaturii la ieșirea din blocul vaporizatoare al centralei electrice geotermale.

Cercetarea se bazează pe sistemul de reglare al temperaturii  $\theta_1$  a  $\text{CO}_2$  după blocul vaporizatoare al centralei electrice geotermale din figura 3.5.2. Elemente legate de centrala electrică geotermală au fost menționate în cadrul subcapitolului 3.1. Circuitul  $\text{CO}_2$  este reprezentat cu linie îngroșată, iar în partea stângă a figurii este evidențiat sistemul de reglare menționat.

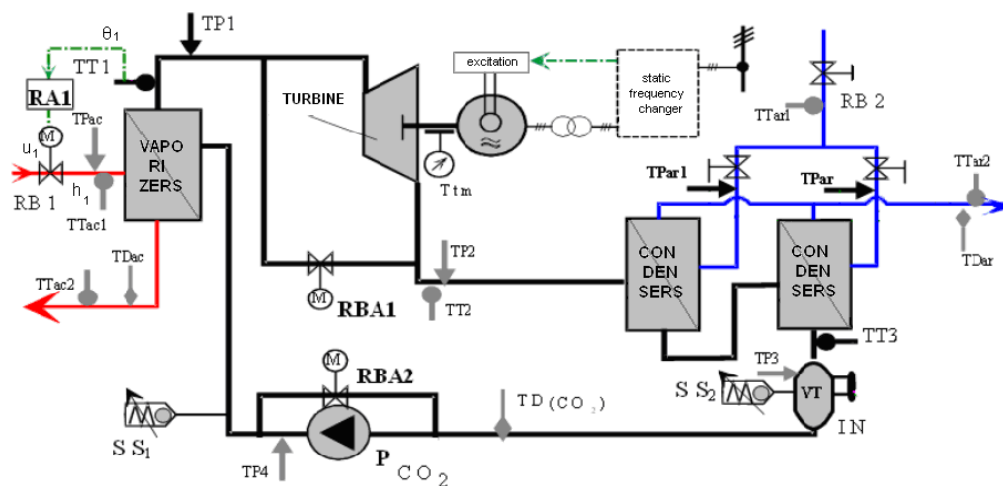


Fig. 3.5.2. Poziția buclei actuale de reglare a temperaturii  $\theta_1$ .

Figura prezintă versiunea inițială a sistemului de reglare folosită ca punct de plecare și reper pentru o nouă structură, cu bloc interpolativ. Temperatura  $\theta_1$  este măsurată de traductorul  $TT_1$ . Regulatorul  $RA_1$  utilizează eroarea de temperatură  $\theta_{1,ref} - \theta_1$  pentru a controla deschiderea valvei  $RB_1$  care controlează (comandă) debitul de apă geotermală  $q_{ac}$ .

S-a considerat că sistemul actual de reglare trebuie reprojecat datorită următoarelor deficiențe referitoare la controlul temperaturii  $\theta_1$ :

- i) suprareglaj mai mare de 10 %,
- ii) timp de reglare pentru ciclul termodinamic mai mare de 30 de minute,
- iii) precizia inadecvată (erori de simulare mai mare de  $1^\circ\text{C}$  și erori în practică mai mari de  $1.6^\circ\text{C}$ ).

Schimbul de căldură în blocul vaporizatoare este un proces modelabil sub forma sistemului dinamic (4) de ordin doi cu orientarea din figura 3.5.3:

$$\{h_1, \delta_{pac}, q_{CO_2_{pv}}, \theta_4, \theta_{iac}\} \rightarrow \{q_{ac}, \theta_1, \theta_{eac}, q_{CO_2_{vom}}\}. \quad (3)$$

Din cele cinci marimi de intrare,  $h_1$  reprezintă variabila de control a valvei  $RB_1$  și  $\text{delta\_}p_{ac}$ ,  $\theta_{iac}$  (variația presiunii și temperaturii apei geotermale), respectiv  $\theta_4$ ,  $q_{CO_2\_pv}$  (variațiile temperaturii și debitului de  $CO_2$  după pompă) sunt considerate perturbații. Atât timp cât perturbațiile din cadrul primului grup au o variație neglijabilă, ele pot fi considerate constante.

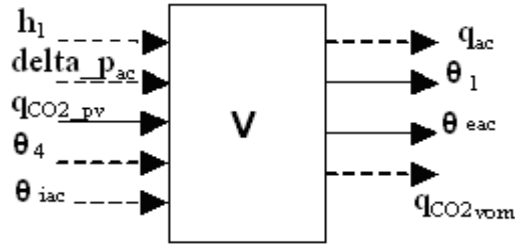


Fig. 3.5.3 Blocul vaporizatoare

Modelul simplificat al blocului vaporizatoare stabilit în [2] reprezintă un sistem nelinear care poate fi rescris sub forma:

$$\left\{ \begin{array}{l} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_{eac} \end{bmatrix} = \begin{bmatrix} f_3(q_{CO_2\_pv}, h_1, \text{delta\_}p_{ac}) & f_2(q_{CO_2\_pv}, h_1, \text{delta\_}p_{ac}) \\ f_4(q_{CO_2\_pv}, h_1, \text{delta\_}p_{ac}) & f_5(q_{CO_2\_pv}, h_1, \text{delta\_}p_{ac}) \end{bmatrix} \cdot \begin{bmatrix} \theta_1 \\ \theta_{eac} \end{bmatrix} + \\ + \begin{bmatrix} 0 & 0 & 0 & -f_3(q_{CO_2\_pv}, h_1, \text{delta\_}p_{ac}) & f_2(q_{CO_2\_pv}, h_1, \text{delta\_}p_{ac}) \\ 0 & 0 & 0 & f_4(q_{CO_2\_pv}, h_1, \text{delta\_}p_{ac}) & f_6(q_{CO_2\_pv}, h_1, \text{delta\_}p_{ac}) \end{bmatrix} \cdot \begin{bmatrix} h_1 \\ \text{delta\_}p_{ac} \\ q_{CO_2\_pv} \\ \theta_4 \\ \theta_{iac} \end{bmatrix} \\ \\ \begin{bmatrix} q_{ac} \\ \theta_1 \\ \theta_{eac} \\ q_{CO_2\_vom} \end{bmatrix} = \begin{bmatrix} 00 \\ 10 \\ 01 \\ 00 \end{bmatrix} \cdot \begin{bmatrix} \theta_1 \\ \theta_{eac} \end{bmatrix} + \begin{bmatrix} f_1(h_1, \text{delta\_}p_{ac}) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & f_7(\theta_1) & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} h_1 \\ \text{delta\_}p_{ac} \\ q_{CO_2\_pv} \\ \theta_4 \\ \theta_{iac} \end{bmatrix} \end{array} \right. \quad (13)$$

unde,

$$f_1(h_1, \text{delta\_}p_{ac}) = \frac{0.1935}{\sqrt{0.5625 + h_1^2}} \cdot \sqrt{\text{delta\_}p_{ac}} \quad (14)$$

$$R_1 = R_{1nom} \left( \left( \frac{m_{CO_2\_nom}}{q_{CO_2\_pv}} \right)^{0.8} + \left( \frac{m_{acnom}}{\frac{0.1935 \cdot h_1}{\sqrt{0.5625 + h_1^2}} \cdot \sqrt{\text{delta\_}p_{ac}}} \right)^{0.8} \right) \quad (15)$$

$$f_2(q_{co2\_pv}, h_1, delta\_pac) = \frac{1}{2 \cdot R_1 \cdot m_{CO_2nom} \cdot c_{CO_2}^{med_v}} \quad (16)$$

$$f_3(q_{co2\_pv}, h_1, delta\_pac) = \frac{q_{CO_2-pv}}{m_{CO_2nom}} + f_2(q_{co2\_pv}, h_1, delta\_pac) \quad (17)$$

$$f_4(q_{co2\_pv}, h_1, delta\_pac) = \frac{-1}{2 \cdot R_1 \cdot m_{acnom} \cdot c_a} \quad (18)$$

$$f_5(q_{co2\_pv}, h_1, delta\_pac) = \frac{-f_1(h_1, delta\_pac) \cdot h_1}{m_{acnom}} - f_4(q_{co2\_pv}, h_1, delta\_pac) \quad (19)$$

$$f_6(q_{co2\_pv}, h_1, delta\_pac) = \frac{f_1(h_1, delta\_pac) \cdot h_1}{m_{acnom}} - f_4(q_{co2\_pv}, h_1, delta\_pac) \quad (20)$$

$$f_7(\theta_1) = \begin{cases} v_1 + (\theta_1 - 60) \cdot 0.11786, & \text{pentru } \theta_1 > 60^\circ \\ v_1 + (\theta_1 - 60) \cdot 0.0263, & \text{altfel} \end{cases} \quad (21)$$

Notațiile sunt următoarele:

- $t$  - timpul
- $i_x$  - entalpia în punctul  $x$  al ciclului termodinamic ( $x = 1$  - la ieșirea blocului vaporizatoare,  $x = 2$  - la ieșirea din turbină,  $x = 3$  - la ieșirea blocului condensatoare,  $x = 4$  - la ieșirea pomper de CO<sub>2</sub>)
- $v_x$  - volumul CO<sub>2</sub> în punctul  $x$
- $q_{xnom}$  - nominal weight rate in point  $x$  of the thermo-dynamical cycle
- $q_{ac}$  - debitul apei calde
- $q_{CO_2\_pv}$  - debitul de CO<sub>2</sub> lichid la intrarea în blocul vaporizatoare
- $m_{acnom}$  - valoarea nominală a masei apei geotermale din blocul vaporizatoare
- $delta\_pac$  - diferența de presiune a apei calde în ansamblul RB<sub>1</sub>+vaporizatoare
- $\theta_4$  - temperatura CO<sub>2</sub> la intrarea în blocul vaporizatoare
- $\theta_1$  - temperatura măsurată a CO<sub>2</sub> la ieșirea din blocul vaporizatoare
- $\theta_{iac}$  - temperatura apei calde la intrarea în vaporizatoare
- $\theta_{eac}$  - temperatura apei calde la ieșirea din vaporizatoare
- $h_1$  - deschiderea valvei RB1
- $c_{CO_2}^{med_v}$  - coeficientul de căldură specifică medie a CO<sub>2</sub> în vaporizatoare  
 $c_{CO_2}^{med_v} = \frac{(i_1 - i_4)_{nom}}{\theta_1 - \theta_4}$ , s-a considerat  $c_{CO_2}^{med_v} = 4100 \frac{kJ}{kg \cdot ^\circ C}$
- $R_1$  - coeficientul de schimb de căldură global; s-a considerat valoarea nominală  
 $R_{1nom} = \frac{1}{K_1 \cdot S_v}$ , unde  $K_1$  - coeficientul de schimb de căldură global în vaporizatoare,  $S_v$  - suprafața de vaporizare

### 3.5.3 Soluția de reglare a temperaturii CO<sub>2</sub> la ieșirea blocului vaporizatoare

#### 3.5.3.1 Structura regulatorului

Ca o soluție alternativă pentru schema de reglare a temperaturii  $\theta_1$  cea menționată în paragraful anterior s-a implementat schema de reglare din figura 3.5.4, unde blocul regulator a fost notat cu RA'1. Reglarea în blocul vaporizatoare se realizează prin deschiderea valvei de către ansamblul M1+RB1 (motor + valvă).

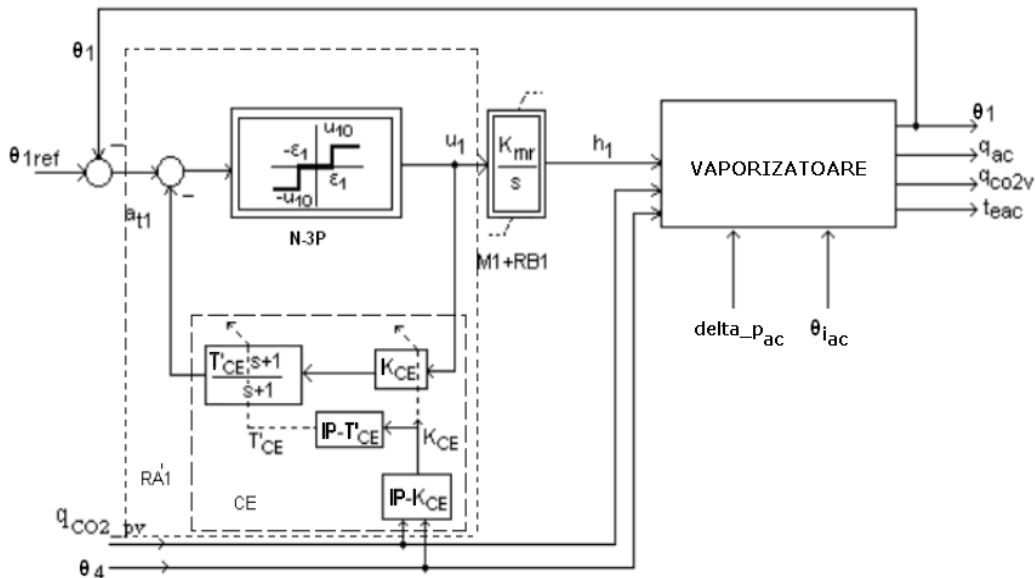


Fig. 3.5.4. Bucla de reglare modificată a temperaturii  $\theta_1$  în cadrul centralei electrice geotermale

Reglarea se realizează de această dată prin regulatorul din figura 3.5.1, compus din elementul tripозиțional cu histerezis împreună cu blocul de pe reacție, N-3P, elementul de corecție operațională CE (cu  $T_{CE} = 1$  sec) și prin ansamblul motor + valvă, asimilată în regim dinamic (atunci când valva se deschide sau se închide) printr-un element integrator cu funcția de transfer

$$H_{M1+RB1}(s) = \frac{K_{mr}}{s} . \quad (22)$$

Diferența conceptuală față de schema din figura 3.5.1 o reprezintă faptul că parametrii  $K_{CE}$  și  $T'_{CE}$  corespunzători elementului de corecție sunt în permanență ajustabili în funcție de valorile perturbațiilor  $q_{CO2\_pv}$  și  $t_4$  prin blocurile de interpolare IP-  $K_{CE}$  și IP-  $T'_{CE}$  (elemente legate de interpolare au fost prezentate în subcapitolul 3.4).

Referitor la regulatorul RA'1, sunt esențiale următoarele specificații:

- i) Nivelul  $u_{10}$  și valoarea parametrului  $T_{CE}$  sunt corelate cu valoarea  $K_{mr}$  astfel încât valva RB<sub>1</sub> să se deschidă din poziția complet închis până la poziția complet deschis într-un minut.

- ii) Lățimea zonei de insensibilitate ( $2 \cdot \varepsilon_I$ ) este corelată cu precizia de reglare a temperaturii  $\theta_1$ , astfel încât în regim staționar să se obțină:  $|\theta_{1ref} - \theta_1| < 0.005 \theta_{1ref}$ .
- iii) Regulatorul tripozițional N-3P nu are histerezis. Considerând faptul că regulatorul RA'1 este implementat numeric, histerezisul necesar pentru limitarea frecvenței pentru mărimea reglată  $u_1$ , este obținut automat prin perioada de eșantionare  $h$ . Comutarea mărimii de reglare de pe un nivel de altul se realizează cu o întârziere de  $h$  secunde, similar comportamentului unui tripozițional cu buclă de reacție cu un histetezis variabil.
- iv) Modelul (22) pentru ansamblul M1 + RB1 este un model de aproximare liniară. În realitate, grupul considerat este un subsistem nelinear de ordin mai mare decât 1. Limitarea ilustrată în figura 3.5.4 corespunde poziției complet deschis (limita superioară) și complet închis (limita inferioară) a valvei, poziții care sunt menținute atâta timp cât polaritatea mărimii  $u_1$  nu se schimbă.
- v) Datorită blocului de corecție din figura 3.5.1, structura implementează cu aproximație un algoritm de forma (11).

Parametrii  $T_F, K_R, T_i$  sunt ajustați în permanență prin  $K_{CE}$  și  $T'_{CE}$  furnizați de blocurile de interpolare IP -  $T'_{CE}$  și IP -  $K_{CE}$ .

Figura 3.5.5 ilustrează poziția noii structuri de reglare (prezentată în figura 3.5.4) în cadrul circuitului de CO<sub>2</sub> din centrala electrică geotermală.

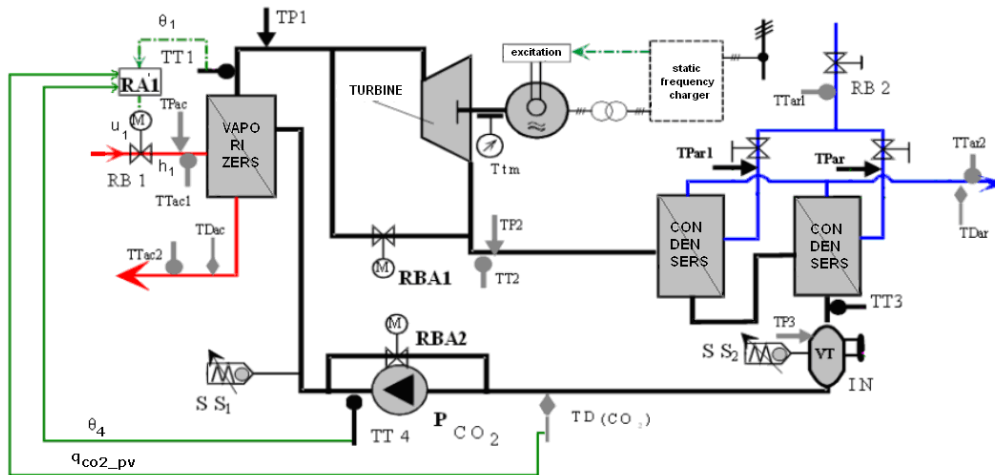


Fig. 3.5.5 Poziția noii bucle de reglare a temperaturii  $\theta_1$

### 3.5.3.2 Aspecte referitoare la proiectarea regulatorului RA'1

Proiectarea regulatorului RA'1 constă în stabilirea valorilor pentru parametrii  $\varepsilon_I, u_{10}, K_{CE}$  și  $T'_{CE}$ .

- Valoarea parametrului  $\varepsilon_I=0.15$  a fost aleasă pentru obținerea temperaturii  $\theta_1$  considerându-se precizia senzorului de temperatură în intervalul  $\pm 0.3$  °C.

- Valoarea  $u_{10}$  este obținută din condiția impusă pentru  $h_1$  de a parcurge distanța de la poziția complet închis la poziția complet deschis,  $0 \leq h_1 \leq 1$ , într-un minut (precum în figura 3.5.6a). Efectul parametrului  $T_{CE}$  este neglijabil.

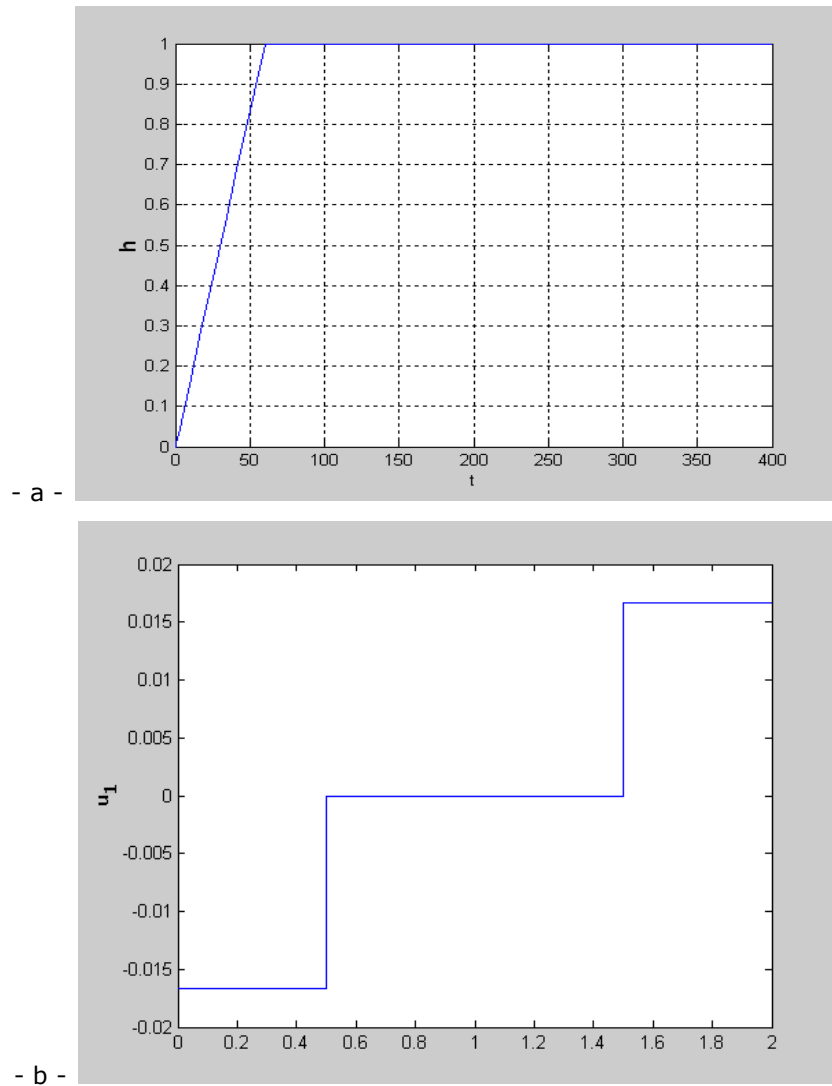


Fig. 3.5.6. Mișcarea de deschidere a valvei  $RB_1$  pentru  $u_1(t) = u_{10}$  (a) și caracteristica elementului tripozițional (b)

Figura 3.5.7a ilustrează răspunsul la semnal treaptă a vaporizatorului în cazul teoretic  $h_1(t) = 1$ , pentru o temperatură inițială  $\theta_1(0) = 33.5^\circ\text{C}$ . Pentru aceeași condiție inițială, figura 3.5.7b ilustrează răspunsul la semnal treaptă a vaporizatorului în cazul real  $u_1(t) = u_{10}$  determinat pentru  $a(t) \geq 1.2\varepsilon_I$ . Datorită temporizării introduse de valvă, răspunsul este lent (în jur de 2.5 minute).

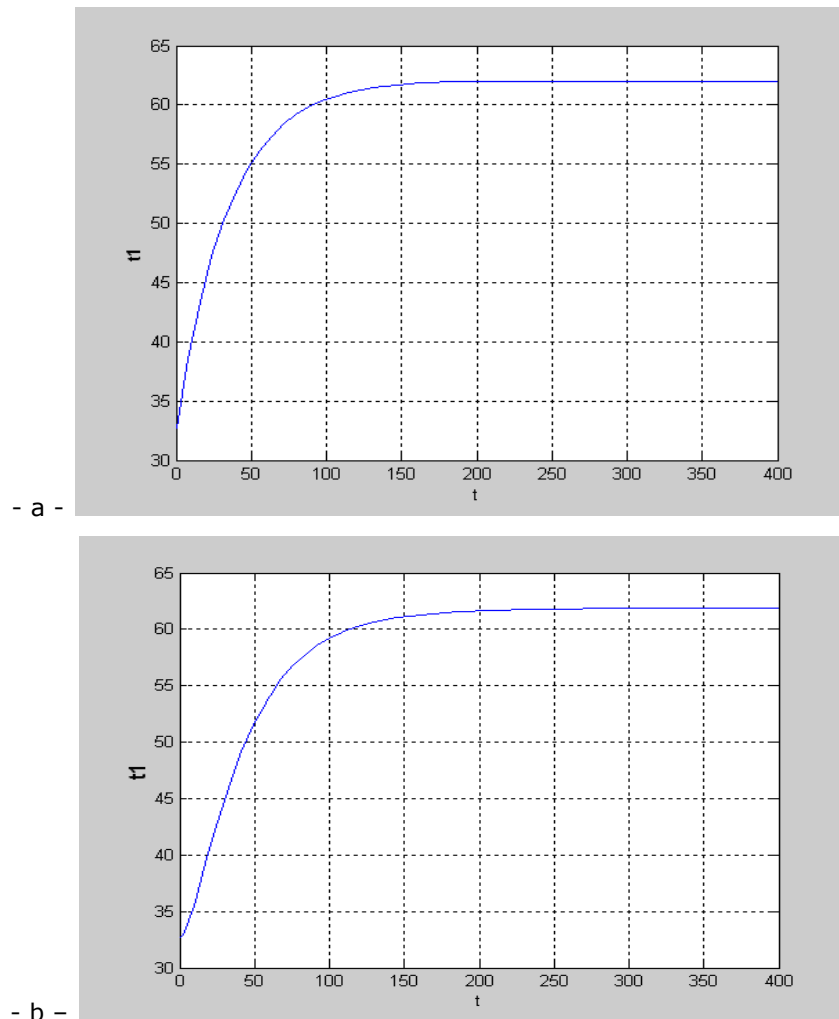
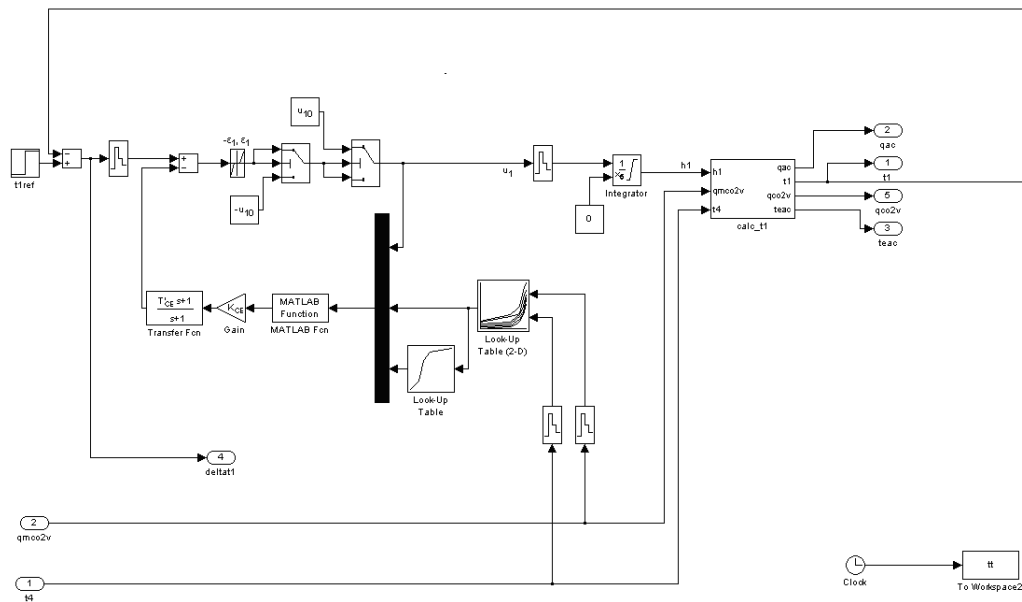


Fig. 3.5.7 Răspunsul la semnal treaptă  $\theta_1(t)$  al vaporizatorului (a. – pentru  $h_1(t)=1$ , b. – pentru  $u_1(t)=U_{10}$ ).

Pentru a determina valorile parametrilor  $K_{CE}$  și  $T'_{CE}$  a fost necesar să se ia în considerare: caracterul nelinear al vaporizatorului, panta limitată a semnalului de comandă  $h_1$ , zona de insensibilitate extinsă  $2\varepsilon_1 \leq 0.3$  a elementului tripozițional și faptul că regulatorul  $RA'_1$  este implementat numeric. Complexitatea problemei impune o abordare cvasi-empirică în proiectare specifică blocurilor interpolative. Pentru a realiza aceste obiective s-au parcurs trei pași:

- Determinarea valorilor maxime pentru parametrii  $K_{CE}$  și  $T'_{CE}$  pentru a asigura condițiile de funcționare pentru blocul CE. Utilizând schema Simulink care corespunde schemei bloc din figura 3.5.4, s-au stabilit valorile  $K_{CEmax}=0.005$  and  $T'_{CEmax}=40$  sec.
- Estimarea setului de valori nominale pentru parametrii  $K_{CE}$  și  $T'_{CE}$  utilizând schema de simulare pentru întreg sistemul de reglare din cadrul ciclului termodinamic:  $K_{CE}=0.001$ ,  $T'_{CE}=30$ sec.

- Stabilirea tabelelor de interpolare corespunzătoare parametrilor  $K_{CE}$  și  $T'_{CE}$  considerând pentru perturbațiile  $q_{CO2pv}$  și  $\theta_4$  domeniile  $[0, 70]$   $kg \cdot s^{-1}$ , respectiv  $[20, 31]^\circ C$ . Valorile de sprijin au fost determinate cu ajutorul algoritmilor genetici [118] utilizând schema Simulink din figura 3.5.8. Rezultatele finale ale proiectării sunt redată în tabelele de mai jos (tabelele 3.4 și 3.5).

Fig. 3.5.8 Schema Simulink utilizată pentru a determina parametrii  $K_{CE}$  și  $T'_{CE}$ Tabelul 3.4 - Tabelul de interpolare pentru parametrul  $K_{CE}$ 

$q_{CO2_{pv}}$ [ $kg \cdot s^{-1}$ ] \ $\theta_4$ [ $^\circ C$ ]	20	28.5	29.3	30	30.5	31
0	x	0.0001	x	0.0003	x	x
45	x	x	x	x	x	x
55	x	0.0004	x	x	0.0009	x
60	x	0.0006	x	x	0.001	0.0015
65	0.0009	x	x	x	0.002	0.003
70	0.0016	0.0019	0.002	0.0022	0.0025	0.0035

 $K_{CE}$

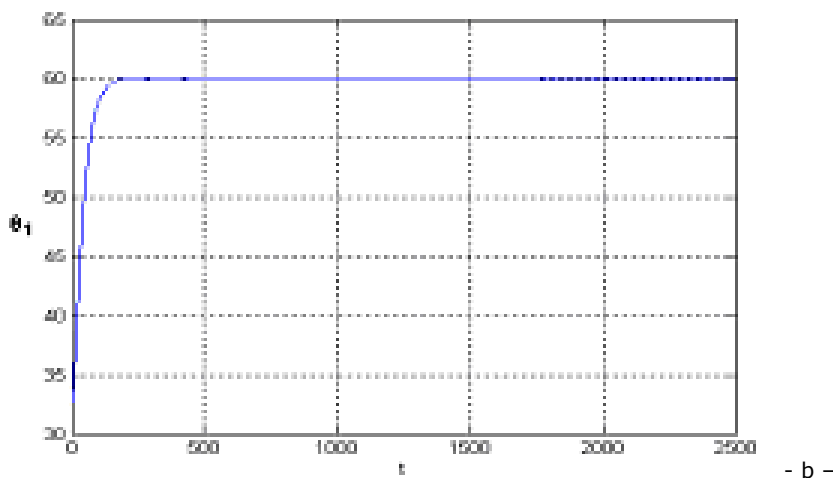
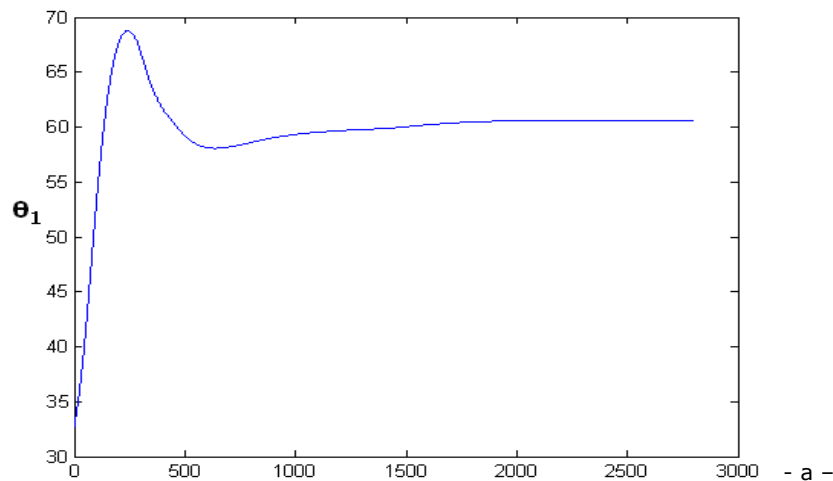


Tabelul 3.5 - Tabel de interpolare pentru parametrul  $T'_{CE}$ 

$K_{CE}$	0	0.0001	0.0004	0.0007	0.0009	0.001	0.0015	0.003	0.0035
$T'_{CE}$	23	23	25	27	28.5	30	32	34	35

### 3.5.3.3 Studiul comportării sistemului de reglare proiectat

Figura 3.5.9 ilustrează cu privire la mărimea de reglare  $\theta_1$  comportarea sistemului înainte de utilizarea noii soluții de reglare (a) și după aplicarea acesteia (b, c), pentru o valoare a temperaturii de referință  $\theta_1 = 60^\circ\text{C}$ , iar figurile 3.5.10 și 3.5.11 evoluția lui  $h_1$  și  $u_1$  pe durata procesului tranzitoriu. Se poate observa că suprareglajul a fost limitat. Raspunsul din figura 3.5.9 b indică un suprareglaj în jur de doar 1% și o precizie în regim staționar de cca. 0.1%. De asemenea trebuie menționat faptul că valoarea temperaturii  $\theta_1$  atinge zona de insensibilitate,  $60 \pm 0.15^\circ\text{C}$ , în aprox. 300 sec = 5 minute.



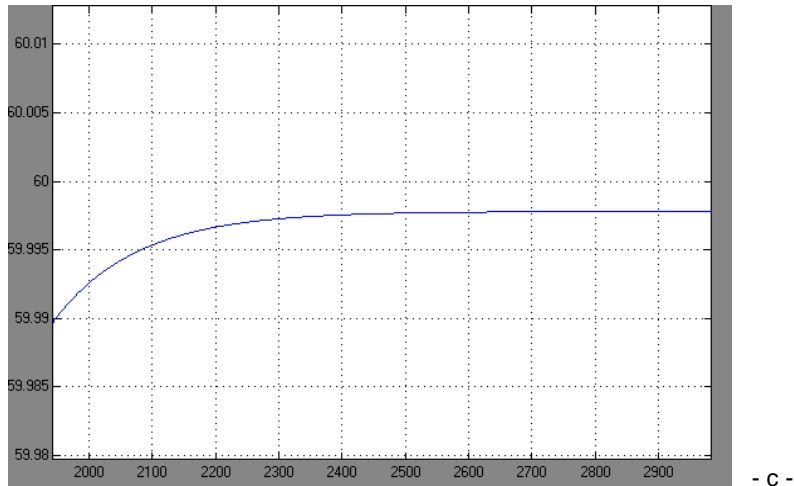


Fig. 3.5.9. Răspunsul  $\theta_1(t)$  al vaporizatorului pentru o valoare referință  $\theta_1 = 60^\circ\text{C}$ , pentru  $K_{CE}=0.001 \text{ kg}\cdot\text{s}^{-1}$ ,  $T'_{CE} = 30 \text{ s}$ ,  $q_{\text{CO}_2\text{-pv}} = 60 \text{ kg}\cdot\text{s}^{-1}$ ,  $\theta_4=30.5 \text{ }^\circ\text{C}$ ,  $\theta_1(0)=33.5 \text{ }^\circ\text{C}$ . (a-utilizând structura veche de reglare, b-utilizând noua structură de reglare, c-reprezentare detaliată pentru noua structură de reglare)

Se poate observa frecvența joasă de comutare în regimul tranzitoriu și faptul că  $u_1$  oscilează cu o perioadă de aproximativ 1 minut, conduce la o stabilizare completă în 30 de minute. Se poate spune că uzura valvei este mai mică decât în situația în care reglarea se face în timp continuu.

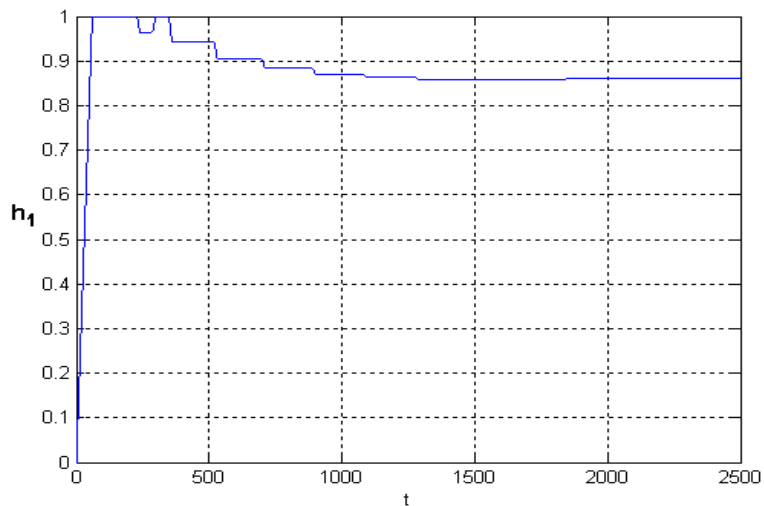


Fig. 3.5.10 Evoluția deschiderii  $h_1$  a valvei  $RB_1$ , în condiții identice ca în figura 3.5.9

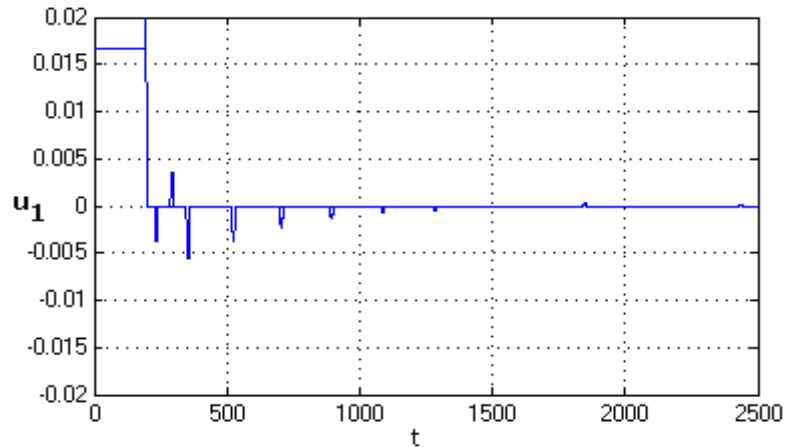


Fig. 3.5.11 Evoluția mărimii de comandă  $u_1$  a valvei  $RB_1$  în condiții identice ca în figura 3.5.9

În cadrul acestui subcapitol s-au dezvoltat aspecte referitoare la sistemul de reglare al temperaturii la ieșirea din blocul vaporizatoare din cadrul centralei electrice geotermale. Era necesară reglarea în doar două puncte, unul în blocul vaporizatoare și celalalt în blocul condensatoare [2], [46]. Prin reglarea temperaturii menționate se realizează stabilizarea unuia dintre cele patru puncte ale ciclului termodinamic al  $CO_2$ , utilizat pentru a opera asupra turbinei centralei electrice geotermale.

Noua structură de reglare utilizează un regulator tripozițional cu histerezis cu o buclă de reglare adaptivă realizată utilizând blocuri interpolative (vezi subcapitolul 3.4) corespunzător principalelor perturbații: debitul masic al  $CO_2$  și temperatura  $CO_2$  la intrarea în vaporizatoare. Noua versiune de reglare, implementată numeric, bazată pe rezultate obținute prin simulare, este mai bună decât cea actuală atât în regim staționar cât și în regim dinamic.

Rezultatele cercetării din cadrul acestui subcapitol au fost diseminate prin lucrarea [65].

### 3.6 Creșterea gradului de disponibilitate în cadrul blocului vaporizatoare

Acest subcapitol se axează pe creșterea disponibilității în cadrul blocului vaporizatoare din centrala electrică geotermală. Potrivit celor prezentate în capitolul curent, abordările referitoare la conceptul de disponibilitate sunt variate, începând cu funcționarea și eficiența proceselor tehnologice până la structurile de reducere a numărului de defecte sau opriri accidentale. Calitatea unui sistem (aspecte referitoare la calitate au fost prezentate în cadrul subcapitolului 1.5) este estimată pe baza abilității acestuia de a își îndeplini sarcinile în funcție de cerințele utilizatorului.

Centrala electrică geotermală reprezintă un sistem automat de o complexitate ridicată conținând mai mult de o buclă de reglare (reglarea sistemelor automate de acest tip prezintă un grad ridicat de complexitate [116], [117]), toate aceste bucle interacționând [46]. Întreg ansamblul poate fi controlat utilizând un automat programabil [79]. Considerând faptul că buclele de reglare utilizate pentru

controlul ciclului termodinamic al agentului intermediar (utilizat pentru conversia energiei termice în energie electrică) nu sunt saturate, acestea au un grad ridicat de separabilitate. Dacă apar unele saturații în procesul de reglare, pentru a menține eficiența ciclului termodinamic este necesar controlul debitului masic al agentului intermediar. Această acțiune implică interacțiunea buclelor de reglare existente. Indiferent de tipul buclei de reglare, dacă bucla curentă funcționează mai sigur atunci crește disponibilitatea întregului sistem [47].

În acest context subcapitolul prezintă o singură buclă de reglare a sistemului de reglare a centralei electrice geotermale, bucla de reglare a temperaturii  $\theta_1$  de la ieșirea blocului vaporizatoare, care prin faptul că reprezintă o parte esențială a centralei electrice geotermale realizează transferul energiei termice a apei geotermale agentului intermediar ( $\text{CO}_2$ ) [46].

Structura de reglare dezvoltată în cadrul subcapitolului 3.5 influențează în mod pozitiv disponibilitatea din perspectiva unei reglări eficiente și sigure. Structura redundantă propusă în cadrul subcapitolului curent extinde structura de reglare din subcapitolul 3.5, introducând elemente suplimentare de creștere a disponibilității prin utilizarea unui sistem redundant pentru măsurarea temperaturii la ieșirea din blocul vaporizatoare. Sistemul de măsurare folosește o logică decizională bazată pe temperatura de la ieșirea din vaporizatoare [46].

### 3.6.1 Creșterea disponibilității prin redundanță a structurii de reglare a temperaturii $\theta_1$ de la ieșirea blocului vaporizatoare

Structura propusă pentru reglarea temperaturii  $\theta_1$  este ilustrată în figura 3.6.1. Diferențele față de structura din subcapitolul anterior sunt detaliate în figurile 3.6.2 și 3.6.3.

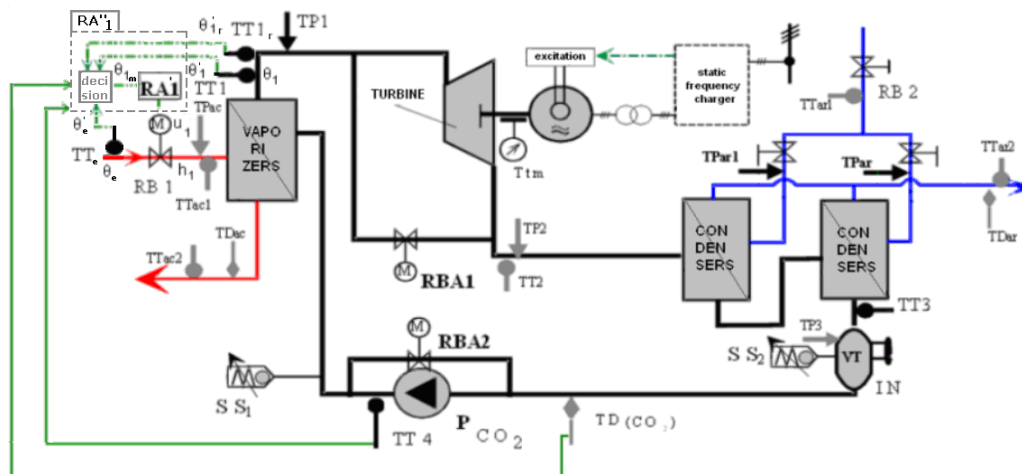


Fig. 3.6.1 Poziția structurii propuse de reglare a temperaturii  $\theta_1$

Creșterea disponibilității se realizează prin structura redundantă utilizată în cadrul regulatorului  $\text{RA}'_1$ . Caracterul redundant este obținut la nivelul traductoarelor utilizate pentru măsurarea temperaturii  $\text{CO}_2$  de la ieșirea blocului vaporizatoare.

Sistemul este controlat utilizând regulatorul RA''<sub>1</sub> care funcționează pe baza informațiilor furnizate de traductoarele de temperatură TT<sub>1</sub> și TT<sub>1r</sub>, utilizate pentru a măsura temperatura  $\theta_1$  și de traductorul de temperatură TTe, utilizat pentru a măsura temperatura apei calde la ieșirea din blocul vaporizatoare. Sistemul de măsurare conține, de asemenea, un sistem de decizie și comutare furnizând semnalul de pe reacție pentru a regla temperatura  $\theta_1$  (ex. pentru comutarea semnalului de pe reacție de la un traductor la altul).

Regulatorul propriu-zis RA'1, parte a regulatorului RA''1, are structura prezentată în figura 3.6.3. Soluția a fost detaliată în subcapitolul 3.5 și constă într-un regulator tripozițional cu buclă dinamică de reacție. Caracterul dinamic al buclei este adaptat în manieră interpolativă. Blocul vaporizatoare este controlat prin deschiderea  $h_1$  a valvei din ansamblul M1+RB1 (motor + valvă). Elementul de noutate îl reprezintă încorporarea structurii redundante din figura 3.6.2 în structura din figura 3.6.3.

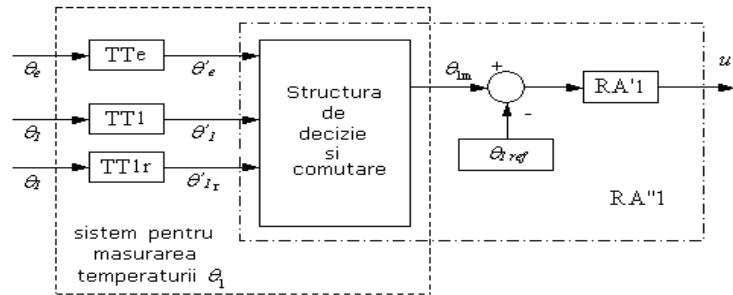


Fig. 3.6.2 Structura sistemului de măsură a temperaturii  $\theta_1$  relativ la regulatorul de temperatură

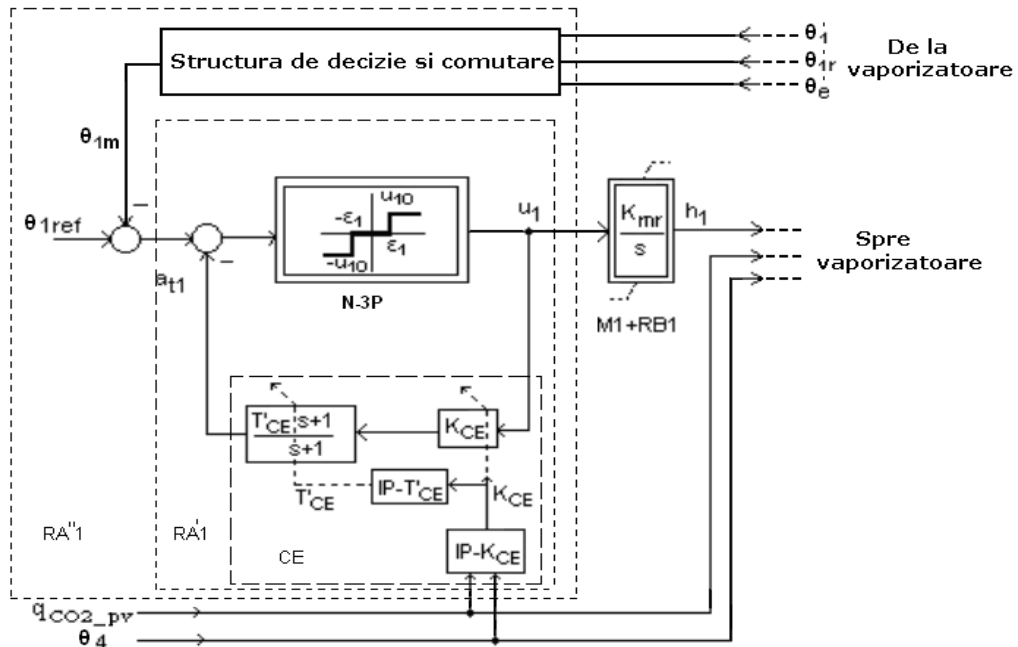


Fig. 3.6.3 Structura modificată a buclei de reglare a temperaturii  $\theta_1$  din cadrul centralei electrice geotermale

Blocul de decizie și comutare din figura 3.6.2 se utilizează pentru a crește gradul de disponibilitate al sistemului de reglare și implicit al întregului sistem. Ieșirea  $\theta_{1m}$  a sistemului de decizie depinde de informațiile furnizate de cele două traductoare care măsoară temperatura  $\theta_1$  (traductorul  $TT_1$  și traductorul de rezervă  $TT_{1r}$ ) și de traductorul folosit pentru măsurarea temperaturii apei geotermale  $\theta_e$  de la ieșirea blocului vaporizatoare (traductorul TTe).

Datorită caracterului nelinear al blocului vaporizatoare, regulile valabile pentru o bună funcționare a sistemului de decizie și comutare au fost stabilite prin simulări care au condus la identificarea diverselor corelații între ieșirile furnizate de fiecare dintre cele trei traductoare. Acestea au condus la stabilirea diferitelor regimuri de funcționare eronate [46]. Pe această bază au fost stabilite următoarele două reguli:

*Regula R1\_1:*

*Dacă ( $|sumTT1r| > deltasumTT1r$  și  $sumteac < deltateac$ ) atunci (transmite semnal - traductor de rezervă defect)*

*Regula R1\_2:*

*Dacă ( $sumTT1r \cdot sumteac > 0$  și  $|sumTT1r| > gdeltasumTT1r$  și  $|sumteac| > gdeltateac$ ) atunci (comută pe traductorul de rezervă și transmite semnal - traductor principal defect)*

Variabilele menționate în aceste două reguli sunt:  $sumTT1r$  - suma variațiilor ieșirii traductorului de rezervă,  $deltasumTT1r$  - limită pentru  $sumTT1r$ ,  $sumteac$  - suma variațiilor temperaturii indicate de traductorul Teac,  $deltateac$  - limită pentru  $sumteac$ ,  $gdeltasumTT1r$  - valoare limită pentru  $sumTT1r$ ,  $gdeltasumteac$  - valoare limită pentru  $sumteac$ .

Algoritmul utilizează și o a treia regulă implicită:

*Regula R1\_3 :*

*Dacă (regulile R1\_1 și R1\_2 nu sunt activate) atunci (valoarea temperaturii în buclă rămâne neschimbată)*

Valorile lui  $sumTT1r$  reprezintă suma variațiilor valorii medii calculate pentru grupuri de cinci valori consecutive (corespunzând la cinci citiri consecutive) ale temperaturii măsurate de către traductorul de rezervă  $TT_{1r}$ . Prima variație este considerată raportat la prima valoare măsurată după activarea programului folosit pentru măsurarea redundantă a temperaturii  $\theta_1$ . Suma variațiilor se consideră diferența dintre valoarea medie curentă și valoarea prescrisă [97]. S-a păstrat ideea de sumă pentru a utiliza variațiile valorilor medii de la un grup de eşantioane la altul pentru semnale de protecție și alarmă.

Platforma de implementare a algoritmului de decizie este un automat programabil Allen Bradley utilizat pentru controlul centralei electrice geotermale, luându-se în considerare faptul că automatul funcționează în timp real.

### 3.6.2 Performanțele structurii propuse pentru reglarea temperaturii $\theta_1$ din perspectiva disponibilității

Performanțele regulatorului propriu-zis al sistemului de reglare au fost ilustrate în cadrul capitolului 3.5.

Pentru a sublinia modul de acțiune al sistemului de măsură în cadrul sistemului de reglare și pentru a-i ilustra eficiența s-au imaginat diverse scenarii de simulare. În aceste scenarii s-a considerat că traductoarele de temperatură sunt sisteme liniare de ordin 1, având funcția de transfer de forma  $\beta K / (Ts + 1)$ , unde T

reprezintă constanta de timp a traductorului, și coeficientul  $\beta$  servește pentru a modela situații reale în care factorul de amplificare al traductorului diferă de valoarea lui nominală  $K$ .

Elementele studiate în cadrul acestui subcapitol au constituit subiectul celor prezentate în [45], lucrare la care a colaborat și autorul acestei teze. Din [46] se rețin figurile 3.6.4, 3.6.5 și 3.6.6 care constituie rezultatele obținute în urma scenariilor de simulare.

Primul scenariu corespunde situației în care traductorul de rezervă  $TT_{1r}$  se defectează, iar traductorul principal  $TT_1$  rămâne funcțional. Situația este prezentată în figura 3.6.4.

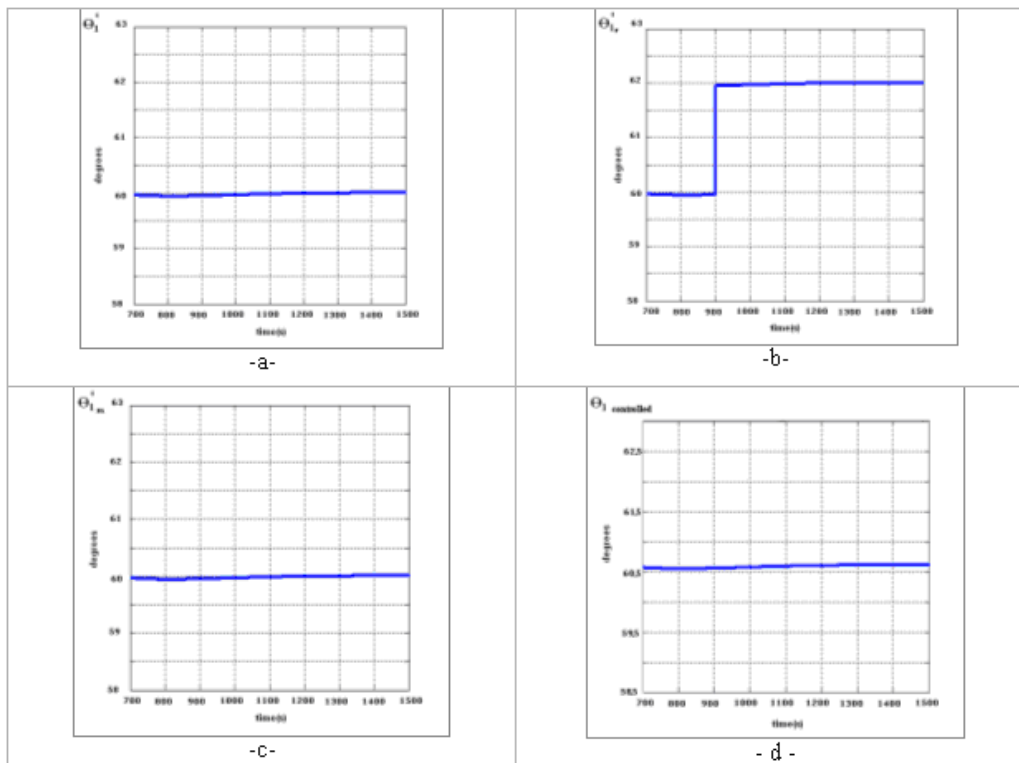


Fig. 3.6.4 Valorile temperaturilor măsurate de traductorul principal,  $\theta_1$  (a), de traductorul de rezervă,  $\theta_{1r}$  (b), semnalul de pe reacție utilizat în cadrul reglării temperaturii,  $\theta_{1m}$ (c) și temperatura reglată,  $\theta_i$  (d), pentru scenariul  $TT_1$  funcțional,  $TT_{1r}$  defect.

Defecțiunea apare la  $t = 900$  secunde când la nivelul ieșirii traductorului  $TT_{1r}$  se constată o variație pozitivă de două grade. În acel moment sistemul funcționa în regim staționar. Blocul de măsurare rămâne insensibil la acest tip de defect iar în consecință temperatura reală  $\theta_1$  rămâne constantă.

Al doilea scenariu consideră situația în care traductorul  $TT_1$  se defectează, traductorul de rezervă rămâne funcțional, iar blocul de decizie și comutare va comuta automat pe traductorul de rezervă  $TT_{1r}$ . Rezultatele sunt prezentate în figura 3.6.5. Defectul apare la momentul  $t=900$  secunde, când ieșirea traductorului  $TT_1$  variază cu două grade în sens negativ. Blocul de măsurare reacționează după 7.5 secunde și practic sistemul nu reacționează la defectul apărut (oscilațiile între  $60.5^{\circ}\text{C} - 60.65^{\circ}\text{C}$  sunt irelevante).

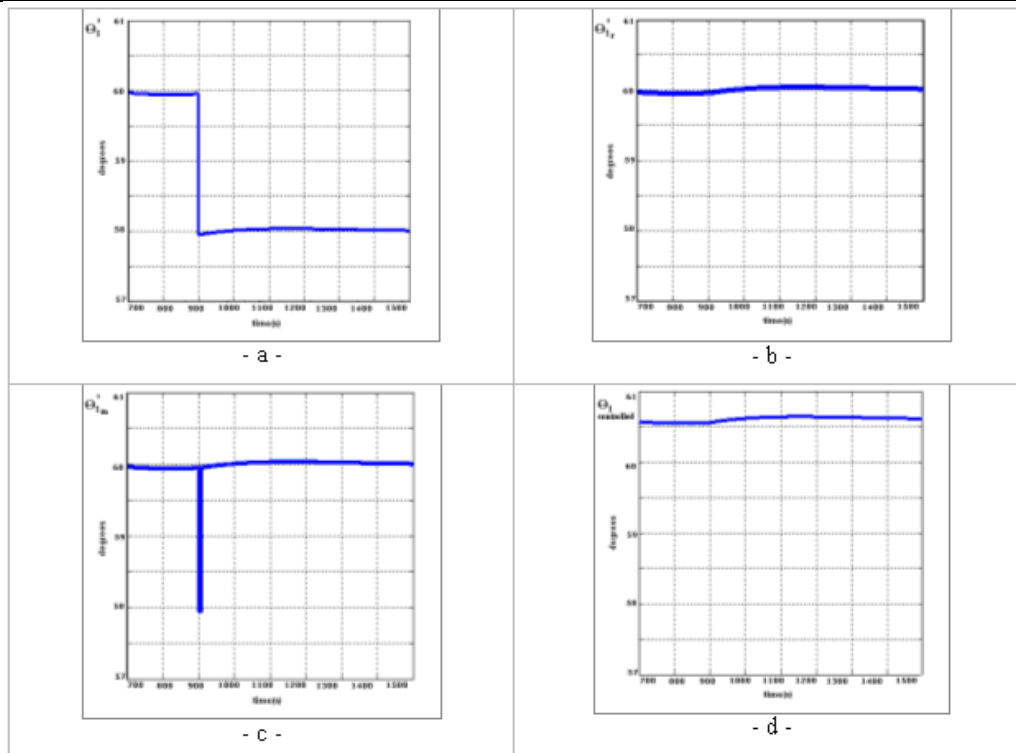


Fig. 3.6.5 Valorile temperaturilor măsurate de traductorul principal  $\theta'_1$  (a), de traductorul de rezervă  $\theta'_{1r}$  (b), semnalul de pe reacție utilizat pentru reglarea temperaturii  $\theta'_{1m}$  (c) și temperatura reglată  $\theta_t$  (d), pentru scenariul *TT1 defect, TT1r funcțional*.

Variația din figura 3.6.5 c este detaliată în figura 3.6.6. Și în acest caz bucla de reglare rămâne insensibilă la defetul apărut la traductorul funcțional precum și la procesul de comutare.

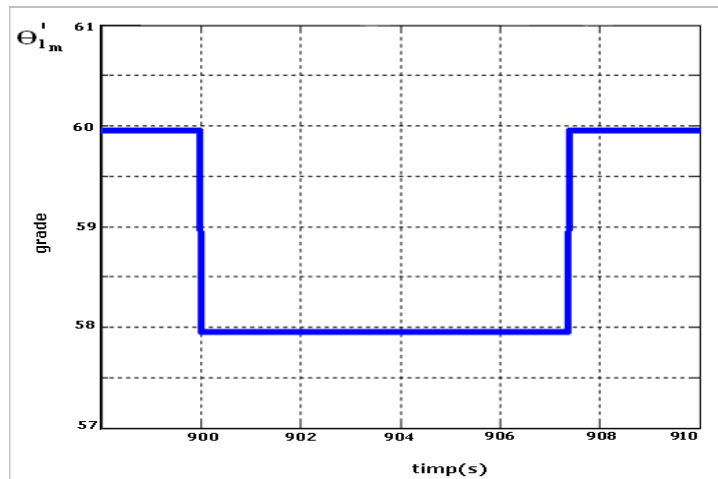


Fig. 3.6.6 Variația detaliată a ieșirii blocului de măsurare a temperaturii  $\theta_1$  corespunzătoare scenariului al doilea



### **3.7 Modul mobil de detecție și diagnoză a defectelor pentru sisteme automate**

Detecția și diagnoza defectelor este o direcție actuală de cercetare pentru a crește siguranța în funcționare a sistemelor automate și pentru a obține toleranță la defecte [20], [98]. Detecția automată a defectelor este capabilă de a reduce timpul de diagnoză, respectiv prin acesta timpul de defectare și în consecință de a crește disponibilitatea [14].

Din rațiuni financiare și datorită limitărilor tehnice, se recomandă să se realizeze diagnoza automată utilizând resursele hardware și software curente. De asemenea, datorită răspândirii largi a PLC-urilor, se pot utiliza echipamente de detecție care pot fi discontinuu conectate la PLC-urile care sunt utilizate pentru controlul proceselor dintr-o arie extinsă. Acest lucru simplifică standardizarea aplicațiilor și facilitează depanarea unei varietăți de procese cu ajutorul unui singur echipament.

Cercetarea s-a axat în principal pe sistemele automate care sunt deja implementate și în stare de funcționare. Defectele acestora sunt uneori ușor de detectat din comportamentul sistemului, dar alteori ele sunt posibil de detectat doar printr-o investigație amănunțită a evoluției sistemului prin corelarea diferitelor simptome. Identificarea acestor corelații necesită o investigație matematică rafinată, realizată online sau offline.

Detecția și diagnoza defectelor se poate realiza prin utilizarea de metode bazate pe modele matematice, precum redundanța analitică, sau într-o manieră diferită, prin realizarea de dependențe corelaționale pentru identificarea diverselor situații. Tehnicile de investigație analitice ale defectelor reprezintă un domeniu de cercetare actual [101], [53], [55] care converge într-un mod evident cu proiectarea structurilor cu redundanță analitică [74], [100]. În acest context autorul propune un modul mobil de detecție și diagnoză a defectelor (MFDDM) pentru a identifica defecte din cadrul centralei electrice geotermale. Este un modul independent, acordat pe proces, fără a influența comportamentul sistemului. Modul său de acțiune este specific acestei aplicații, dar conceptul este unul general. Toate corelațiile obținute în urma numeroaselor simulări realizate pe modelul matematic al centralei electrice geotermale [2], [46] sunt descrise în cadrul cercetării doar printr-un raționament strict calitativ, simptomele fiind posibil de identificat în regimul staționar.

În cadrul subcapitolului, prezentarea începe cu descrierea unor elemente legate de centrala electrică geotermală pentru a putea explica algoritmul de detecție și diagnoză a defectelor. Urmează implementarea MFDDM și prezentarea unor experimente pentru a-i evidenția eficiența.

#### **3.7.1 Algoritmul de detecție și diagnoză a defectelor**

Posibilitatea creșterii siguranței în funcționare prin detecție și diagnoză a defectelor bazată pe dependențe corelaționale va fi evidențiată pe un sistem automat. Sistemul automat considerat este tot centrala electrică geotermală prezentată în subcapitolul 3.1 (vezi [65]).

În cadrul ciclului termodinamic al CO<sub>2</sub>,  $\theta_1$  și  $\theta_3$  sunt reglate în bucle închise în blocurile vaporizatoare și condensatoare pentru a obține valorile nominale  $\theta_{1_{nom}}$  și  $\theta_{3_{nom}}$ . Întregul proces este susținut de controlul la un nivel superior al CO<sub>2</sub> realizat de pompă.

Algoritmii de detecție și diagnoză a defectelor se bazează pe dependențe corelaționale obținute prin simulări asupra modelului analitic al centralei electrice geotermale. Ideea este generală în sensul că aceste tipuri de algoritmi au ca punct de plecare studiul comportamentului sistemului ținând în diverse regimuri anormale, pentru a putea elabora logica de detecție corectă.

Comportamentul anormal al centralei electrice geotermale în cazul apariției unui defect se poate observa și analiza monitorizând diverse puncte cheie asociate unor elemente sau procese critice (care pot produce defecte). În cazul de față algoritmul propus de autor este destinat să furnizeze detecția și diagnoza defectelor apărute la nivelul senzorilor TT<sub>1</sub> și TT<sub>3</sub>, și la valvele RB<sub>1</sub> și RB<sub>3</sub>, identificate ca puncte critice în sistem. Defectele la nivelul acestor elemente pot influența în mod distinct valorile unor variabile. O cercetare detaliată bazată pe simulări efectuate pe modele matematice precum și pe analiză practică a condițiilor tehnice a arătat că tipurile de defecte considerate pot fi identificate analizând valorile următoarelor mărimi: temperaturile  $\theta_{ac}$ ,  $\theta_{ar}$ ,  $\theta_1$ ,  $\theta_3$ , semnalele de comandă  $u_1$  și  $u_3$ , măsurabile în diverse puncte ale centralei geotermale.

În cadrul următoarelor scenarii sunt prezentate pe scurt (tabelar) tipurile de defecte care pot să apară (simptomele sunt marcate cu fond închis), precum și dependențele corelaționale în funcție de posibilitățile de detecție.

Pentru a simplifica prezentarea, au fost introduse următoarele simboluri:

*nom* - indice pentru valoarea nominală a mărimii;

↗ - valoarea actuală a mărimii este mai mare decât valoarea nominală;

↘ - valoarea actuală a mărimii este mai mică decât valoarea nominală;

max, min - indici pentru valoarea maximă și minimă admisibilă pentru temperaturile  $\theta_{ac}$ , respectiv  $\theta_{ar}$  când valvele RB<sub>1</sub>, respectiv RB<sub>2</sub> sunt complet deschise.

*Scenariul 1: Caracteristica intrare-ieșire a senzorului de temperatură TT<sub>1</sub> suferă deviații mari în sens negativ (NL) de la valoarea nominală ca în fig. 3.7.1a, determinând pompa de CO<sub>2</sub> să acționeze. Linia continuă reprezintă caracteristica nominală, iar linia întreruptă reprezintă caracteristica afectată de defect.*

Bucula de reglare a temperaturii  $\theta_1$  este ilustrată în fig. 3.7.1b (această problemă este detaliată în [65], precum și în subcapitolul 3.5). Mai jos, în tabel, se prezintă tipul de defect identificat în funcție de simptomele care sunt ilustrate pe fond închis:

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$U_1$	$u_3$
$\geq \theta_{ac_{max}}$	↘	O.K.	↗	O.K.	O.K.

⇓

Concluzie: Defect de tipul TT<sub>1</sub>-NL

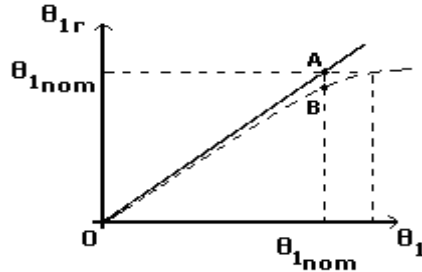


Fig. 3.7.1a Caracteristica senzoruului TT<sub>1</sub>

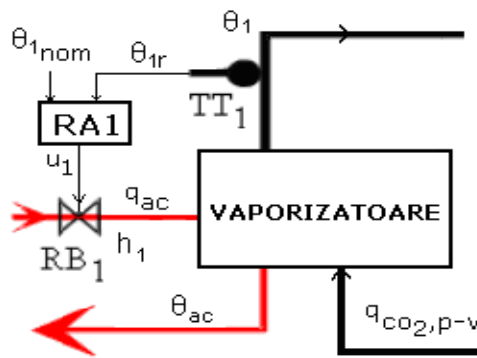


Fig. 3.7.1b Bucla de reglare a temperaturii  $\theta_1$

*Discuție* (explicația celui de al doilea rând al tabelului):

Datorită deviației caracteristicii senzoruului, valoarea nominală a semnalului de pe reacție este obținută pentru  $\theta_1 > \theta_{1nom}$ . Pentru a putea elimina eroarea de reglare (segmentul AB), inițial robinetul RB<sub>1</sub> se deschide la maxim rezultând  $h_1 = \max$  și  $\theta_{ac} \geq \theta_{acmax}$ ; apoi pompa de CO<sub>2</sub> intră în funcțiune (realizând controlul la un nivel superior)  $q_{CO_2, p-v} \searrow$  până când  $AB \cong 0$ . Debitul  $q_{CO_2, p-v} \searrow \Rightarrow$  un debit  $q_{CO_2}$  mai mic și  $\theta_1 > \theta_{1nom}$  în grupul turbină-generator  $\Rightarrow \theta_2 \nearrow, q_{CO_2, t-c} \searrow$ . Efectul este că  $\theta_{ar} \searrow$ . În timpul întregului proces  $\theta_3$  și  $u_3$  au valori normale.

*Notă 1:* Ultimul efect poate fi evidențiat plecând de la următoarele două egalități din modelul condensatorului [46]:

$$\begin{cases} \theta_3 = q_{CO_2, t-c} \cdot (\theta_2 - \theta_3) - \frac{(\theta_2 + \theta_3) - (\theta_{iar} + \theta_{ar})}{2 \cdot R_2 \cdot c_{CO_2}^{med}} \\ \theta_{ar} = q_{ar} \cdot (-\theta_{ar} + \theta_{iar}) = \frac{(\theta_2 + \theta_3) - (\theta_{iar} + \theta_{ar})}{2 \cdot R_2 \cdot c_a} \end{cases}$$

din care rezultă că în regim permanent staționar avem:

$$\theta_{ar} = \theta_{iar} + (1 - q_{CO_2, t-c} \cdot c) \theta_2 + (1 + q_{CO_2, t-c} \cdot c) \theta_3 \tag{23}$$

și avînd în vedere că:  $c = \text{constant}$ , temperatura apei reci la intrarea în condensatoare  $\theta_{iar} = \text{constantă}$ , produsul  $(1 - q_{CO_2, t-c} \cdot c) \cdot \theta_2 < 0$  și crescut în

valoare absolută (observații experimentale),  $\theta_3 = \text{constantă}$  datorită buclei de reglare a temperaturii  $\theta_3$ , iar suma  $1 + q_{CO_2, t-c} \cdot c > 0$  și scăzută.

*Notă 2:* Datorită faptului că  $RA_1$  este un regulator PI implementat sub forma unui regulator tripozițional cu histerzis și cu reacție, când  $h_1 = \text{max}$ , motorul valvei face față fără probleme semnalului de comandă [65].

*Scenariul 2: Valva  $RB_1$  este blocată în poziția complet deschis (FOC).*

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$U_3$
$\geq \theta_{ac \text{ max}}$	$\searrow$	O.K.	O.K.	O.K.	O.K.



Concluzie: Defect de tip  $RB_1$ -FOC

*Discuție* (explicații pentru al doilea rând al tabelului):

În momentul în care se produce blocajul, la început temperatura  $\theta_1$  crește, urmând ca atâta timp cât  $h_1 = \text{max}$  acțiunea regulatorului  $RA_1$  este blocată. Semnalul de comandă  $u_1$  este de valoare negativă ( $RA_1$  încearcă să închidă valva) și  $\theta_{ac} \geq \theta_{ac \text{ max}}$ . În consecință pompa de  $CO_2$  acționează, așadar  $q_{CO_2, p-v} \nearrow$ . Intervenția pompei este o acțiune redundantă în reglarea temperaturii  $\theta_1$ , rezultând  $\theta_1 = \theta_{1 \text{ nom}}$  și de asemenea  $u_1 = 0$ . În ceea ce privește debitul de  $CO_2$ ,  $q_{CO_2, p-v} \nearrow \Rightarrow q_{CO_2, t-c} \nearrow$ . Efectul este că  $q_{ar} \nearrow$ , dar  $\theta_{ar} \searrow$ . La finalul procesului  $\theta_3$  și  $u_3$  au valori normale.

*Scenariul 3: Senzorul de temperatură  $TT_1$  se defectează astfel încât caracteristica acestuia este opusă (deviație pozitivă P) caracteristicii din fig. 3.7.1a.*

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$U_3$
$\searrow$	$\nearrow$	O.K.	$\searrow$	O.K.	O.K.



Concluzie: Defect de tip  $TT_1$ -P

*Discuție:* Deviația caracteristicii senzorului determină obținerea valorii nominale a semnalului de pe reacție pentru  $\theta_1 < \theta_{1 \text{ nom}}$ . Pentru a elimina eroarea de reglare  $h_1 \searrow \Rightarrow q_{ac} \searrow$ , rezultând  $\theta_{ac} \searrow$  și  $\theta_1 \searrow$ . În final  $\theta_{1r} = \theta_{1 \text{ nom}}$  și  $u_1 = 0$ . Valoarea scăzută a temperaturii  $\theta_1$  determină  $\theta_2 \searrow$ . În consecință  $\theta_{ar} \nearrow$ . În timpul întregului proces  $\theta_3$  și  $u_3$  au valori normale.

*Scenariul 4: Valva  $RB_1$  se blochează în poziția închis (FC) (blocată într-o poziție sub valoarea nominală).*

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
$\searrow$	$\nearrow$	O.K.	$\searrow$	positive	O.K.



Concluzie: Defect de tip  $RB_1$ -FC

*Discuție:* Blocarea valvei  $RB_1$  determină  $q_{ac} \searrow \Rightarrow \theta_{ac} \searrow$ . Blocarea cauzează permanent  $\theta_1 < \theta_{1nom}$ , o eroare de reglare pozitivă și o valoare pozitivă pentru  $u_1$  (Regulatorul  $RA_1$  încearcă să acționeze asupra motorului valvei).  $\theta_1 \searrow$  determină  $\theta_2 \searrow$ , în consecință  $\theta_{ar} \nearrow$ . Pe parcursul procesului  $\theta_3$  și  $u_3$  au valori normale.

*Scenariul 5: Senzorul de temperatură  $TT_1$  suferă deviații negative de amplitudine redusă (NS), ca și în figura 3.7.2a, fără să determine punerea în funcțiune a pompei de  $CO_2$ .*

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
$\nearrow$	$\searrow$	O.K.	$\nearrow$	O.K.	O.K.



Concluzie: Defect de tip  $TT_1$ -NS

*Discuție:* La început  $\theta_{1r} < \theta_{1nom}$ . Pentru a elimina eroarea de reglare  $h_1 \nearrow$  dar  $h_1 \neq \max$ . Valoarea crescută a debitului de apă caldă  $q_{ac}$  determină  $\theta_{ac} \nearrow$  și  $\theta_1 \nearrow$ . În final  $\theta_{1r} = \theta_{1nom}$  și în consecință valoarea mărimii  $u_1$  este 0. Valoarea crescută a temperaturii  $\theta_1$  și un debit  $q_{CO_2, p-v}$  constant, determină un  $\theta_2 \nearrow$  și un debit  $q_{CO_2, t-c}$  constant. Efectul este  $\theta_{ar} \searrow$ . Pe parcursul procesului  $\theta_3$  și  $u_3$  au valori normale.

*Scenariul 6: Valva  $RB_1$  este blocată în poziția deschis (FO) (blocată într-o poziție deasupra valorii nominale dar nu într-o poziție complet deschis (N)).*

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
$\nearrow$	$\searrow$	O.K.	$\nearrow$	negative	O.K.



Concluzie: Defect de tip  $RB_1$ -FON

*Discuție:* Valva este defectă, blocată deschis, în consecință debitul apei calde  $q_{ac}$  este crescut, cauzând  $\theta_{ac} \nearrow$ . Blocarea cauzează valabilitatea în permanență a relației  $\theta_1 > \theta_{1nom}$ , o eroare de reglare negativă în bucla de reglare și în consecință mărirea  $u_1$  are o valoare negativă. Valoarea crescută a temperaturii  $\theta_1$  determină  $\theta_2 \nearrow$ , iar efectul este  $\theta_{ar} \searrow$ . La finalul procesului  $\theta_3$  și  $u_3$  au valori normale.

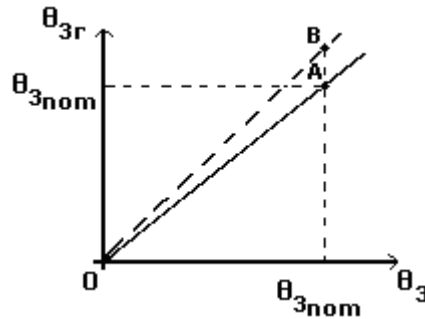
Următoarele șase scenarii prezintă defecte la nivelul blocului condensatoare, la senzorul de temperatură  $TT_3$  și la valva  $RB_3$ . Raționamentul este similar și în consecință sunt ilustrate doar rezultatele.

*Scenariul 7: Caracteristica intrare-ieșire a senzorului de temperatură  $TT_3$  suferă deviații pozitive mari (PL) de la valoarea nominală ca în fig. 3.7.2, determinând pompa de  $CO_2$  să acționeze.*

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
$\nearrow$	$\leq \theta_{ar \min}$	$\searrow$	O.K.	O.K.	O.K.



Concluzie: Defect de tip  $TT_3$ -PL

Fig. 3.7.2 Caracteristica senzorului  $TT_3$ 

Scenariul 8: Valva  $RB_3$  este blocată în poziția complet deschis (FOC).

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
↗	$\leq \theta_{ar\min}$	O.K.	O.K.	O.K.	negative



Concluzie: Defect de tip  $RB_3$ -FOC

Scenariul 9: Senzorul de temperatură  $TT_3$  se defectează și caracteristica intrare-ieșire a acestuia este opusă (negative deviations N) celei prezentate în figura 3.7.1a.

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
↘	↗	↗	O.K.	O.K.	O.K.



Concluzie: Defect de tip  $TT_3$ -N

Scenariul 10: Valva  $RB_3$  este blocată în poziția închis (FC) (blocată într-o poziție sub valoarea nominală).

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
↘	↗	↗	O.K.	O.K.	Positive



Concluzie: Defect de tip  $RB_3$ -FC

Scenariul 11: Caracteristica senzorului de temperatură  $TT_3$  suferă deviații pozitive de amplitudine redusă (PS) deviații ca în fig. 3.7.2, dar fără să cauzeze punerea în funcțiune a pompei de  $CO_2$ .

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
↗	↘	↘	O.K.	O.K.	O.K.



Concluzie: Defect de tip  $TT_3$ -PS

Scenariul 12: Valva  $RB_1$  este blocată în poziția deschis (FO) (blocată într-o poziție deasupra valorii nominale dar nu într-o poziție complet deschis (N)).

$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$
↗	↘	↘	O.K.	O.K.	negative



Concluzie: Defect de tip  $RB_3$ -FON

În tabelul 3.6 sunt sintetizate datele determinate în cadrul celor 12 scenarii. Se poate observa că cele 12 tipuri de defecte au simptome diferite și în consecință ele pot fi identificate pe baza acestor simptome. Mărimile cheie utilizate pentru a determina caracteristicile defectelor sunt accesibile, iar accesul și utilizarea lor nu influențează comportamentul sistemului.

Tabel 3.6

Simptome						Tipuri de defecte
$\theta_{ac}$	$\theta_{ar}$	$\theta_3$	$\theta_1$	$u_1$	$u_3$	
$\geq \theta_{ac \max}$	↘	O.K.	↗	O.K.	O.K.	$TT_1$ -NL
$\geq \theta_{ac \max}$	↘	O.K.	O.K.	O.K.	O.K.	$RB_1$ -FOC
↘	↗	O.K.	↘	O.K.	O.K.	$TT_1$ -P
↘	↗	O.K.	↘	positive	O.K.	$RB_1$ -FC
↗	↘	O.K.	↗	O.K.	O.K.	$TT_1$ -NS
↗	↘	O.K.	↗	negative	O.K.	$RB_1$ -FON
↗	$\leq \theta_{ar \min}$	↘	O.K.	O.K.	O.K.	$TT_3$ -PL
↗	$\leq \theta_{ar \min}$	O.K.	O.K.	O.K.	O.K.	$RB_3$ -FOC
↘	↗	↗	O.K.	O.K.	O.K.	$TT_3$ -N
↘	↗	↗	O.K.	O.K.	positive	$RB_3$ -FC
↗	↘	↘	O.K.	O.K.	O.K.	$TT_3$ -PS
↗	↘	↘	O.K.	O.K.	negative	$RB_3$ -FON

Tabelul 3.6 se utilizează în cele ce urmează pentru implementarea MFDDM în vederea identificării tipurilor de defecte pe baza dependențelor corelaționale determinate.

### 3.7.2 Implementarea MFDDM

Modulul mobil de detecție și diagnoză a defectelor (MFDDM) este un ansamblu care implementează informațiile prezentate în tabelul 3.6, considerat o schemă de detecție și diagnoză a defectelor pentru regimuri staționare. Desigur,

conceptul depinde de două categorii de elemente: procesul analizat și configurația hardware-software utilizată pentru detecție și diagnoză. În situația considerată, procesul monitorizat apare sub forma structurii și informațiilor din tabelul 3.6, împreună cu faptul că pentru măsurarea temperaturilor  $\theta_1$  și  $\theta_3$  sunt disponibili câte doi senzori și toate variabilele necesare sunt accesibile la terminalele PLC-ului care este utilizat pentru controlul sistemului din figura 3.1.2 din capitolul 3.1. Relativ la configurația hardware-software, se consideră că sunt utilizate următoarele elemente: placa de achiziție NI6036E, un laptop, modulul de distribuție CB-68LP și mediul LabView.

Structura MFDDM este ilustrată în fig. 3.7.3. Aici,  $\theta_1'$  și  $\theta_1''$  reprezintă ieșirile celor doi senzori de temperatură care măsoară temperatura  $\theta_1$ , unul legat la bucla de reglare (furnizează temperatura  $\theta_{1r}$  din fig. 3.7.1b) și unul disponibil pentru detecția defectelor (furnizează temperatura  $\theta_1$  din tabelul 3.6). Același lucru este valabil de asemenea pentru  $\theta_3'$  și  $\theta_3''$ .

Semnalele preluate de la PLC pot fi analogice sau digitale. În cele ce urmează se consideră numai prima situație. Placa de achiziție este conectată la laptop pe portul PCMCIA și preia semnalele pe canalele ach0 – ach6 utilizând modulul de distribuție.

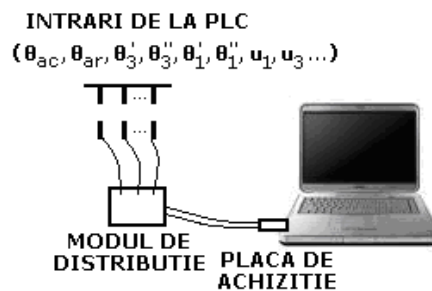


Fig. 3.7.3 Arhitectura MFDDM

În consecință, implementarea software a MFDDM este axată pe crearea unui instrument virtual, incluzând algoritmul de detecție și diagnoză a defectelor ca și în tabelul 3.6 și realizarea interfeței cu utilizatorul.

Achiziția de semnale se realizează la o frecvență de 2 kHz setată în cadrul blocului din figura 3.7.4.



Fig. 3.7.4 Bloc de achiziție de semnale

Relativ la tabelul 3.6, în crearea instrumentului virtual în LabView, unele situații trebuie programate. Acestea vor fi tratate în maniera următoare:

- $\nearrow, \searrow, O.K.$  (pentru  $\theta_{ac}$ ,  $\theta_{ar}$ ,  $\theta_3$  și  $\theta_1$ ): necesită o valoare nominală, un interval de toleranță și operații de comparare a valorii măsurate cu limitele intervalului de toleranță;
- pozitiv, negativ (pentru  $u_1$  și  $u_2$ ): se referă la caracterul tripozițional al semnalelor  $u_1$  și  $u_3$ . De exemplu, fig. 3.7.5a prezintă un segment din



evoluția în timp a semnalului  $u_1$ , ilustrând tranziția de la nivelul 0 (O.K.) la nivelul pozitiv, și fig. 3.7.5b, tranziția de la nivelul 0 (O.K.) pe nivelul negativ.

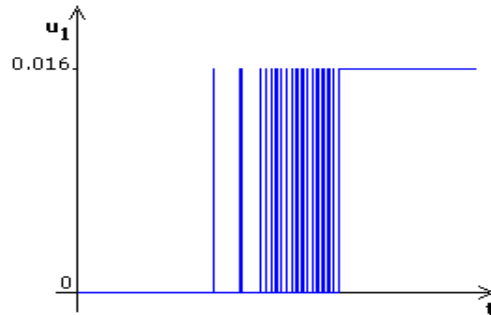


Fig. 3.7.5a O tranziție pozitivă a semnalului  $u_1$

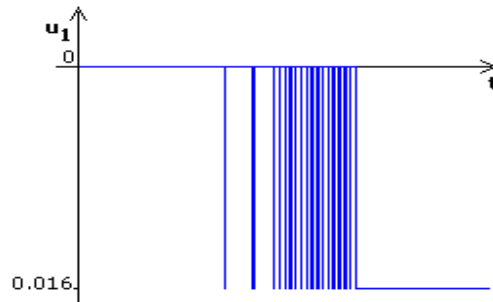


Fig. 3.7.5b O tranziție negativă a semnalului  $u_1$

-  $\geq \theta_{acmax}, \leq \theta_{armin}$  (pentru  $\theta_{ac}$  și  $\theta_{ar}$ ): este tratat similar cu primul grup de situații, valoarea nominală este înlocuită de  $\theta_{acmax}$ , respectiv  $\theta_{armin}$ .




Situațiile de mai sus se implementează utilizând blocurile din figurile 3.7.6a, 3.7.6b și 3.7.6c. Logica decizională este implementată utilizând blocuri logice: , , .



Fig. 3.7.6a Bloc care compară intrarea cu o valoare nominală superioară

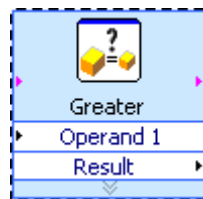


Fig. 3.7.6b Bloc care compară intrarea cu o valoare nominală inferioară

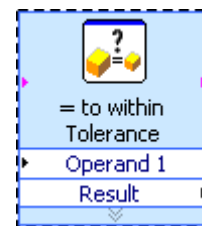


Fig. 3.7.6c Bloc care verifică egalitatea cu o valoare nominală plus un interval de toleranță

Precum a fost menționat, MFDDM este utilizat în regimuri staționare ale centralei electrice geotermale. Dacă MFDDM este conectat la sistem în timpul regimurilor tranzitorii furnizând succesiv mai multe tipuri de defecte, experimentul trebuie refăcut.

Analizând tabelul 3.6, se subliniază faptul că dacă apar două sau mai multe defecte simultan, considerând că sistemul de protecție nu a acționat, MFDDM va identifica doar un defect, cel dominant. După înlăturarea acestuia, următoarea detecție va furniza următorul defect. Deci, dacă defectele de la scenariile 11 și 12 apar simultan, MFDDM va identifica unul dintre ele la început și după înlăturarea acestuia îl va identifica pe celălalt.

Pentru a comunica rezultatele detecției și diagnozei, informațiile din tabelele diferitelor scenarii sunt redată grafic. S-au considerat patru exemple (scenarii) și s-au efectuat experimentele corespunzătoare. Rezultatele sunt prezentate în figurile 3.7.7, 3.7.8, 3.7.9 și 3.7.10. Aspectele studiate în cadrul acestui subcapitol au fost diseminate prin lucrarea [67].

În vederea relatării posibilităților de implementare, se descrie interfața cu utilizatorul corespunzătoare implementării în LabView a algoritmului de detecție și diagnoză a defectelor. Butoanele de activare a semnalelor din lista "Intrări de la PLC" (fig. 3.7.3) sunt grupate pe prima linie a interfeței cu utilizatorul. A doua linie ilustrează instrumentele de măsură ale temperaturilor, respectiv ale tensiunilor. Pe a treia linie apar indicatorii stării temperaturilor. Nivelul temperaturilor relativ la domeniul acceptabil (evidențiat în fig. 3.7.7 prin săgeți) este colorat. A patra linie indică starea detectată (prima coloană – defect sau normal, a doua coloană – indică localizarea defectului, a treia coloană – descrie tipul defectului localizat).

Primul experiment (fig. 3.7.7) este dedicat scenariului 1 (identificarea defectului de tip: TT<sub>1</sub>-NL).

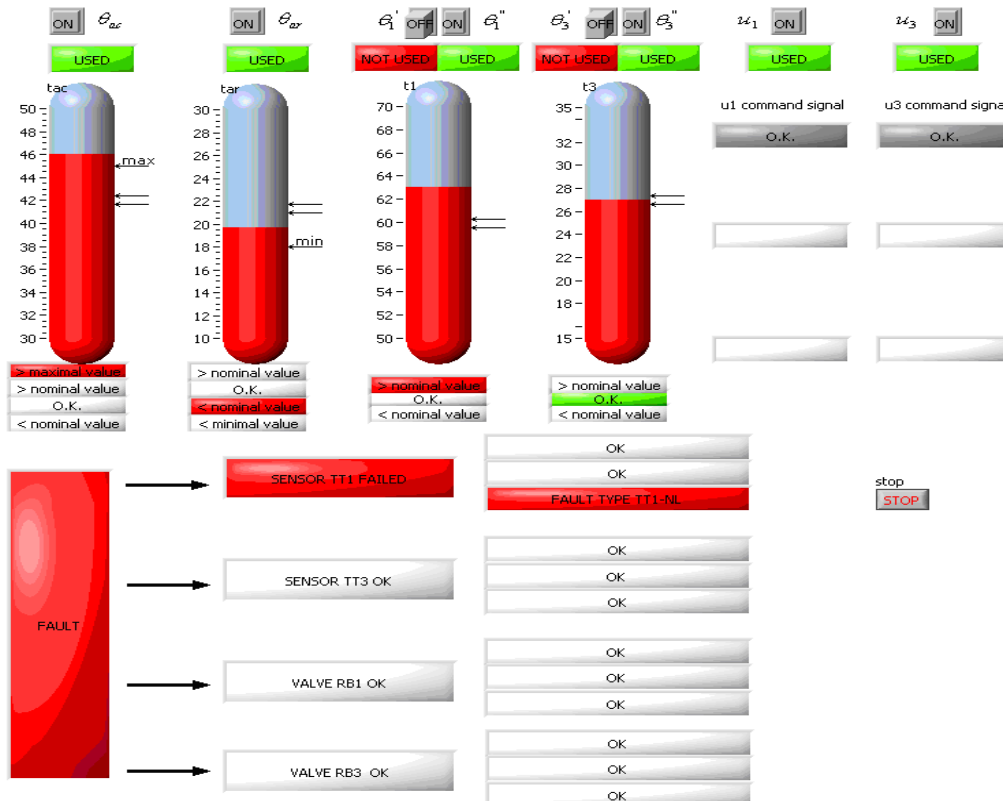


Fig. 3.7.7 Defect de tip TT<sub>1</sub>-NL

Fig. 3.7.8 se referă la defectul de tip TT<sub>3</sub>-N. Spre deosebire de figura anterioară, este ilustrată doar imaginea normală fără săgeți auxiliare.

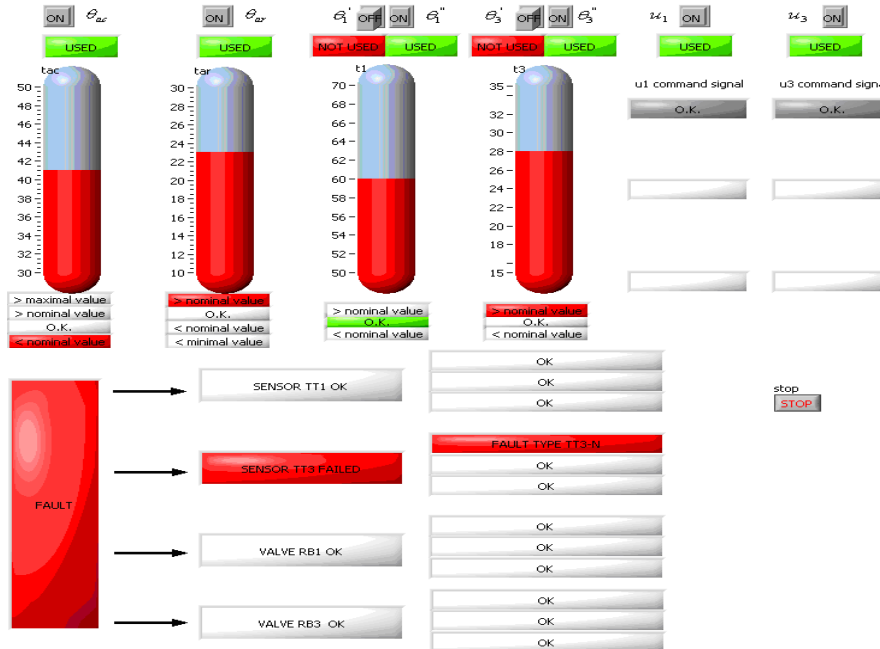


Fig. 3.7.8 Defect de tip TT<sub>3</sub>-N

Același procedeu este descris în figurile 3.7.9 și 3.7.10, pentru detecția și diagnoza defectelor de tip RB<sub>1</sub>-FC, respectiv RB<sub>3</sub>-FON.

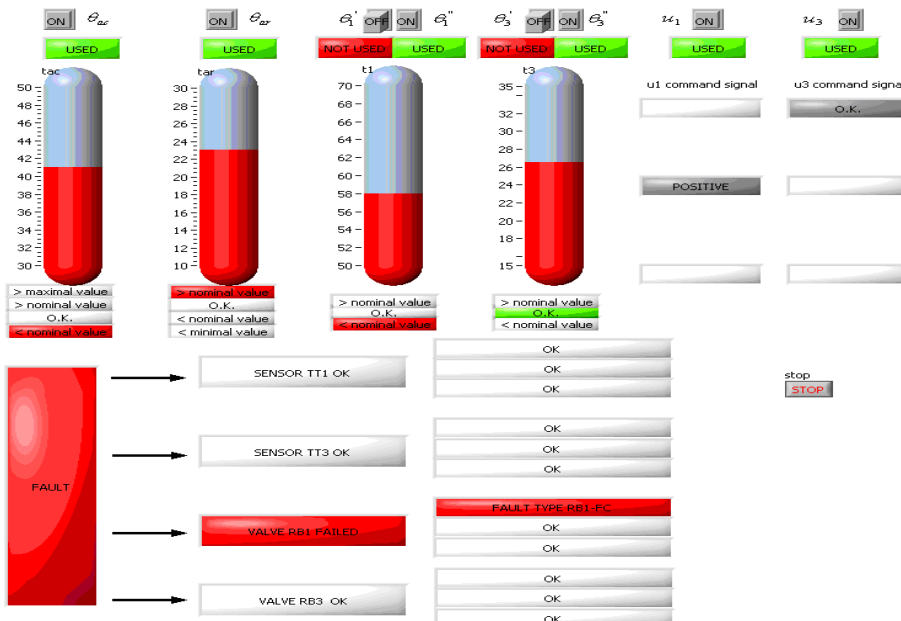
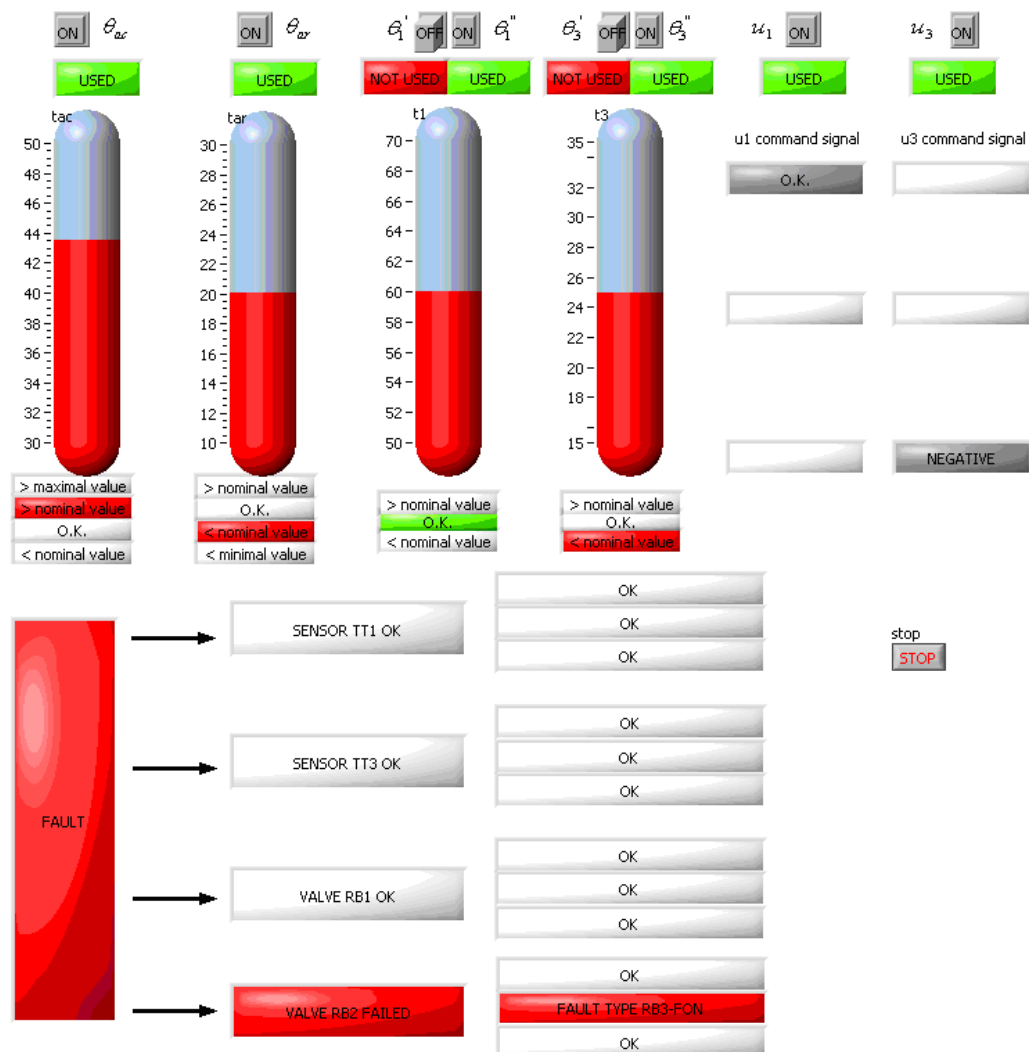


Fig. 3.7.9 Defect de tip RB<sub>1</sub>-FC

Fig. 3.7.10 Defect de tip RB<sub>3</sub>-FON

Pentru detecția defectelor apărute la nivelul a patru elemente considerate critice în cadrul centralei electrice geotermale, a fost conceput un modul mobil de detecție și diagnoză a defectelor. Conceptual, soluția este aplicabilă în diverse alte situații din energetică și nu numai. Echipamentul poate fi utilizat în orice cameră de comandă, sau poate fi mutat într-un arial tehnic. Pentru centrala geotermală, detecția defectelor la nivelul celor patru elemente este realizată prin implementarea unei matrici de corespondență reprezentată prin tabele. Matricea de corespondență a fost obținută prin studii analitice, care au fost înlocuite în cercetare de raționamente calitative în funcționarea diferitelor părți ale procesului. Este important de subliniat faptul că soluția prezentată face uz de resursele existente ale sistemului incluzându-le pe cele proiectate pentru a realiza un nivel minimal de redundanță.

## **4. Studiul unor aspecte referitoare la dependabilitatea roboților mobili**

Un al doilea domeniu aplicativ al tezei îl constituie roboții mobili cu roți. În cazul unui astfel de sistem automat, aspectele referitoare la dependabilitate se tratează dintr-o perspectivă puțin diferită decât în cazul centralei electrice geotermale (cap. 3). Capitolul curent își propune, întâi, să dezvolte o structură tolerantă la defecte pentru roboții mobili cu roți, iar apoi, să elaboreze și să implementeze o metodă de corecție a direcției de deplasare (orientare) a robotului prin preluarea și procesarea de imagine. Studiile prezintă, de asemenea, analize de fiabilitate și disponibilitate referitoare la dezvoltările realizate.

Capitolul se împarte în patru subcapitole. În cadrul primului subcapitol se prezintă structurile de reglare utilizate pentru sistemele robotizate, iar în al doilea se descriu defectele care pot să apară în sistemele robotizate. Al treilea subcapitol prezintă un concept de regulator interpolativ tolerant la defecte, iar ultimul subcapitol descrie o modalitate de creștere a disponibilității și siguranței în deplasarea roboților mobili cu roți.

### **4.1 Structuri de reglare utilizate pentru sisteme robotizate**

Subcapitolul descrie, în primul paragraf, tipurile de regulatoare utilizate pentru sisteme robotizate. Al doilea paragraf prezintă procedura de căutare și interpolare în cazul tabelor de interpolare tridimensionale, iar ultimul paragraf descrie modalitatea de tratare a depășirilor în tabelele de interpolare. Scopul prezentărilor din paragrafele acestui subcapitol este de a crea baza teoretică și motivațională pentru dezvoltările realizate în subcapitolul 4.3 (realizarea unui regulator tolerant la defecte). În acest context, primul paragraf converge spre regulatoarele tolerante la defecte, iar următoarele două redau elemente specifice implementării regulatoarelor interpolative cu trei intrări.

#### **4.1.1 Tipuri de regulatoare utilizate în sisteme robotizate**

Acest paragraf sintetizează pe baza unor lucrări de dată recentă principalele tipuri de regulatoare utilizate în sisteme robotizate. Se evidențiază principalele avantaje și dezavantaje aduse de fiecare structură în parte.

Studiul este axat pe roboți mobili care operează în medii cunoscute, pe roboți mobili care acționează în medii cu hazard și pe manipuloare robotice.

Problema controlului acestor tipuri de roboți este foarte complexă ca urmare a nelinearității și varianței în timp a modelelor roboților, precum și ca urmare a perturbațiilor exterioare imprevizibile etc. În cazul extrem al roboților mobili care evoluează în medii cu un grad mare de hazard, cum sunt roboții spațiali, roboți

subacvatici, roboți de căutare și salvare etc., acțiunile perturbatoare externe sunt imprevizibile. În majoritatea acestor cazuri un regulator liniar, chiar foarte complex, poate să eșueze datorită schimbărilor care apar în sistem și în mediul înconjurător în timpul operării robotului.

Totodată, în cadrul paragrafului, se prezintă din perspectiva satisfacerii cerințelor de siguranță în funcționare necesitățile și tendințele de evoluție ale reguletoarelor utilizate în cadrul sistemelor robotizate. Reguletoarele urmărite trebuie să prezinte pe lângă robustețe, capabilități de analiză și de acțiune în cazul apariției defectelor în sistem. Studiile legate de structurile de reglare sunt orientate spre obținerea reglării tolerante la defecțiuni. Regulatorul utilizat trebuie să fie cababil să-și ajusteze parametrii acordabili atunci când performanța generală se degradează.

Cu privire la sistemele robotizate, în literatură se menționează mai multe principii de reglare și tipuri de reguletoare, dintre care în continuare se rețin cele considerate mai semnificative prin polarizarea cercetărilor.

#### *Reglarea în regim modal alunecător*

Ea restricționează stările sistemului într-un anumit subspațiu al întregului spațiu al stărilor sistemului și face ca traiectoriile de stare să convergă asimptotic spre punctul de echilibru. Pentru proiectarea suprafeței de comutare și a legii de reglare cu structură variabilă acest tip de reglare necesită o estimare prealabilă a parametrilor și a incertitudinilor din sistem. Deși reglarea în regim modal alunecător este apreciată pentru robustețea sa la variații ale parametrilor procesului condus, ea prezintă problema inerentă a fenomenului de „chattering”, adică evoluția oscilatorie a comenzii procesului controlat în regimul modal alunecător. Fenomenul apare de obicei atunci când starea sistemului ajunge în apropierea suprafeței de comutare. Chattering-ul solicită exagerat elementele de execuție și de comutare.

Fenomenul ce „chattering” a fost studiat în ideea reducerii acestor oscilații. O soluție consacrată de reducere controlată a frecvenței oscilațiilor o reprezintă folosirea unor blocuri bipoziționale cu histeresis (fig. 4.1.1), lățimea  $2\varepsilon$  a semnalului de comutare adoptându-se astfel încât să rezulte o frecvență de comutare dorită. Drept consecință, starea sistemului evoluează spre punctul de echilibru pe o traiectorie în zig-zag. De regulă când sistemul ajunge în vecinătatea punctului de echilibru sistemul de reglare comută comanda pe o lege de reglare continuă.

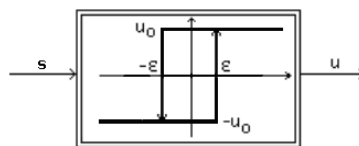


Fig. 4.1.1 Element bipozițional cu histeresis (s-semnal de comutare, u-mărime de comandă)

O soluție interesantă de reducere a oscilațiilor provenite în urma fenomenului de chattering se prezintă în [120]. Ea utilizează algoritmi genetici în sinteza regulatorului fuzzy modal alunecător.

În [123] s-a dezvoltat pentru un robot subacvatic un regulator adaptiv modal alunecător în care se utilizează un model nelinear al sistemului. În momentul în care perturbația generală acționează astfel încât starea sistemului depășește nivelul de toleranță al regimului modal alunecător, valoarea în exces este utilizată în a reconfigura parametrii modelului nelinear și intrarea regulatorului. În [102] s-a dezvoltat un regulator fuzzy de regim modal alunecător la care suprafața de

comutare este generată folosind logică fuzzy. Reglarea în regim modal alunecător este recomandată pentru aplicații de robotică și mecatronică în [56].

În situația roboților care lucrează în medii cu hazard, dar și în celelalte cazuri, apar în timpul funcționării incertitudini la nivelul parametrilor sistemului, precum și perturbații externe necunoscute. În acest context se recomandă *reglarea adaptivă*, care modifică parametrii legilor de reglare (în practică cel mai adesea factorii de proporționalitate din legile de reglare) atunci când apar schimbări în dinamica procesului sau când asupra procesului acționează perturbații. În situația în care viteza de schimbare a dinamicii sistemului este peste capacitățile de adaptare, acest tip de reglare poate da greș. În acest context, în [60] se propune un regulator adaptiv robust pentru roboți mobili care se bazează pe modele ale robotului.

Lucrarea [127] propune un regulator adaptiv cu observator de perturbații aplicat pentru roboți autonomi subacvatici. Reglarea se face cu ajutorul a două bucle, fig. 4.1.2, una internă unde este amplasat observatorul de perturbații, iar cea exterioară controlată de un regulator adaptiv nonregresor. Acest regulator nu necesită nici o informație fizică despre modelul robotului, exceptând numărul de intrări și de ieșiri ale acestuia. Robustețea este conferită de observatorul de perturbații care elimină perturbațiile externe precum și erorile de modelare și face ca sistemul să se comporte cât mai aproape de modelul nominal conceput de utilizator.

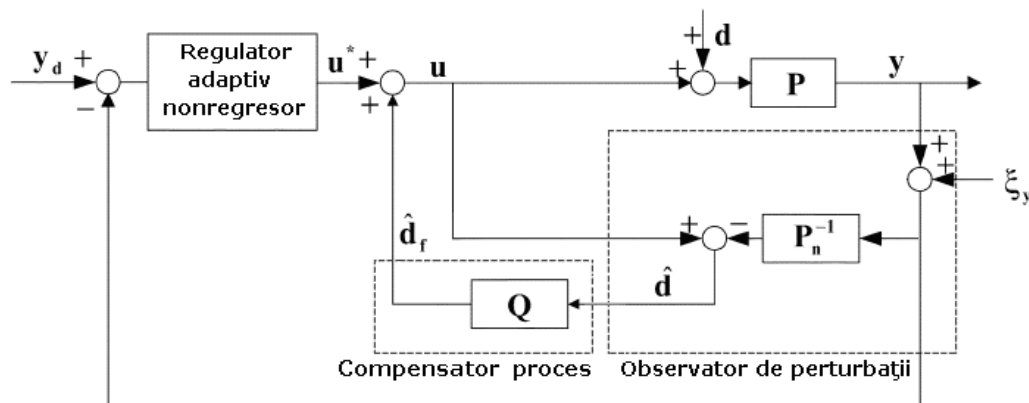


Fig. 4.1.2 Reglarea adaptivă utilizată în [127]

În figura 4.1.2,  $P_n$  este modelul nominal (de referință) al procesului  $P$ ,  $d$  este perturbație aditivă la intrarea procesului  $P$ , iar  $\xi_y$  este zgomotul de măsurare aditiv la ieșirea procesului. Prin intermediul filtrului  $Q$  se compensează efectul perturbației  $d$  prin modificarea adecvată a intrării  $u$ . Principiul prezentat prezintă două deficiențe: pe de-o parte modelul nominal este practic realizabil numai prin aproximare (trebuie să fie fizic realizabil), pe de altă parte modelul realizat împreună cu filtrul nu fac distincție între perturbația  $d$  și zgomotul  $\xi_y$ . În consecință efectul real în raport cu cel scontat este parțial.

De asemenea, o reglare adaptivă se utilizează și în cazul manipuletoarelor robotice în [1].

#### Reglarea bazată pe logică fuzzy

Reglarea interpolativă oferă un mijloc suplimentar de valorificare a numeroaselor metode de proiectare a sistemelor de reglare liniare și neliniare, întărindu-le importanța, [33]. Din acest punct de vedere, una dintre metodele care intră în discuție este metoda fuzzy.

Funcțiile de apartenență și regulile fuzzy alese au întotdeauna o măsură de „arbitrar” și astfel sistemele fuzzy sunt adesea bune dar nu sunt optimale. Suprafețele de control (comandă) obținute în cazul sistemelor fuzzy destinate reglării, cauzează adesea instabilitate. Simplitatea proiectării sistemelor fuzzy este uneori cauza pentru care acestea sunt considerate mai atractive decât rețelele neurale. În hardware, implementarea acestora este de actualitate pentru aplicațiile bazate pe microprocesoare. Problemei implementării structurilor fuzzy i-au fost dedicate numeroase cercetări, [15].

O problemă majoră în utilizarea practică a sistemelor fuzzy este problema resurselor. În general resursele (spațiul de memorie din chip) sunt limitate și astfel apar două mari neajunsuri care pot fi asociate în special numărului de intrări/ieșiri ale blocurilor fuzzy, precum și numărului funcțiilor de apartenență asociate intrărilor (granularitatea intrărilor). În acest context apare următoarea contradicție:

- Pe de-o parte, din cauză că baza de reguli crește exponențial cu fiecare intrare adăugată, numărul valorilor lingvistice (adică numărul funcțiilor de apartenență) trebuie limitat.
- Pe de altă parte, un număr mic de valori lingvistice (de funcții de apartenență) conduce la rezultate foarte slabe.

Dacă resursele ar fi nelimitate, iar viteza de procesare foarte mare, contradicția nu ar mai apărea.

În contextul implementării contradicția menționată apare prin necesitatea luării unei decizii cu privire la raportul dintre numărul de intrări și numărul funcțiilor de apartenență. Cu cât numărul intrărilor este mai mare, cu atât valoarea raportului trebuie să fie mai mare. Astfel, în cazul în care sistemul are intrări puține se pot folosi mai multe valori lingvistice pentru intrări, adică mai multe funcții de apartenență, rezultând erori mici, iar dacă prezintă intrări multe, se folosește un număr mai mic de funcții de apartenență cu riscul de a obține erori mari.

Funcțiile de apartenență cele mai utilizate sunt cele de tip trapezoidal, triunghiular și gaussian, iar metodele de defuzificare cele mai folosite sunt metodele Zadeh și Tagagi-Sugeno.

În [127], se subliniază alte două aspecte care îndeamnă la prudență în utilizarea metodei fuzzy. Un prin aspect este faptul că determinarea regulilor lingvistice și a funcțiilor de apartenență necesită date experimentale, în consecință necesită timp. Al doilea aspect este legat de faptul că baza de reguli face dificil de autorizat comportamentul sistemului în buclă închisă pentru a determina timpul de răspuns și stabilitatea.

Prin prisma ideilor care se discută cu privire la conducerea roboților sunt de reținut și lucrările [72] și [112]. În [72] se propune un sistem de inferență auto-adaptiv bazat pe logică neuro-fuzzy, care este utilizat pentru roboți subacvatici. În [112] se utilizează reglarea fuzzy pentru evitarea obstacolelor statice de către un robot mobil.

#### *Reglarea bazată pe rețele neurale*

Neuronul este elementul de bază al rețelelor neurale. Acesta este în esență o structură de calcul cu mai multe intrări și o ieșire. Operația efectuată de neuron constă în însumarea ponderată a intrărilor și în procesarea acestei valori pe baza unei funcții de activare rezultând astfel ieșirea.

Rețelele neurale necesită deseori calculul unor funcții de activare complexe, cum sunt cele bazate pe tangentă hiperbolică, ceea ce conduce la o suprasolicitare a microprocesoarelor simple. Cu toate acestea, sistemele bazate pe rețele neurale se implementează în primul rând în variantă software (prin programe de calcul) [15],



iar proprietățile de aproximare foarte bune fac din ele o alternativă în raport cu implementările hardware.

De regulă se folosesc rețelele neurale cu structură în cascadă și MLP. Diferența dintre cele două tipuri o constituie numărul de neuroni ce sunt dispuși pe un strat (în cazul rețelei cascadă fiecare strat conține un neuron, în timp ce rețeaua MLP poate avea pe fiecare strat mai mulți neuroni). Dimensiunea mai mică și numărul mic de neuroni constituie un avantaj pentru rețelele neurale de tip cascadă. Astfel, acestea sunt mult mai ușor de implementat hardware.

În prezent implementarea sistemelor bazate pe rețele neurale este de tip software, adică printr-un program care face legătura dintre intrările și ieșirile corespunzătoare. Codul astfel generat se compilează și se încarcă în memoria procesorului. O implementare de rețele neurale pe un modul FPGA este ilustrată în [91].

În [52] se descrie pentru un manipulator robotic un regulator în regim modal alunecător dezvoltat utilizând rețele neurale. Rețele neurale multistrat sunt utilizate pentru a compensa incertitudinile din sistem. Concluzia studiului este că regulatorul bazat pe rețele neurale este mai performant decât regulatorul de la care s-a plecat și care utilizează doar reglarea în regim modal alunecător.

Analiza implementării sistemelor fuzzy în comparație cu implementarea sistemelor bazate pe rețele neurale conduce la următoarele concluzii:

- Codul implementat pe un procesor în cazul rețelelor neurale este mult mai simplu și mai scurt decât în cazul sistemelor fuzzy. În consecință, se folosește mai puțină memorie.
- Suprafața de comandă obținută în cazul sistemelor bazate pe rețele neurale conferă sistemelor de reglare o stabilitate mai mare și un control mai bun decât în cazul sistemelor fuzzy.
- Sistemele fuzzy sunt mult mai ușor de proiectat, de realizat și de implementat. Ele sunt mai simple și mai ușor de înțeles.
- Un dezavantaj major al sistemelor fuzzy este capacitatea limitată de a rezolva problemele complexe (intrări-ieșiri multiple).

#### *Reglarea tolerantă la defecțiuni*

Principiile de reglare prezentate se utilizează în mod obișnuit în contextul în care nu se pune problema toleranței la defecțiuni, ci doar problema asigurării performanțelor de reglare, cel mult în condiții de incertitudine [38]. În cazul de față interesează utilizarea lor și în contextul realizării unor sisteme tolerante la defecțiuni. Se vorbește despre reglare tolerantă la defecțiuni atunci când sistemul de reglare trebuie să facă față ieșirii din limitele normale de funcționare, sau defectării propriu-zise a diferitelor elemente ale sistemului. Este vorba de o proprietate de robustețe diferită de cea care face obiectul domeniului sistemelor de reglare robuste.

Regulatoarele tolerante la defecțiuni reprezintă una dintre direcțiile esențiale spre care se îndreaptă studiile referitoare la reglarea sistemelor cu roboți (un studiu referitor la defectele care apar în sistemele cu roboți se prezintă în subcapitolul 4.2). În lucrările [127], [81], [82] se specifică o astfel de direcție de dezvoltare. De remarcat este faptul că cele trei referințe bibliografice acoperă cele trei tipuri de roboți la care se face referire pe parcursul tezei (subcapitolul 4.2): roboții mobili care evoluează în medii cu hazard, manipolatoarele robotice și roboții mobili care evoluează în spații semicunoscute.

În cadrul subcapitolului 4.3 al acestei lucrări, se dezvoltă un regulator tolerant la defecțiuni. Capitolul prezintă și elemente teoretice legate de astfel de regulatoare.

### 4.1.2 Căutarea și interpolarea tridimensională locală

În acest paragraf se consideră un regulator de tip interpolativ cu trei intrări și o mărime de ieșire pentru care se prezintă formulele de interpolare liniară locală folosite de autor la implementare, precum și modalitatea prin care se utilizează aceste formule. Atributul „local” se referă la faptul că pentru interpolare într-un punct curent se utilizează cele opt puncte de sprijin amplasate în colțurile cubului în care se înscrie punctul curent (vezi fig. 4.1.3). Elemente referitoare la regulatoarele interpolative au fost prezentate în cadrul subcapitolului 3.4.

Pentru interpolare sunt necesare opt puncte de sprijin amplasate în colțurile unui cub (fig. 4.1.3).

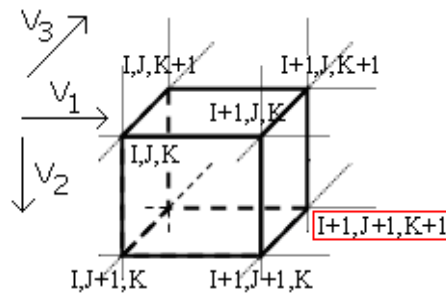


Fig. 4.1.3 Punctele care contribuie la interpolarea în trei dimensiuni

Notațiile din figuri se referă la cele trei direcții rectangulare indicate prin săgețile asociate celor trei coordonate  $V_1$ ,  $V_2$  și  $V_3$ , reprezentând în limbaj curent linii ( $i$ ), coloane ( $j$ ), respectiv fețe ( $k$ ) ale tabelului de interpolare.

Calculul interpolativ parcurge două etape:

- căutarea
- interpolarea

Căutarea înseamnă determinarea punctului de sprijin esențial, printr-un proces de parcurgere a domeniilor de intrare pentru a identifica cele trei coordonate ale locului de sprijin esențial. Poziția locului de sprijin esențial în cadrul domeniilor de intrare va conduce la identificarea valorii de sprijin esențiale.

Domeniile de intrare se prezintă în felul următor:

$D_a = \{V_{a1}, V_{a2}, \dots, V_{ap}\}$ , unde  $V_{ai}$  reprezintă coordonatele de rang  $a$  (locurile de sprijin),  $a \in \{1, 2, 3\}$ .

Căutarea locului de sprijin esențial se va realiza în funcție de intrarea curentă pe dimensiunea respectivă. Fig. 4.1.4 ilustrează modalitatea de căutare a locului de sprijin esențial pe o coordonată de rang  $a$ ,  $V_a$  reprezentând intrarea curentă în funcție de care se face căutarea.

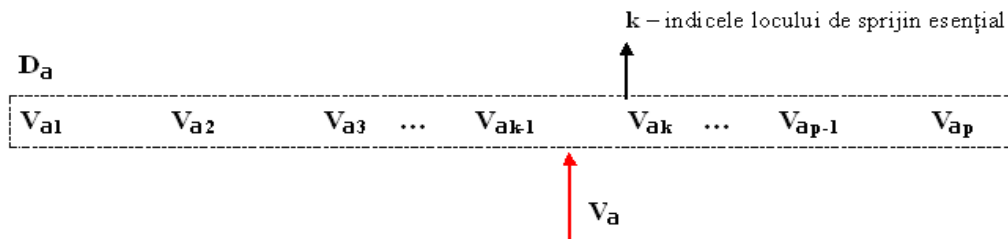


Fig. 4.1.4 Căutarea locului de sprijin esențial în domeniul de intrare

După ce s-a identificat coordonata locului de sprijin esențial pe fiecare din cele trei dimensiuni ale tabelului de interpolare, se extrage din tabel valoarea de sprijin esențială. Fig. 4.1.5 ilustrează valoarea de sprijin esențială  $R_{i,j,k}$ , precum și valorile de sprijin care intră în calculul final al ieșirii interpolate  $R_{i-1,j-1,k-1}$ ,  $R_{i-1,j,k-1}$ ,  $R_{i,j-1,k-1}$ ,  $R_{i,j,k-1}$ ,  $R_{i-1,j-1,k}$ ,  $R_{i-1,j,k}$ ,  $R_{i,j-1,k}$ .

$R_{11k-1}$	...	$R_{1j-1k-1}$	$R_{1jk-1}$ ...	$R_{1nk-1}$				
						$R_{1j-1k}$	$R_{1jk}$ ...	$R_{1nk}$
$R_{i-11k-1}$	...	$R_{i-1j-1k-1}$	$R_{i-1jk-1}$ ...	$R_{i-1nk-1}$				
$R_{i-1k-1}$	...	$R_{ij-1k-1}$	$R_{ijk-1}$ ...	$R_{ink-1}$		$R_{i-1j-1k}$	$R_{i-1jk}$ ...	$R_{i-1nk}$
						$R_{ij-1k}$	$R_{ijk}$ ...	$R_{ink}$
$R_{n1k-1}$	...	$R_{nj-1k-1}$	$R_{njk-1}$ ...	$R_{nkk-1}$				
						$R_{nj-1k}$	$R_{njk}$ ...	$R_{nkk}$

Valoare de sprijin esențială

Fig. 4.1.5 Valoarea de sprijin esențială

Procedura prezentată anterior redă modalitatea de obținere a punctului de sprijin esențial  $P_{i,j,k} = (V_{1i}, V_{2j}, V_{3k}, R_{i,j,k})$ .

După ce s-a finalizat etapa de căutare și s-a identificat cu succes punctul de sprijin esențial, se trece la următoarea etapă: interpolarea.

Interpolarea tridimensională furnizează valoarea funcției  $f$ , notată cu  $R$ , în punctul curent  $V_1$ ,  $V_2$  și  $V_3$ :  $R = f(V_1, V_2, V_3)$ . Ea se realizează prin următoarele formule de calcul în care  $R'$ ,  $R''$ ,  $R'''$ ,  $R''''$ ,  $R_1$  și  $R_2$  reprezintă valori ajutătoare:

$$R' = R_{i-1j-1k} + \frac{V_2 - V_{2j-1}}{V_{2j} - V_{2j-1}} (R_{i-1jk} - R_{i-1j-1k}) \quad (1)$$

$$R'' = R_{ij-1k} + \frac{V_2 - V_{2j-1}}{V_{2j} - V_{2j-1}} (R_{ijk} - R_{ij-1k}) \quad (2)$$

$$R_1 = R' + \frac{V_1 - V_{1i-1}}{V_{1i} - V_{1i-1}} (R'' - R') \quad (3)$$

$$R''' = R_{i-1j-1k-1} + \frac{V_2 - V_{2j-1}}{V_{2j} - V_{2j-1}} (R_{i-1jk-1} - R_{i-1j-1k-1}) \quad (4)$$

$$R'''' = R_{ij-1k-1} + \frac{V_2 - V_{2j-1}}{V_{2j} - V_{2j-1}} (R_{ijk-1} - R_{ij-1k-1}) \quad (5)$$

$$R_2 = R''' + \frac{V_1 - V_{1i-1}}{V_{1i} - V_{1i-1}} (R'''' - R''') \quad (6)$$

$$R = R_2 + \frac{V_3 - V_{3k-1}}{V_{3k} - V_{3k-1}}(R_1 - R_2) \quad (7)$$

În aceste relații:

$V_1$  – intrarea pe linii,  $V_2$  – intrarea pe coloane,  $V_3$  – intrarea pe fețe,  
 $R_{i-1,j-1k-1}$ ,  $R_{i-1,jk-1}$ ,  $R_{i,j-1k-1}$ ,  $R_{i,jk-1}$  reprezintă valorile de sprijin de pe fața  $k-1$ ,  
 $R_{i-1,j-1k}$ ,  $R_{i-1,jk}$ ,  $R_{i,j-1k}$ ,  $R_{i,jk}$  reprezintă valorile de sprijin de pe fața  $k$ . Cu ajutorul acestor puncte se va calcula  $R_1$  și  $R_2$  care reprezintă valorile funcției  $f$  pe fețele respective obținute prin interpolare bidimensională. Valoarea interpolării tridimensionale  $R$  se va calcula apoi pe baza acestor două puncte.

#### 4.1.3 Tratarea depășirilor pentru tabelele de interpolare

În cazul în care semnalul de intrare al regulatorului interpolativ pe cele trei coordonate ( $V_1$ ,  $V_2$  sau  $V_3$ ) este mai mic decât prima valoare a domeniului de intrare pe linii, coloane, respectiv fețe, cursorul de căutare se fixează pe primul element ( $V_{11}$ ,  $V_{21}$ , sau  $V_{31}$ ) și interpolarea nu mai beneficiază de toate valorile (se vor putea utiliza doar jumătate din valorile necesare, anume cele din margini). Dacă semnalul de intrare ( $V_1$ ,  $V_2$  sau  $V_3$ ) este mai mare decât ultima valoare a domeniului de intrare pe linii, coloane, respectiv fețe, cursorul de căutare nu va găsi elementul pe care să se poziționeze, astfel interpolarea nu va beneficia de locul și valoarea de sprijin esențială.

Pentru a elimina această deficiență a fost imaginată o modalitate de rezolvare care se va ilustra în cele ce urmează.

Pentru interpolarea tridimensională, tabelul de interpolare va fi completat în următoarea manieră,:

- cu două coloane la extremitățile fiecărei fețe a tabloului tridimensional
- cu două linii pe fiecare față a tabloului tridimensional; liniile adăugate au dimensiunea deja modificată, în sensul că se ține cont și de adăugarea celor două coloane suplimentare din prima etapă
- cu două fețe suplimentare la extremitățile tabloului tridimensional; aceste fețe adăugate au dimensiunea celor deja modificate.

Tabelul la care s-a ajuns este ilustrat în fig. 4.1.6, iar tot ce s-a adăugat este trecut cu roșu.

Raționamentul modificării tabelului este următorul:

- Noile linii, coloane, respectiv fețe s-au adăugat ca fiind valori de sprijin pentru noi valori de intrare (locuri de sprijin) notate cu  $V_{MIN}$  și  $V_{MAX}$ . Valorile  $V_{MIN}$  și  $V_{MAX}$  se aleg astfel încât să se afle în afara domeniului anticipat de variație al intrărilor  $V_1$ ,  $V_2$  și  $V_3$  ceea ce înseamnă că la orice moment de timp intrările  $V_1$ ,  $V_2$  sau  $V_3$  sunt cu siguranță mai mari decât  $V_{MIN}$  și mai mici decât  $V_{MAX}$ , algoritmul având de fiecare dată la dispoziție opt valori pentru interpolare.
- Pentru a nu modifica rezultatul interpolării, elementele adăugate la extremități sunt copii ale liniilor, coloanelor, respectiv ale fețelor vecine. În felul acesta capetele asociate muchiilor de tabel ajung să aibă valori egale.

Dacă unul dintre semnalele de intrare pe o dimensiune depășește valoarea primului sau ultimului loc de sprijin, ieșirea interpolată pe acea dimensiune este identică cu valoarea de sprijin esențială.

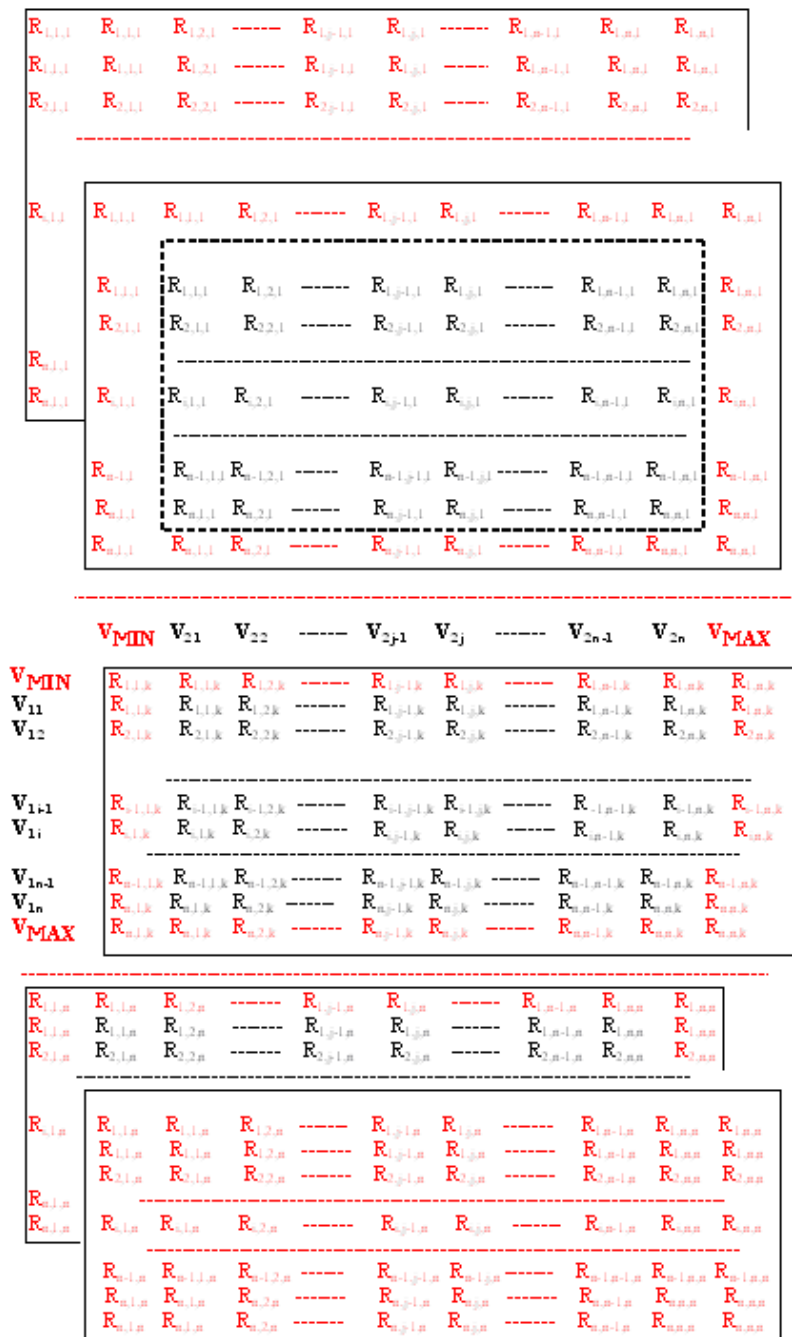


Fig. 4.1.6 Tabelul tridimensional rezultat în urma tratării depășirilor

## 4.2 Defecte în sisteme robotizate

În cele ce urmează se consideră trei categorii de roboți:

- roboți mobili care evoluează în medii cu hazard ridicat
- roboți mobili care evoluează în spații semicunoscute
- manipolatoare robotice

Paragrafele acestui subcapitol prezintă aspecte referitoare la defectele care apar în cazul fiecărei categorii de roboți. Informațiile furnizate în cadrul subcapitolului reprezintă punctul de plecare și direcția studiilor referitoare la dependabilitatea roboților. De exemplu, fundamentul studiilor realizate în subcapitolele 4.3 și 4.4 (studii care se axează pe dependabilitatea roboților mobili care evoluează în spații semicunoscute) se regăsește în paragraful 4.2.2.

### 4.2.1 Roboți care evoluează în medii cu hazard ridicat

Prin mediu cu hazard ridicat înțelegem un mediu înconjurător ostil, cu un grad ridicat de risc, inclusiv cu privire la integritatea robotului. Exemple de roboți care evoluează în astfel de medii sunt: roboți spațiali, subacvatici, roboți utilizați în misiuni de cercetare și salvare etc. Integritatea lor poate fi periclitată de o serie de evenimente neprevăzute. Din această cauză siguranța în funcționare a unor astfel de roboți trebuie să fie ridicată.

Pentru a putea analiza siguranța în funcționare a sistemelor robotizate care evoluează în medii cu hazard ridicat este nevoie de o identificare a principalelor tipuri de defecte și de determinarea ratelor de defectare în diferite situații. Identificarea defectelor, respectiv determinarea ratelor de defectare se realizează în principiu în două contexte:

- experimente de analiză a evoluției sistemului robotizat pe o perioadă îndelungată de timp într-un mediu creat de elementul uman, care, prin scenariu adecvate, să surprindă diverse situații test;
- monitorizare și analiză a evoluției sistemului robotizat pe parcursul unei situații reale.

În acest sens în literatură se întâlnesc un număr redus de studii (considerând ambele tipuri de experimente). Rezultatele cele mai semnificative ale acestor studii sunt sintetizate în cele ce urmează.

În [23] se sintetizează o cercetare bazată pe zece studii în care s-au utilizat 15 modele de roboți mobili în medii cu hazard ridicat. Roboții mobili testați diferă semnificativ atât ca dimensiuni, cât și ca grad de autonomie. Cel mai important studiu efectuat se referă la o situație de criză reală (misiuni de salvare în cazul atentatelor de la World Trade Center). Un alt studiu acoperă monitorizarea a 13 roboți pe o perioadă de doi ani, iar celelalte opt studii au fost efectuate pe teren de către o organizație de specialitate.

Cercetarea a urmărit identificarea defectelor, determinarea cauzelor și efectelor acestora, precum și posibilitatea de reparare. În acest context, concluzia studiului este că fiabilitatea în condițiile de mediu cu hazard este foarte scăzută, timpul mediu de defectare luând valori între 6 și 20 de ore, ceea ce înseamnă o rată de defectare  $\lambda \in [0.05, 0.17] \text{ ore}^{-1}$ .

Lucrarea concluzionează că în proporție de 50 % efectorii sunt cauza primară de defectare a roboților, iar unul dintre studiile referite arată că în proporție de 54% cauza principală a defectării provine de la sistemul de reglare.

Tot în [23] se face o clasificare a defectelor în funcție de sursa acestora. Această clasificare, ilustrată în fig. 4.2.1, este utilă în distingerea categoriilor de defecte posibile, precum și a surselor de defecte la roboții mobili care evoluează în medii cu hazard.

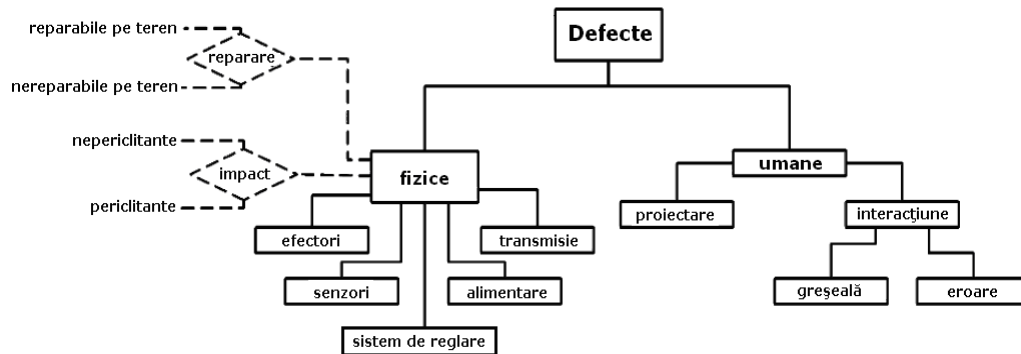


Fig. 4.2.1 Schemă de clasificare a defectelor la roboții mobili din medii cu hazard

Defectele se clasifică în două categorii: *fizice* și *umane*. Defectele fizice ale roboților se împart în cinci categorii: defecte la *efectori*, defecte la *sistemul de reglare*, defecte la *sistemul senzorial*, defecte la *alimentare* și defecte la *transmisie*. Defectele fizice au, de asemenea, două caracteristici legate de *reparabilitate* și de *impact*. Un defect se consideră *reparabil pe teren* dacă poate fi reparat la locul incidentului de către un operator instruit, în condiții favorabile de mediu. Așa cum s-a precizat în cadrul subcapitolului 1.3, în funcție de severitatea defectului fizic se disting: defecte *periclitante* și *nepericlitante*. Prin defect periclitant se înțelege un defect care cauzează o imposibilitate de continuare a misiunii. Defectul nepericlitant produce o degradare a capacității robotului de a-și realiza sarcinile, dar acesta poate să-și continue misiunea.

Defectele umane se împart în defecte de *proiectare* și defecte de *interacțiune*. Defectele de proiectare apar în urma unor erori în proiectare, realizare sau a unor modificări post-producție la nivelul robotului. Defectele de interacțiune se împart în funcție de sursa lor în două categorii: defecte provocate de greșeli și defecte provocate de erori, denumite în mod simplificat *greșeli* și *eroari*. O greșeală se consideră o utilizare neconformă conștientă cauzată de neînțelegerea situației și realizarea unui obiectiv diferit de cel impus. O eroare se definește ca o utilizare neconformă inconștientă cauzată de operator în încercarea nereușită de a realiza obiectivul propus și înțeles corect.

Unul dintre aspectele studiate în [23] și prezentat în continuare se referă la trei roboți de tipuri diferite. În figurile 4.2.2, 4.2.3 și 4.2.4 se ilustrează cele trei tipuri de roboți. Dimensiunile acestora diferă. Primul aspect urmărit este cel al defectelor fizice. Studiul nu se referă la defecte de transmisie deoarece acestea nu au fost identificate. În tabelul 4.1 sunt defalcate defectele sub formă procentuală, pe categorii mari de defecte, pentru sistemele din figurile 4.2.2 și 4.2.4.

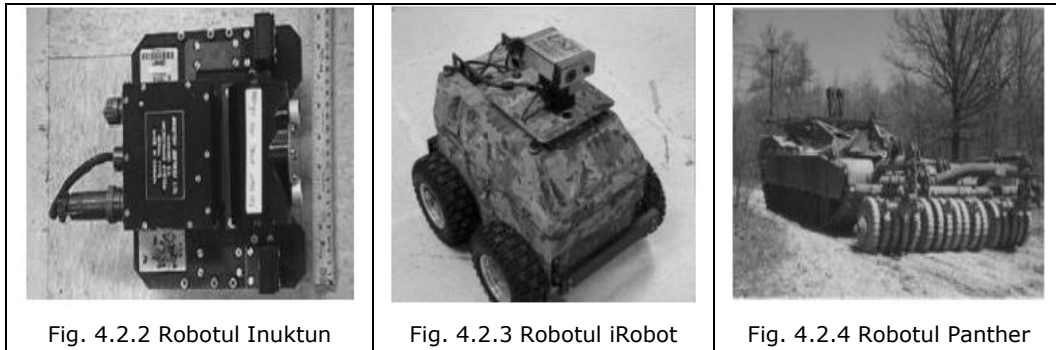


Fig. 4.2.2 Robotul Inuktun

Fig. 4.2.3 Robotul iRobot

Fig. 4.2.4 Robotul Panther

Model	Efectori	Sistem de reglare	Alimentare	Senzori
<b>Panther</b>	11%	54%	9%	26%
<b>Inuktum</b>	47%	38%	3%	12%

Tabel 4.1 Defectarea subsistemelor roboților mobili

Totodată s-a determinat că 94 % dintre defectele apărute la robotul din fig. 4.2.4 au fost defecte periclitante.

Al doilea aspect este cel al determinării indicatorilor de fiabilitate. În tabelul 4.2 sunt precizați indicatori de fiabilitate referitori la roboții din figurile 4.2.2 și 4.2.3, determinați pe baza unui studiu care s-a desfășurat timp de doi ani asupra a 13 roboți, reprezentați prin 7 modele diferite și provenite de la trei producători. Evaluarea caracteristicilor defectelor apărute a arătat că 80 % din totalul defectelor au fost periclitante, iar 65 % au fost reparabile pe teren.

Model	MTBF(ore)	Disponibilitate	Timp mediu în stare de defectare MDT(ore)
<b>Inuktum</b>	6.14	90%	177
<b>iRobot</b>	6.27	36%	207
<b>Total</b>	6.17	64%	185

Tabel 4.2 Elemente de fiabilitate pentru doi roboți mobili

Din tabelul 4.2 se poate observa percepția eronată a autorilor asupra disponibilității. Indicatorii MTBF și MDT prezintă o relevanță deosebită pentru studii ulterioare de specialitate, dar disponibilitatea calculată nu este disponibilitatea operațională (vezi subcapitolul 1.2) ci disponibilitatea inerentă. Din această cauză, pentru un studiu relevant de fiabilitate este absolut necesară cunoașterea timpului mediu în stare de defectare MDT furnizat în tabel, idee cunoscută de către autori.

Lucrarea [21] stabilește un timp mediu până la defectare de 8.3 ore pentru roboții mobili care operează în medii cu hazard ridicat analizați. În fine, lucrarea [24] identifică sistemul senzorial ca fiind cel care necesită cea mai mare îmbunătățire.

În concluzie, se poate afirma că roboții care evoluează în medii cu hazard necesită o atenție accentuată relativ la siguranța în funcționare. Din datele de mai sus rezultă că, disponibilitatea acestora este scăzută. Un aspect îngrijorător și important este procentul ridicat de defecte periclitante din totalul defectelor apărute, rezultând o toleranță scăzută la defecte. Principala consecință a numărului mare de defecte periclitante este un grad scăzut de autonomie al robotului.



### 4.2.2 Roboți mobili care evoluează în spații semicunoscute

După cum s-a precizat, pentru a descoperi problemele asociate funcționării unui sistem robotizat, defectele care pot apărea, precum și pentru a determina ratele de reparare și de defectare, este necesară o utilizare și o analiză a sistemului robotizat într-un timp îndelungat. În acest domeniu există foarte puține cercetări. O astfel de testare și analiză pe un orizont de timp îndelungat, este realizată în [5] și se referă la un robot mobil care evoluează în spații semicunoscute (un spațiu cunoscut în care pot apărea obstacole dinamice). Acest robot a fost conceput să funcționeze fără intervenție umană, reîncărcarea bateriilor fiind realizată autonom de robot.

Sistemul robotizat este prezentat în fig. 4.2.5:

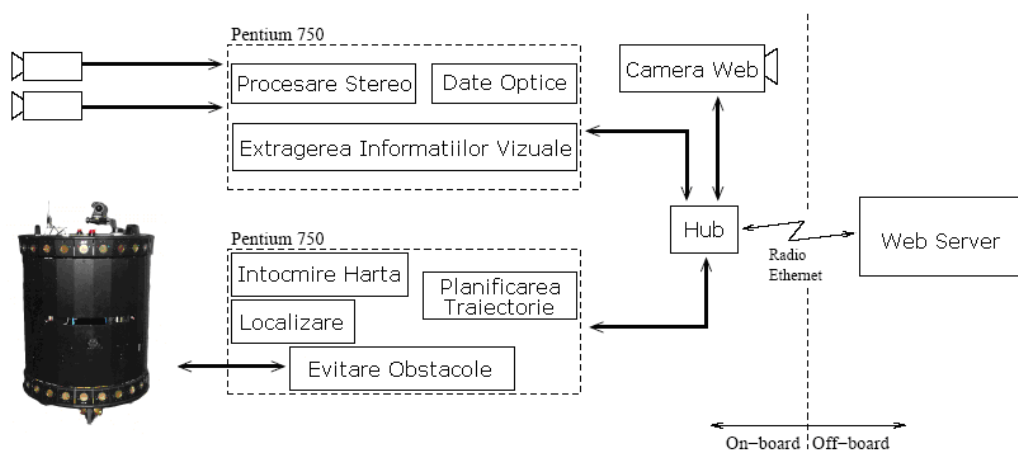


Fig. 4.2.5 Componentele hardware și software ale robotului mobil

Sistemul robotizat conține două calculatoare interne cu procesoare Pentium la 750 Mhz. Un procesor rulează software-ul de navigare: întocmirea hărții, determinarea rutei de deplasare, localizarea, evitarea obstacolelor. Celălalt procesor se ocupă de preluarea și procesarea imaginilor preluate.

Sistemul robotizat a fost pus la dispoziția utilizatorilor pe internet, pentru a-l manipula de la distanță. Robotul a fost testat timp de 140 de ore în cursul cărora a parcurs peste 25 de km. Defectele sistemului robotizat în această perioadă sunt ilustrate în fig. 4.2.6.

După cum se poate observa în fig. 4.2.6 sistemul robotizat nu poate să funcționeze fără a se recurge totuși la intervenția umană, fiind necesară o supervizare și o intervenție prin corecție manuală a erorilor.

Prima dată s-a defectat regulatorul, după 24 de ore de funcționare, chiar înainte de procesul de reîncărcare a bateriilor. Drept consecințe, se recomandă realizarea de regulatoare tolerante la defecte, precum implementarea de structuri redundante la nivelul sistemului de achiziție senzorial.

Al doilea tip major de defect este cel de localizare. Dacă robotul nu poate să-și determine poziția sau să determine poziția elementelor care îl înconjoară atunci el nu mai poate să continue acțiunea, defectul fiind unul major. Ca urmare, se recomandă dezvoltarea de module de localizare redundante, pentru a mări gradul de siguranță în funcționare. O localizare eronată poate fi cauzată de erorile de

odometrie (vezi paragraful 4.4.3), de structurile de achiziție de informații (senzori), dar și de defecte software sau hardware din interiorul sistemului.

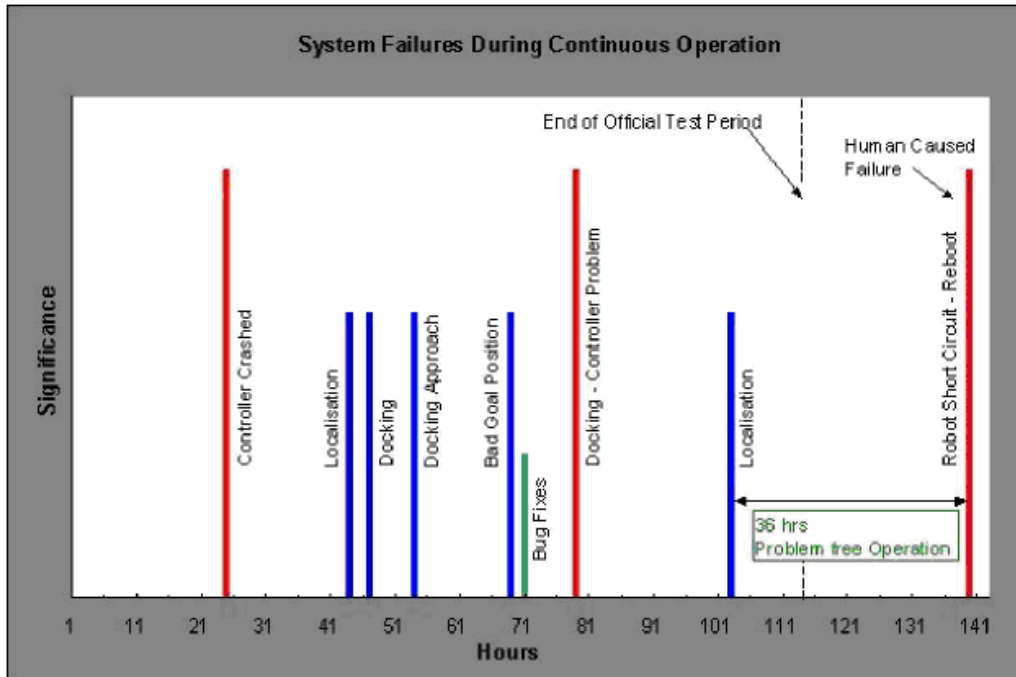


Fig. 4.2.6 Defectele sistemului robotizat pe perioada testării și semnificația lor

„Bad goal position” este un defect care trimite spre fiabilitatea software, deoarece utilizatorul prin teleoperare a putut să trimită robotul la un punct care nu exista. Această situație indică spre o dezvoltare ulterioară de programe software robuste, precum și utilizarea de structuri redundante software și informaționale.

De asemenea în fig. 4.2.6 se observă și defecte datorate intervenției umane. În concluzie, nu se poate neglija nici elementul fiabilității umane.

Pentru creșterea nivelului de disponibilitate, funcțiile critice ale unui sistem necesită o abordare redundantă. La modulele de localizare aplicarea structurilor redundante este evident iminentă. Roboții mobili au o fiabilitate ridicată în momentul în care au cel puțin două scheme de localizare, cu senzori multipli și metode multiple de preluare ale informației. Modulele destinate evitării obstacolelor trebuie de asemenea implementate de o manieră redundantă.

Există un număr mic de lucrări care tratează evoluția roboților mobili în medii dinamice (obstacole mobile). În [4] se face un astfel de studiu, dar nu se analizează posibilitatea de a mări siguranța sistemului de achiziție de date referitoare la obstacolele mobile cu care se poate interacționa, sistem care prezintă un risc mare de defectare.

În [88] se analizează patru roboți autonomi utilizați timp de 5 ani într-un muzeu. Funcționarea propriu-zisă a acestora excede 7 ani. Roboții utilizați au ajuns la în final la 72-216 de ore timp mediu până la defectare. Autorii susțin în [89] faptul că acest MTBF este maximul care se poate obține într-o situație reală. Pe parcursul acestor 7 ani s-au efectuat în permanență îmbunătățiri pentru a obține o disponibilitate cât mai ridicată și în final s-a putut obține un astfel de MTBF.

În [113] se descrie un experiment de același tip, unde roboții au fost analizați timp de 64 de zile, ajungându-se la 7 ore timp mediu până la defectare. Acest MTBF de 7 ore a fost obținut după îmbunătățiri repetate și reparații, deoarece în prima săptămână în urma analizei s-a obținut un MTBF de sub o oră. Graficul evoluției MTBF este ilustrat în fig. 4.2.7 (perioada de după prima săptămână).

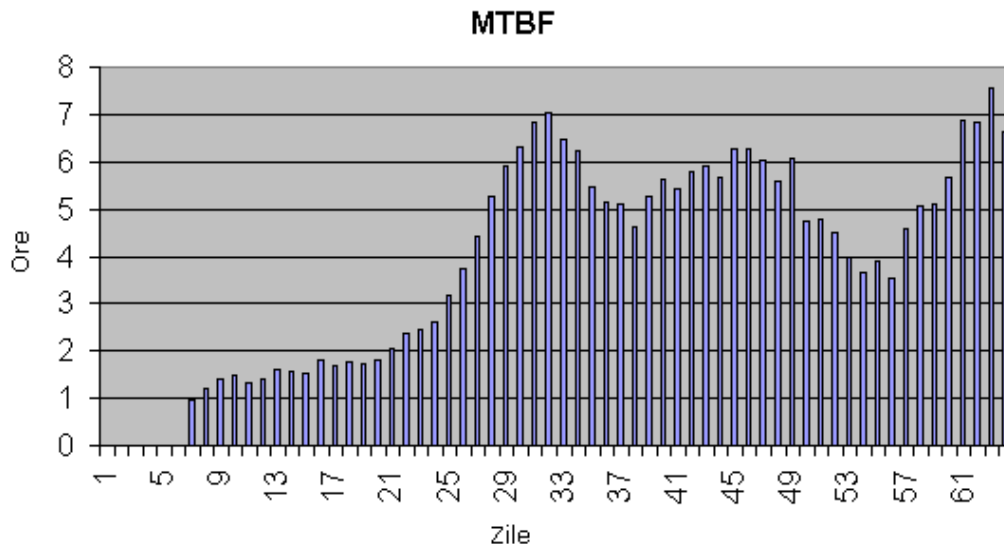


Fig. 4.2.7 Evoluția MTBF pentru roboții testați în [113].

În ambele cazuri, [88] și [113], roboții și-au desfășurat activitatea în interior, în spații semicunoscute.

Studiile prezentate în [21] și [22] redau un MTBF de 19.5 ore pentru roboții mobili care evoluează în medii semicunoscute. Cercetările s-au efectuat pe parcursul a trei ani și reprezintă un din puținele studii utile de unde se pot extrage date referitoare la MTTR și MDT în vederea analizei disponibilității.

Aspecte prezentate în [4], [5], [88], [113], [21] și [22] vor fi discutate și utilizate în cadrul analizei disponibilității din paragraful 4.4.6.

### 4.2.3 Manipulatoare robotice

Și în cazul manipulatoarelor robotice, cercetările legate de defecte posibile și siguranță în funcționare sunt reduse la număr. S-au identificat doar două lucrări care examinează defectele la manipulatoarele robotice.

În lucrarea [13] se prezintă studii experimentale referitoare la influența factorului uman în sisteme cu manipulatoare robotice. Lucrarea [106] analizează defectele și rapoartele de mentenanță ale manipulatoarelor robotice.

Rapoartele cuprind 200 de roboți, care își desfășoară activitatea pe o perioadă de 21 de săptămâni, reprezentând cinci firme producătoare. S-a constatat că timp de 3.95 ore/săptămână manipulatoarele robotice au fost defecte, în stare de reparare, sau în proces de mentenanță.

Defectele identificate predominante au fost defectele de poziționare (localizare – robotul nu reușea să-și deplaseze brațul pe traiectoria stabilită) în

proporție de 45%, urmate de defectele sistemului de conducere, în proporție de 25%.

Tendențele actuale de creștere a siguranței în funcționare a manipuletoarelor robotice includ structurile redundante. Lucrarea [86] dezvoltă ideea acționării redundante.

### 4.3 Conceperea unui regulator tolerant la defecte pentru sisteme robotizate

Problematika reguletoarelor tolerante la defecte reprezintă o direcție spre care în prezent se îndreaptă numeroase cercetări [16], [19], [81], [82], [98], [105], [127].

În general există două abordări ale reglării tolerante la defecte : *abordarea pasivă* și *abordarea activă*. *Abordarea pasivă* se bazează pe un regulator robust care asigură insensibilitatea sistemului în buclă închisă la anumite defecțiuni. Această strategie este eficientă doar în anumite situații cu o gamă foarte îngustă de defecte posibile. *Abordarea activă* este mult mai eficientă și mai complexă, bazându-se pe reconfigurarea buclei de reglare potrivit estimării defectelor și specificațiilor care urmează a fi îndeplinite de către sistemul defect.

În cadrul abordării active a reglării tolerante la defecte se disting două tipuri de metode: metode bazate pe proiectare, respectiv metode bazate pe reconfigurarea/corecția automată on-line a regulatorului. Metodele bazate pe proiectare implică ideea reconfigurării buclei de reglare pe baza selectării în funcție de apariția diferitelor tipuri de defecțiuni, precum și de severitatea acestora a unor legi de reglare preimplementate. Metodele bazate pe reconfigurarea/corecția automată on-line a regulatorului implică proceduri de recalculare a parametrilor regulatorului la apariția defectului.

Realizarea unui regulator tolerant la defecte înseamnă parcurgerea a trei etape:

- proiectarea unui regulator propriu-zis suficient de robust care să se poată reconfigura
- realizarea unei scheme de detecție și diagnoză a defectelor
- conceperea și implementarea mecanismului de reconfigurare/corecție potrivit

Dacă structura sau parametrii regulatorului se pot ajusta on-line în conformitate cu măsurările în timp real ale efectelor defectelor, atunci regulatorul este numit tolerant la defecte.

Acest subcapitol este structurat în patru părți. Prima parte prezintă realizarea unui regulator interpolativ propriu-zis cu trei intrări de la care se pornește în realizarea regulatorului tolerant la defecte. A doua parte propune o structură redundantă bazată pe referință, care reprezintă o schemă de detecție și diagnoză a defectelor, oferind în același timp o posibilitate de înlocuire a unui element la nivelul căruia a apărut un defect. A treia parte a capitolului descrie mecanismul de corecție automată on-line a regulatorului. Ultima parte a capitolului descrie experimentele efectuate care ilustrează eficiența elementelor prezentate.

Regulatorului tolerant la defecte obținut aparține categoriei abordărilor active, iar corecția se face în mod automat on-line. O reprezentare schematică în cadrul buclei de reglare a regulatorului tolerant la defecte care se propune în cadrul acestui capitol este prezentată în figura 4.3.1:

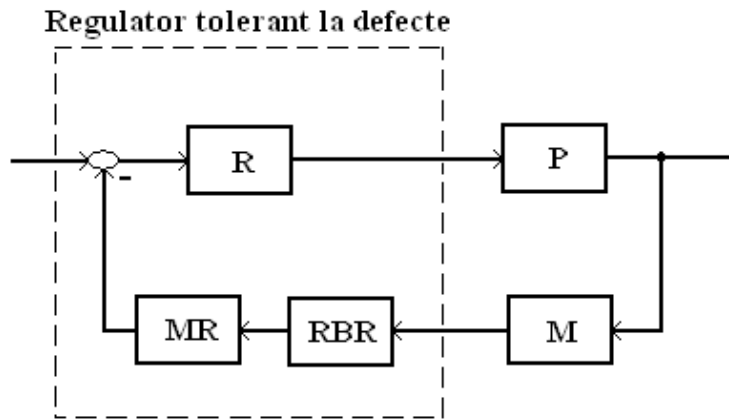


Fig. 4.3.1 Încadrarea regulatorului tolerant la defecte în bucla de reglare

Realizarea regulatorului tolerant la defecte necesită parcurgerea a trei etape corespunzătoare celor trei blocuri componente din fig. 4.3.1:

- regulatorul interpolativ propriu-zis – R
- blocul de implementare a redundanței bazată pe referință– RBR
- mecanismul de corecție - MR

#### 4.3.1 Regulatorul interpolativ propriu-zis

O caracteristică a sistemelor robotizate descrisă în subcapitolul 4.1, omniprezentă la cele cu roboți mobili, este neliniaritatea proceselor care implică și utilizarea unor regulatoare neliniare. Regulatoarele fuzzy, respectiv regulatoarele bazate pe rețele neurale, care reprezintă în esență regulatoare de tip interpolativ, satisfac necesitățile legate de compensarea neliniarității. În subcapitolele 4.1 și 3.4 s-a realizat o comparație între regulatoarele menționate și regulatoarele interpolative, prezentându-se avantajele și dezavantajele care survin în urma utilizării fiecărei structuri.

Sistemul robotizat (procesul condus) utilizat în cadrul cercetării reprezintă un robot mobil cu o masă ( $m_{\text{robot}}$ ) de 5 kg, care se deplasează pe roți cu raza ( $r_{\text{aza}_r}$ ) de 0.02 m, în spații semicunoscute, pe o traiectorie rectilinie, unde pot să apară obstacole dinamice. În ipoteza că întreaga mișcare poate fi redusă la cea a axului motorului de acționare, al cărui moment de inerție înglobează întreaga sarcină acționată, sistemului robotizat îi corespund relațiile de mai jos, respectiv schema bloc din figura 4.3.2.

$$u = R \cdot i + L \frac{di}{dt} - u_e$$

$$M = k_t \cdot i$$

$$J \dot{v} = M - M_f$$

$$M_f = c \cdot v$$

$$u_e = k_b \cdot v$$

$$J_{robot} = m_{robot} \cdot raza_r^2 \cdot 1.1$$

$$J = J_{motor} + \frac{J_{robot}}{g_r^2}$$

Notățiile semnifică:

u – tensiunea de comandă

r – rezistența indusului

i – curentul prin indus

L – inductanța

u<sub>e</sub> – tensiunea indusă

M – cuplu activ al motorului

k<sub>t</sub> – constanta de cuplu

J – momentul total de inerție (la arborele motorului)

v – viteza

M<sub>f</sub> – cuplu de frecare

c – coeficientul de frecare vâscoasă la nivelul axului motorului

k<sub>b</sub> – constanta tensiunii electromotoare

J<sub>robot</sub> – momentul de inerție al robotului (partea diferită de motor)

m<sub>robot</sub> – masa robotului

raza<sub>r</sub> – raza roților robotului

J<sub>motor</sub> – momentul de inerție al motorului

g<sub>r</sub> – raportul de transmisie

d – distanța parcursă de robot

Schema corespunde în esență unei acționări cu motor de curent continuu comandat pe indus a cărui mișcare de rotație este transformată în mișcare de translație printr-un reductor-convertor.

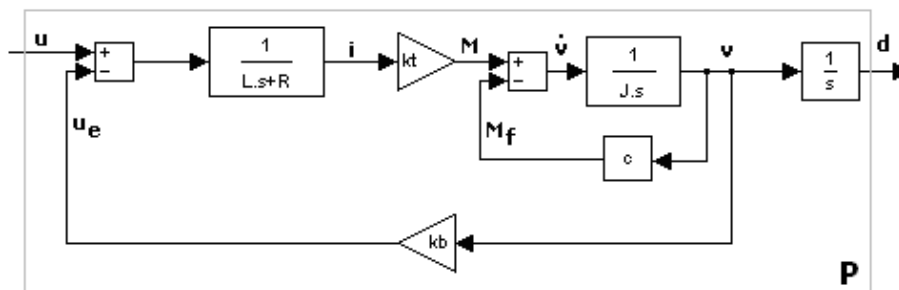


Fig. 4.3.2 Schema bloc asociată procesului condus în cadrul sistemului cu regulator interpolativ

Structura de reglare propusă are două obiective:

- să controleze viteza de deplasare a sistemului pentru a parcurge distanța prescrisă pe o traiectorie rectilinie
- să controleze viteza de deplasare a sistemului la apariția obstacolelor mobile astfel încât să se evite situații periclitante.

O schemă generală a structurii de reglare se prezintă în fig. 4.3.3, unde d<sub>p</sub> și v<sub>c</sub> sunt mărimile d și v din cadrul schemei 4.3.2 furnizate de elemente de măsură specifice M<sub>1</sub> și M<sub>2</sub> (de exemplu, d<sub>p</sub>, distanța parcursă, este furnizată de codificatoarele de impulsuri de la roți sau de scheme de localizare bazate pe senzori de distanță, iar v<sub>c</sub> este furnizată de senzori de viteză).

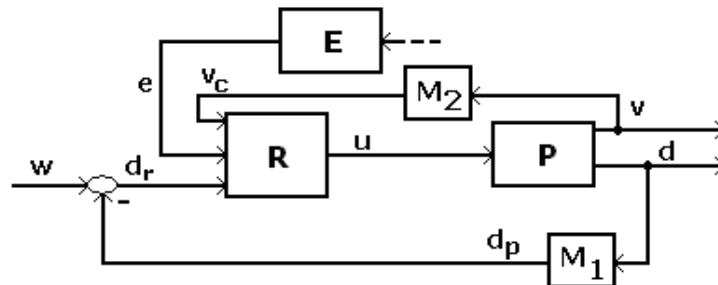


Fig. 4.3.3 Schemă generală a structurii de reglare

Dezvoltările prezentate în continuare își propun realizarea unui regulator interpolativ propriu-zis care să controleze viteza  $v_c$  a robotului mobil în funcție de distanța care a mai rămas de parcurs ( $d_r$ ) pentru a ajunge la destinație (distanța prescrisă până la destinație este  $w$ , iar distanța parcursă este  $d_p$ ), de viteza curentă ( $v_c$ ), respectiv în funcție de distanța până la obstacolul dinamic ( $e$  - furnizată de un senzor de distanță  $E$ ) care se apropie, sau se depărtează de sistem (lucrările care se ocupă cu tratarea problemei elementelor dinamice în spațiul de mișcare al robotului sunt reduse la număr [4]). Sistemul va detecta apariția în plaja lui de observabilitate (intervalul de distanțe observabile față de obstacolul dinamic, din cadrul sectorului de cerc care poate fi furnizat de senzorul de distanță) a unui element mobil, determinând la fiecare pas de eșantionare distanța față de acesta.

Se precizează că regulatorul interpolativ propriu-zis ( $R$  în figurile 4.3.1 și 4.3.3) este conceput doar pentru situațiile în care la nivelul senzorului de distanță ( $E$ ) nu se produc defecte.

Regulatoarele interpolative se bazează pe implementarea prin tabele a punctelor de sprijin (vezi subcapitolul 3.4 și paragraful 4.1.2). Un punct de sprijin are ca și coordonate coordonatele unui loc de sprijin și valoarea de sprijin asociată locului de sprijin respectiv. În acest context punctul de sprijin este  $P_{i,j,k} = (V_{1i}, V_{2j}, V_{3k}, R_{i,j,k})$ . Locurile de sprijin ( $V_{1i}, V_{2j}, V_{3k}$ ) corespund mărimilor de intrare în regulator (domeniul de intrare al tabelului de interpolare), iar valoarea de sprijin,  $R_{i,j,k} = f(V_{1i}, V_{2j}, V_{3k})$  (vezi subcapitolul 3.4 și paragraful 4.1.2), corespunde mării de ieșire. Valorile de sprijin se identifică în tabelul de interpolare ca date existente la eticheta definită de coordonatele locurilor de sprijin respective (vezi figura 4.1.4 și 4.1.5 din paragraful 4.1.2).

Scopul subcapitolului este de a realiza un regulator interpolativ tolerant la defecte. Regulatorul interpolativ tolerant la defecte va include în structura sa regulatorul interpolativ propriu-zis  $R$  (vezi figura 4.3.1) și în plus va menține sistemul în stare de perfectă funcționare în pofida apariției unor defecțiuni la nivelul senzorului de distanță ( $E$ ). Prima etapă în realizarea regulatorului interpolativ tolerant la defecte este dezvoltarea regulatorului interpolativ propriu-zis ( $R$ ). Regulatorul interpolativ propriu-zis realizat inițial (în cadrul acestui subcapitol) se va reconfigura în urma detecției unui anumit defect.

Regulatorul interpolativ propriu-zis implementat este destinat controlului vitezei de deplasare a robotului mobil prin intermediul tensiunii de alimentare a acestuia. Extrăgând regulatorul  $R$  din schema din figura 4.3.3 se obține figura 4.3.4 care redă un subsistem cu orientarea  $\{v_c, d, e\} \rightarrow \{u\}$ , ce furnizează tensiunea de comandă a motorului ( $u$ ) în funcție de:

- $v_c$  - viteza curentă
- $d$  - distanța rămasă de parcurs
- $e$  - distanța față de obstacolul mobil

Dependența  $u = f(v_c, d, e)$  este implementată prin interpolare liniară.

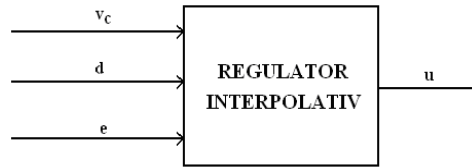


Fig. 4.3.4 Schema bloc a regulatorului interpolativ propriu-zis

Blocul de interpolare este tridimensional, cu dimensiunea  $7 \times 7 \times 7$ . Liniile, coloanele, respectiv fețele de la extremități sunt astfel concepute încât să prezinte siguranță din punctul de vedere al depășirii limitelor semnalelor de intrare (paragraful 4.1.3). Dimensiunea  $7 \times 7 \times 7$  poate fi asimilată în 7 tabele bidimensionale de dimensiune  $7 \times 7$ .

Relațiile următoare prezintă ca exemplu câteva dintre regulile care guvernează reglarea interpolativă. Reguliile prezentate sunt cele care reproduc deplasarea robotului mobil în situația în care nici un obstacol mobil nu se află în aria de observabilitate a robotului, mărimea  $e=0$ , și se concretizează în al 7-lea tabel bidimensional din cadrul tabelului de interpolare al regulatorului interpolativ propriu-zis.

$$\left\{ \begin{array}{l} \text{Dacă } (d = 0) \text{ atunci } u = 0 \\ \text{Dacă } (d = \text{mică}) \text{ atunci } \{u = 0.5 + 0.2 \cdot v_c\} \\ \text{Dacă } (d = \text{mică\_mijlocie}) \text{ atunci } \{u = 1 + 0.2 \cdot v_c\} \\ \text{Dacă } (d = \text{mijlocie}) \text{ atunci } \{u = 1.5 + 0.2 \cdot v_c\} \\ \text{Dacă } (d = \text{mare}) \text{ atunci } \{u = 2 + 0.04 \cdot v_c\} \end{array} \right.$$

Celelalte 6 tabele bidimensionale tratează prin reguli adaptate sau modificate situațiile în care obstacolul mobil se apropie de robot ( $e \neq 0$ ).

Schema sistemului de reglare este ilustrată în figura 4.3.5.

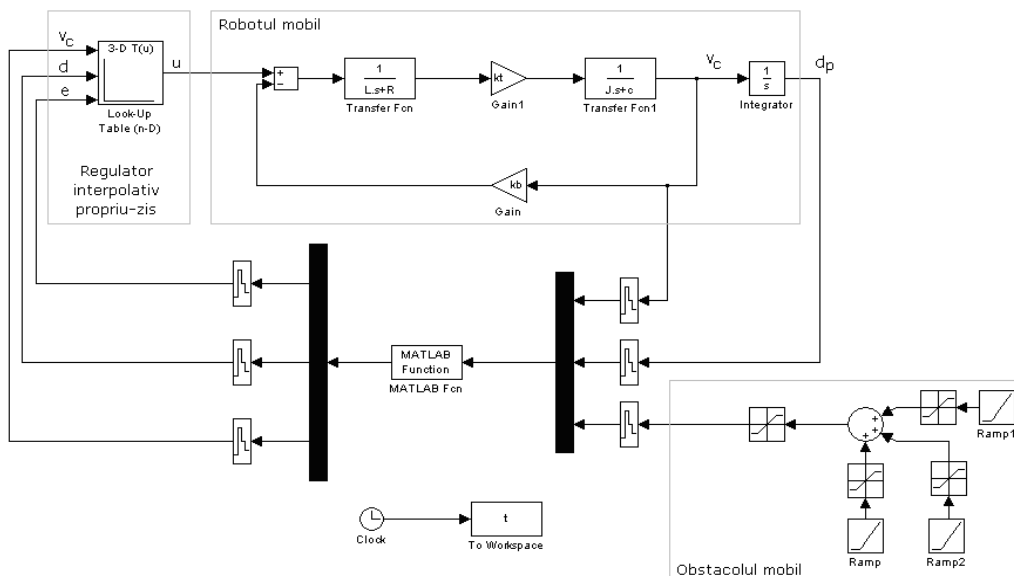


Fig. 4.3.5 Schema de simulare a evoluției robotului mobil



Prin blocul Matlab Function se implementează cele două elemente care definesc (împreună cu regulatorul interpolativ propriu-zis) regulatorul interpolativ tolerant la defecte: redundanța bazată pe referință și mecanismul de corecție al regulatorului.

Mișcarea robotului mobil pentru care se concepe structura de reglare tolerantă la defecte este caracterizată de următorul scenariu:

- robotul mobil trebuie să parcurgă o distanță prescrisă evoluând pe o traiectorie rectilie
- dacă nici un obstacol mobil nu se va apropia de el, atunci acesta accelerează până la viteza maximă de 2 m/s (denumim viteza maximă – viteza de deplasare în condițiile în care nu există nici un element perturbator în plaja de observabilitate a robotului mobil), continuându-și astfel drumul până la destinație, unde va încetini și se va opri.
- dacă robotul detectează pe parcursul mișcării sale un obstacol mobil care intră în plaja lui de observație, atunci va încetini în funcție de distanța față de acesta. În cazul în care obstacolul mobil se apropie prea mult de robot, acesta din urmă se va opri și va staționa până când elementul se va îndepărta de el. În momentul în care entitatea mobilă începe să se depărteze, robotul își reia mișcarea ajungând la viteza maximă dacă nu mai detectează nimic în plaja de observabilitate.

Pentru a modifica dinamica robotului, se poate corecta sau reconfigura tabelul de interpolare (prin reconfigurare se înțelege modificarea punctelor de sprijin din tabelul de interpolare; posibilitatea reconfigurării reprezintă unul dintre avantajele reguletoarelor interpolative). În paragrafele următoare se va utiliza această proprietate a reglării interpolative pentru a realiza o corecție online asupra locurilor de sprijin ale regulatorului interpolativ propriu-zis.

Evoluția robotului mobil controlat de regulatorul interpolativ tridimensional este ilustrată printr-un experiment. În cadrul experimentului robotul mobil trebuie să parcurgă o distanță prescrisă și să trateze problema apariției obstacolelor mobile în plaja de observabilitate (fig. 4.3.6, 4.3.7 și 4.3.8). Distanța de deplasare prescrisă este 30 m (în figură distanțele se reprezintă în cm). Fig. 4.3.6 ilustrează evoluția distanței obstacolului mobil față de robot, fig. 4.3.7 descrie distanța parcursă de robotul mobil, iar fig. 4.3.8 prezintă evoluția vitezei acestuia. Sistemul pornește și accelerează până la viteza maximă. La momentul  $t = 4,6$  s, un obstacol mobil intră în plaja de observabilitate a robotului și se apropie de acesta cu viteza de 1m/s. În consecință, robotul începe să încetinească gradat în funcție de distanța față de obiect. Obstacolul mobil se apropie până la distanța de 1 m, iar robotul va încetini până la viteza 0.3 m/s. Distanța rămâne constantă timp de 3,5 s, iar viteza robotului mobil se stabilizează la valoarea de 0.3 m/s în acel interval de timp. La 10s, elementul perturbator se depărtează cu viteza de 1.5 m/s. Se observă cum robotul accelerează ca urmare a distanțării față de element, ajungând la viteza maximă în momentul în care acesta iese total din limita de observabilitate. Se continuă mișcarea până la punctul final.

Paragraful curent prezintă regulatorul interpolativ propriu-zis, demonstrând eficiența acestuia în a controla viteza robotului mobil pentru a parcurge o distanță prescrisă într-un mediu dinamic cu posibilitatea apariției unor obstacole mobile. Paragrafele următoare prezintă cele două elemente care dezvoltă regulatorul interpolativ propriu-zis conferindu-i toleranță la defecte (dacă senzorul de distanță care furnizează distanța până la un eventual obstacol mobil se defectează).

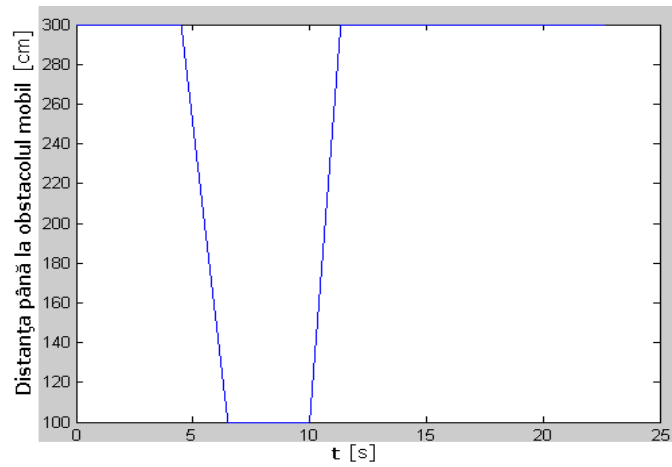


Fig. 4.3.6 Distanța obstacolului mobil față de robot

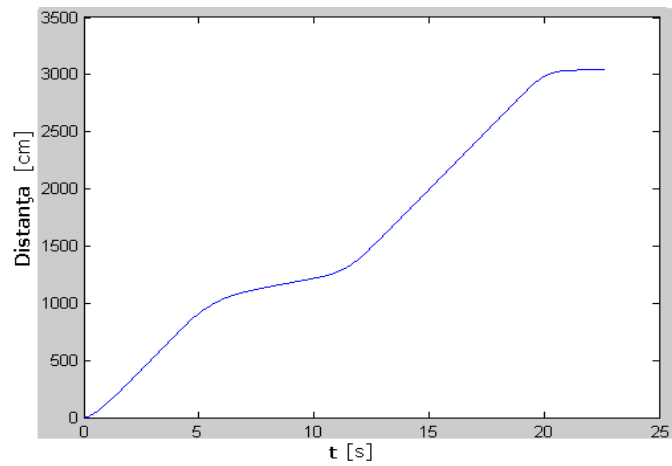


Fig. 4.3.7 Distanța parcursă de robotul mobil

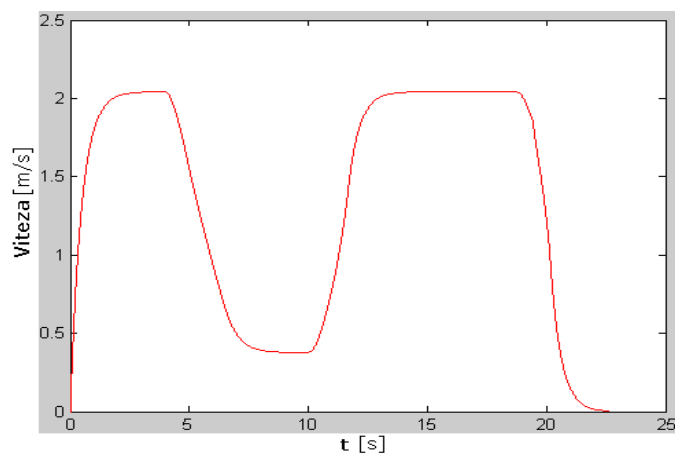


Fig. 4.3.8 Evoluția vitezei robotului mobil

### 4.3.2 Redundanța bazată pe referință

Regulatorul interpolativ propriu-zis (R) al sistemului de reglare din figurile 4.3.1 și 4.3.3 este destinat să controleze viteza robotului mobil astfel încât sistemul să parcurgă rectiliniu distanța prescrisă și să reacționeze la apariția unui obstacol mobil care poate să apară în plaja lui de observabilitate. După cum s-a specificat în paragraful anterior, realizarea regulatorului interpolativ propriu-zis (R) reprezintă prima etapă în obținerea regulatorului interpolativ tolerant la defecte.

Datorită faptului că în cadrul sistemelor cu roboți structurile de achiziție de date reprezintă entități cu una dintre cele mai mari rate de defectare, respectiv fiabilitate scăzută, [19], precum și importanța crescândă a asigurării capabilității roboților mobili de a detecta și reacționa corespunzător la obstacolele mobile [4], studiul întreprins în continuare este destinat asigurării toleranței la defectele produse în structura senzorială de determinare a distanței (de achiziție de date) robotului mobil față de elementul perturbator mobil.

Pentru a obține regulatorul interpolativ tolerant la defecte schema de reglare se completează cu modulul de detecție și diagnoză a defectului (etapa a doua). Modulul va trebui să confere împreună cu mecanismul de corecție (MR), toleranța la defecte pentru regulatorului interpolativ propriu-zis conceput în paragraful anterior. Pe de altă parte, se dorește ca schema să permită o înlocuire de manieră redundantă a elementului defect în situația în care se produce un defect major al sensorului de distanță, astfel încât acesta devine neutilizabil în continuare. Poziția schemei de detecție și diagnoză a defectelor, în cazul curent redundanța bazată pe referință (RBR), în cadrul structurii de reglare este ilustrată în figura 4.3.9. Cu linie punctată este ilustrată corecția aplicată de către blocul MR,  $y_d$  și  $e$  sunt intrarea și ieșirea (distanța față de obstacolul mobil furnizată de structura de achiziție) blocului RBR iar  $a$  reprezintă factorul de corecție.

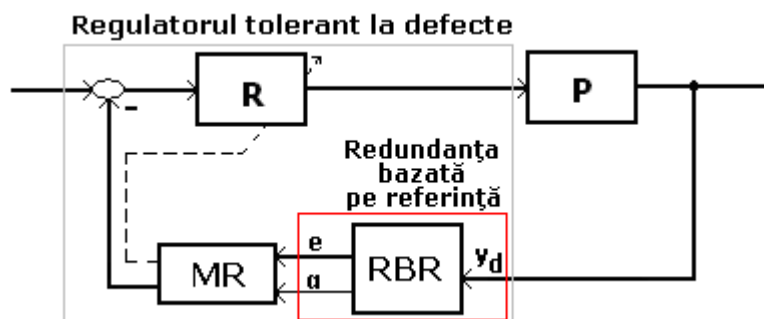


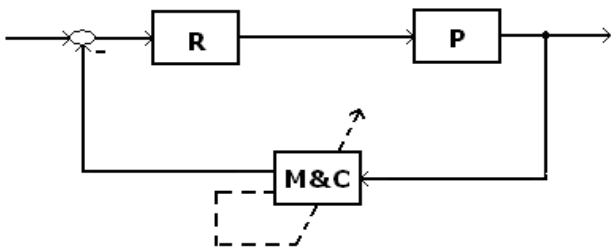
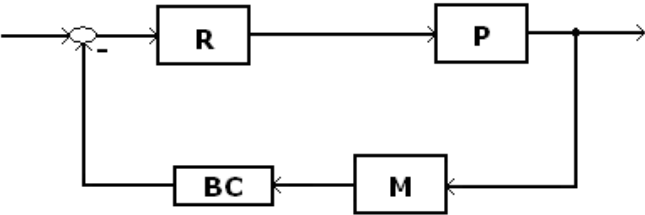
Fig. 4.3.9 Evidențierea redundanței bazate pe referință în cadrul regulatorului tolerant la defecte

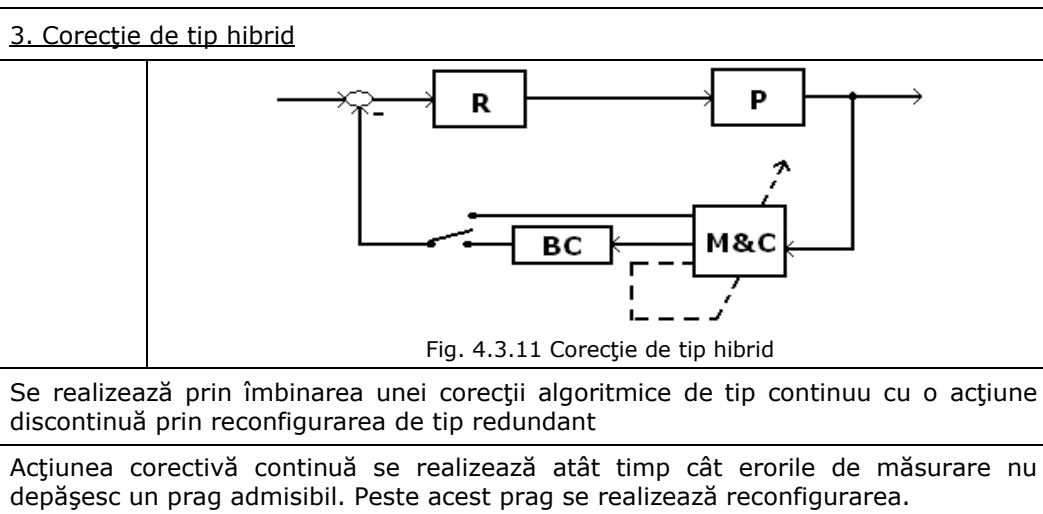
#### 4.3.2.1 Modalitatea de acțiune utilizată în cadrul blocurilor de măsurare

Senzorul de distanță folosit pentru detectarea obstacolului mobil reprezintă un element esențial pentru dependabilitatea întregului sistem. Dacă această parte a sistemului de achiziție de informații de defectează, se poate ajunge la un defect cu urmări grave. Dacă nu se pot recepționa corect informațiile referitoare la distanța față de un potențial obstacol mobil, atunci reglarea vitezei robotului în raport cu

situația curentă este eronată, pot apărea coliziuni cu efecte grave, respectiv situația în care robotul nu va ajunge la destinație sau distanța prescrisă va fi parcursă într-un timp foarte lung.

După cum s-a menționat, structura tolerantă la defecte se axează pe senzorul de distanță. În acest context, se vor prezenta principiile de acțiune (corectivă) utilizate în sistemele de măsurare. Tabelul următor prezintă aceste principii de acțiune (figurile 4.3.10a, 4.3.10b și 4.3.11) dintre care se identifică în figura 4.3.11 principiul dezvoltat în cadrul acestui paragraf. În figurile 4.3.10a, 4.3.10b și 4.3.11, ansamblurile de pe calea de reacție reprezintă grupurile formate din blocurile de măsurare și blocurile de corecție.

<u>1. Corecție de tip discontinuu</u>	
	<p>Fig. 4.3.10a Corecție de tip discontinuu (M&amp;C–bloc de măsurare și configurare)</p>
<p>Se realizează prin reconfigurare de tip redundant utilizând:</p>	<ul style="list-style-type: none"> <li>- o structură redundantă mascată</li> <li>- o structură redundantă standby</li> <li>- o structură redundantă m-n-k hibridă</li> </ul>
<p>Principiul se bazează pe operațiile de detecție, diagnoză, izolare și reconfigurare. Ultima variantă de reconfigurare este superioară celorlalte două prin modul în care asigură aceste operații [63], [64].</p>	
<u>2. Corecție de tip continuu</u>	
	<p>Fig. 4.3.10b Corecție de tip continuu (M – bloc de măsurare, BC – bloc de corecție)</p>
<p>Se realizează prin blocuri de corecție care implementează caracteristici de corecție sau algoritmi de corecție.</p>	
<p>Acest principiu este vulnerabil din punctul de vedere al adaptabilității.</p>	



Pentru implementarea blocului de corecție se disting două niveluri posibile de implementare:

- nivelul local – nivelul la care se găsesc elementele de măsură
- nivelul central – nivelul la care se găsește blocul de reglare

Tipul de implementare al blocului de corecție	La nivel local	
	La nivel central	Fără acțiune asupra nivelului local
		Cu acțiune asupra nivelului local

Conform clasificărilor anterioare, ansamblul format din schema de detecție și diagnoză a defectelor și mecanismul de corecție, adoptat în cazul de față, se situează în zona marcată din fig. 4.3.12.

Principiul de acțiune utilizat pentru sistemele de măsurare	Tipul de implementare a corecției	Corecție de tip discontinuu			Corecție de tip continuu	Corecție de tip hibrid
		Structură redundanță paralel	Structură redundanță standby	Structură redundanță m-n-k hibridă		
	Nivel local					✕
Nivel central	fără acțiune asupra nivelului local					
	cu acțiune asupra nivelului local					✕

Fig. 4.3.12 Poziția pe care se situează dezvoltarea ulterioară a celorlalte două elemente care contribuie la realizarea regulatorului tolerant la defecte (schema de detecție și diagnoză a defectelor și mecanismul de corecție a regulatorului propriu-zis)

#### 4.3.2.2 Despre principiul redundanței bazate pe referință

Există mai multe modalități prin care se poate realiza detecția și diagnoza defectelor. O modalitate este adoptarea redundanței analitice (vezi paragraful 2.1.4.8). Redundanța analitică se poate aplica doar în situațiile în care se pot obține informații analitice și funcționale despre sistem utilizând modele matematice ale structurilor din sistem. Acest tip de redundanță este utilizat în [19], [54], [74], [84]. În cazul de față este neaplicabil din cauza lipsei unui model de estimare a distanței până la obstacolul mobil, mai precis a unui model de previziune și prevenție al obstacolelor dinamice cu care robotul ar putea să intre în contact.

În consecință, dintre metodele de detecție și diagnoză a defectelor [19], [32], [74], se poate aplica doar o metodă bazată pe o referință fizică (un senzor suplimentar) cu care să se poată compara ieșirea furnizată de senzor.

În acest context se vorbește despre **redundanța bazată pe referință**. Redundanța bazată pe referință utilizează doi senzori de distanță: *senzorul bază* și *senzorul referință*. Ea va trebui să îndeplinească un dublu rol:

- determinarea defectului la senzor și furnizarea informației necesare reconfigurării on-line a regulatorului
- utilizarea senzorului suplimentar ca element de rezervă în cazul în care primul senzor nu mai este funcțional

Pentru a putea detalia conceptul de redundanță bazată pe referință este necesar să se distingă două categorii de defecte specifice senzorilor de distanță (așa cum s-a precizat în subcapitolul 1.3): *defectul brusc* și *defectul evolutiv*.

- a) Un *defect brusc* este caracterizat de o variație de la un pas de discretizare la altul a amplitudinii semnalului furnizat de senzor, mai mare decât un prag admisibil.
- b) Un *defect evolutiv* corespunde unei defectări temporizate (mai lente) a senzorului, de obicei cauzată de uzura acestuia datorită condițiilor de mediu, și care se manifestă prin furnizarea de către senzor a unui semnal atenuat sau amplificat în raport cu semnalul corect (semnalul furnizat de un element de referință). Diferența dintre semnalul furnizat de senzor în cazul unui astfel de defect și semnalul corect este considerată ca o măsură a stării de defectare. Cu cât diferența este mai mare cu atât senzorul este într-o stare mai avansată de defectare. Atâta vreme cât diferența este sub pragul admisibil (dar peste pragul de sensibilitate) acest tip de defect este tolerat (funcționare în regim de defect tolerat).

Din punctul de vedere al persistenței, defectele se clasifică în *defecte permanente* și *defecte temporare*.

Un *defect permanent* este:

- un *defect brusc* care se menține o anumită perioadă de timp (definită în prealabil), fără ca senzorul să își revină la funcționarea corectă,
- un *defect evolutiv* care iese din regimul de defect tolerat.

Un *defect temporar* se definește ca fiind un *defect brusc* care nu este catalogat ca *defect permanent*.

Cu ajutorul celor doi senzori se va determina distanța față de un obstacol mobil care apare în plaja de observabilitate a robotului, regulatorul trebuind să controleze, în funcție de informația furnizată de senzori, viteza sistemului, pentru ca acesta să aibă o deplasare corectă și sigură.

În funcție de stările senzorului bază și senzorului referință se definesc două regimuri de funcționare ale întregului sistem:

- *regimul normal* – caracterizat prin următoarele aspecte:
  - *senzorul bază* nu este afectat de un defect permanent și este utilizat la fiecare pas de eșantionare pentru determinarea distanței față de un eventual obstacol mobil întâlnit în „raza vizuală” a robotului;
  - *senzorul referință* se utilizează pentru măsurare din  $k$  în  $k$  pași de eșantionare;
  - eroarea de măsurare a senzorului bază față de senzorul referință este determinată din  $k$  în  $k$  pași de eșantionare și este folosită pentru a determina dacă la nivelul senzorului bază a apărut un *defect evolutiv*. Eroarea de măsurare este folosită pe parcursul următorilor pași pentru a corecta (în cadrul mecanismului de corecție), prin intermediul unui factor de corecție, valoarea distanței furnizată de senzorul bază. Evident, dacă ieșirile senzorilor coincid între anumite limite, factorul de corecție este unitar.
- *regimul de funcționare în stare de siguranță (safe-mode)* caracterizat prin următoarele aspecte (se menționează faptul că starea de siguranță reprezintă o stare a sistemului în care a apărut o defecțiune și sistemele de protecție au acționat):
  - la nivelul *senzorului bază* s-a detectat un defect permanent.
  - *senzorul referință* acționează într-o manieră redundantă, înlocuind *senzorul bază* care nu mai este utilizat. Ca urmare, în acest regim de funcționare nu mai există referință și se pierde astfel capacitatea de a detecta un *defect evolutiv*. În consecință, viteza de deplasare a sistemului mecanic este redusă.
  - în cazul apariției unui *defect brusc* la nivelul senzorului rămas, care este catalogat *defect permanent*, sistemul se oprește.

Primul rol al structurii redundante bazată pe referință este de a detecta cele două tipuri de defecte care pot apărea la senzori (defect brusc sau defect evolutiv) și de a furniza în eventualitatea unui defect evolutiv la *senzorul bază* informația necesară mecanismului de corecție, precum și de a determina și semnala defectele permanente. Astfel, primul rol al structurii redundante revine la îndeplinirea următoarelor funcții:

- i) Determinarea unui defect evolutiv și transmiterea informației necesare (factorul de corecție  $\alpha$  – vezi figura 4.3.9) mecanismului de corecție în vederea obținerii toleranței la defecte. Operația se realizează în cadrul regimului normal de funcționare, periodic, din  $k$  în  $k$  pași de eșantionare. Valoarea lui  $k$  este corelată cu valoarea lui  $h$  (pasul de discretizare) în ideea că intervalul de timp  $kh$  este suficient de mic pentru ca pe parcursul lui să poată fi tolerate modificări ale caracteristicii traductorului.

Figurile 4.3.13 și 4.3.14 simbolizează senzorul bază și senzorul referință. Senzorul referință (care printr-o utilizare mai rară se păstrează într-o stare de funcționare corectă) furnizează un semnal  $e_r$ , care se compară cu semnalul furnizat de senzorul bază  $e_b$ .

$$\alpha = \frac{e_b}{e_r} \quad (8)$$

$\alpha$  este factorul de corecție. Dacă  $\alpha \neq 1$  și  $\alpha \in [\alpha_{lim1}, \alpha_{lim2}]$ ,  $\alpha_{lim1}$  și  $\alpha_{lim2}$  fiind limitele prestabilite, atunci s-a detectat un defect evolutiv și factorul este transmis mecanismului de corecție.



Fig. 4.3.13 Senzorul bază

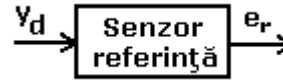


Fig. 4.3.14 Senzorul referință

Pentru a evita nedeterminările în calculul factorului de corecție (o eventuală atingere a nivelului zero de către numitorul formulei (8)) se adoptă și se setează o valoare  $e_{\min}$ . Dacă  $e_r < e_{\min}$  atunci pentru a determina factorul de corecție se va utiliza valoarea  $e_{\min}$ .

- ii) Determinarea unui defect brusc se realizează cu ajutorul redundanței de timp (vezi paragraful 2.1.4.6). Valoarea curentă  $e[t]$  (în regimul normal de funcționare  $e=e_b$ , iar în safe-mode  $e=e_r$ ) a semnalului furnizat de senzor este comparată cu valoarea anterioară  $e[t-1]$ .

$$|e[t] - e[t-1]| > p_a \quad (9)$$

Dacă inegalitatea (9) este satisfăcută ( $p_a$  - valoarea prestabilită a pragului admisibil), atunci a avut loc o variație de o amplitudine mare a semnalului, detectându-se un defect brusc.

Detecția defectului brusc se realizează atât în regimul normal, cât și în regimul de funcționare în stare de siguranță.

- iii) Determinarea unui defect permanent se realizează în două moduri, în funcție de tipul defectului:

- a) în cazul unui defect evolutiv la nivelul senzorului bază, prin determinarea momentului în care factorul de corecție iese dintre limitele prestabilite.
  - b) în cazul unui defect brusc, prin utilizarea redundanței de timp.
- Dacă inegalitatea

$$|e[t+i] - e[t-1]| > p_a, \quad i=\overline{1, n_p} \quad (10)$$

este adevărată pentru următorii  $n_p$  pași de eșantionare atunci se declară un defect permanent la nivelul senzorului analizat.

Al doilea rol al structurii redundante bazată pe referință este de a utiliza senzorul referință de o manieră redundantă, ca element de rezervă, în cazul în care senzorul bază nu mai este funcțional. Dacă se diagnostichează un defect permanent la nivelul senzorului bază, atunci se intră în starea de siguranță și senzorul referință va prelua toate sarcinile senzorului defect.

Redundanța bazată pe referință este un tip de redundanță care are ca avantaje costul redus și consumul mic de energie (se utilizează doar doi senzori în comparație cu alte structuri redundante de o complexitate ridicată). Consumul mic de energie este important în cazul roboților mobili care funcționează pe baterii.

Redundanța bazată pe referință este un tip de redundanță care are în componență și elemente din redundanța de timp, sau din redundanța standby hot (vezi paragraful 2.1.4.4).

Caracterul dinamic al redundanței bazate pe referință se păstrează până în momentul în care se intră în safe-mode (funcționare în stare de siguranță), situație în care redundanța de timp va permite doar evitarea defectelor bruște care nu au un caracter permanent.



### 4.3.2.3 Algoritm de funcționare al redundanței bazate pe referință

Arhitectura structurii redundante bazate pe referință este prezentată în figura 4.3.15, iar o schemă logică generală de funcționare nedetaliată este ilustrată în figura 4.3.16. Cele două figuri sunt explicate în cele ce urmează.

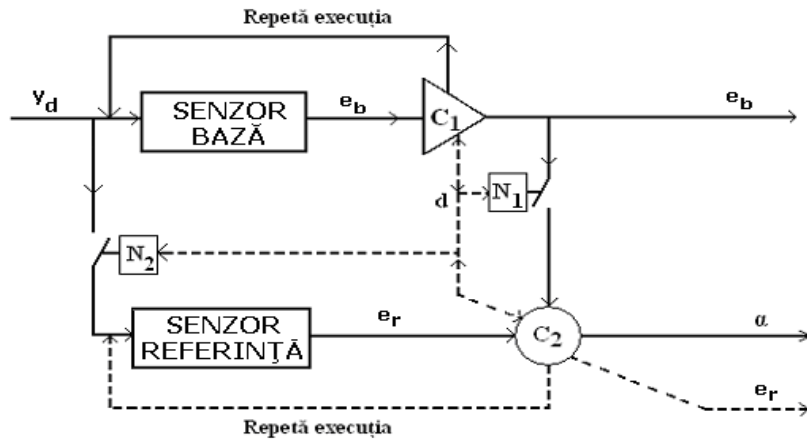


Fig. 4.3.15 Schemă funcțională a redundanței bazate pe referință

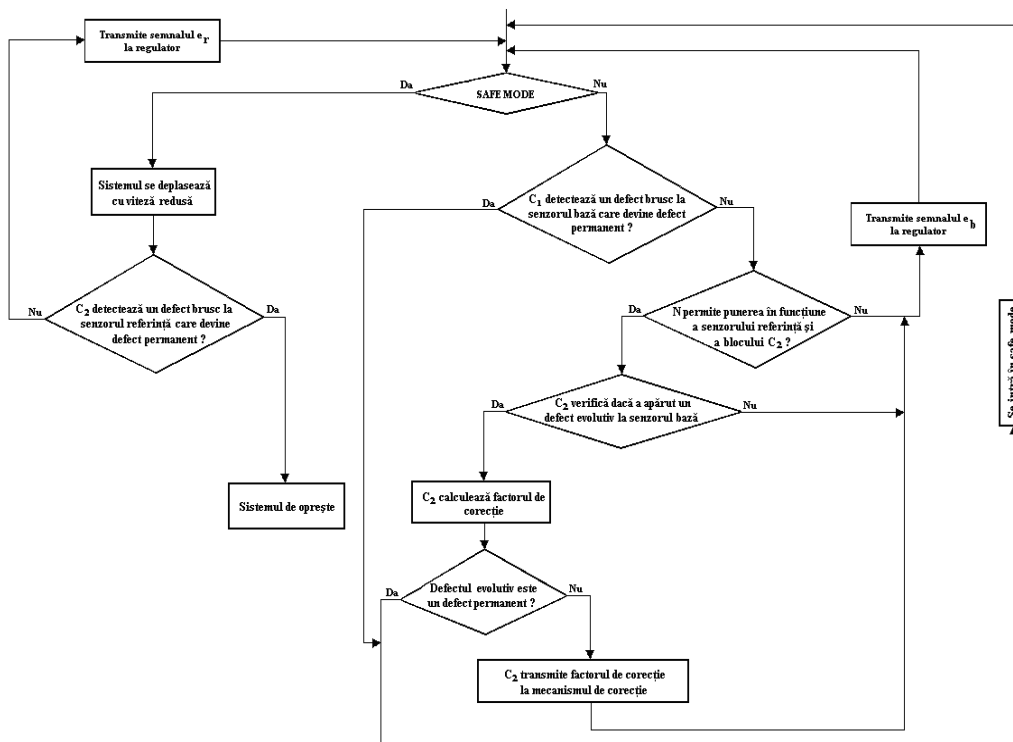


Fig. 4.3.16 Organigramă pentru funcționarea redundanței bazată pe referință

Semnalul  $y_d$  reprezintă intrarea sensorului de distanță, semnalele de ieșire  $e_b$ ,  $e_r$  se transmit la regulator, iar  $\alpha$  se transmite la mecanismul de corecție.

Notațiile sunt următoarele:

- SENZOR BAZĂ – reprezintă sensorul bază, folosit pentru obținerea distanței față de obstacolul mobil
  - $e_b$  – este semnalul furnizat de sensorul bază, iar în situația în care sistemul este în regim normal de funcționare, acest semnal va fi transmis la regulator
  - SENZOR REFERINȚĂ – reprezintă sensorul referință, sensorul suplimentar folosit pentru a realiza două tipuri de acțiuni cu caracter redundant:
    - a) detectarea unui defect evolutiv și totodată calcularea și furnizarea factorului de corecție mecanismului de corecție pentru a se dobândi toleranța la defecte
    - b) utilizarea ca element de rezervă pentru funcționarea în regimul safe-mode, după apariția unui defect permanent la nivelul sensorului bază, preluând rolul acestuia
  - $N_1$  și  $N_2$  – reprezintă numărătoare care funcționează în două moduri:
    - a) în cazul regimului de funcționare normal, din  $k$  în  $k$  pași de eșantionare determină utilizarea sensorului referință (și astfel transmiterea semnalului  $e_r$  spre blocul  $C_2$ ), respectiv transmiterea spre blocul  $C_2$  a semnalului  $e_b$ .
    - b) în cazul funcționării în stare de siguranță, blocul  $N_1$  și  $N_2$  sunt în permanentă stare de resetare, adică structura redundantă se bazează exclusiv pe singurul senzor rămas funcțional, sensorul referință.
- În figura 4.3.17 este redată schema logică după care funcționează blocurile  $N$ .
- $d$  – este semnalul furnizat de:
    - a) blocul  $C_1$  în situația în care sensorul bază suferă un defect brusc care în final e catalogat defect permanent. Acest semnal este transmis blocurilor  $N$  și blocului  $C_2$  pentru a intra în safe-mode.
    - b) blocul  $C_2$  în situația în care la nivelul sensorului bază apare un defect permanent provenit în urma unui defect evolutiv care iese din limitele admise prestabilite. Semnalul  $d$  este transmis blocurilor  $N$  și  $C_1$  pentru a indica intrarea în funcționarea în stare de siguranță.
  - $\alpha$  – este factorul de corecție transmis mecanismului de corecție al regulatorului interpolativ propriu-zis, conferindu-i acestuia toleranță la defecte
  - $e_r$  – reprezintă semnalul furnizat de sensorul referință blocului  $C_2$  pentru a se putea calcula factorul de corecție  $\alpha$ . De asemenea, în safe-mode,  $e_r$  este semnalul care se transmite regulatorului
  - Repetă execuția – este o reprezentare simbolică a redundanței de timp și înseamnă păstrarea valorii de la pasul precedent a semnalului pentru a putea fi comparată cu valoarea curentă preluată. În situația în care apare un defect brusc, pentru a putea elabora o diagnoză finală și a identifica un eventual defect permanent, ultima valoare corectă se păstrează pentru o comparație cu valori ale semnalului la următorii  $n_p$  pași de eșantionare.
  - Linia întreruptă – reprezintă starea de siguranță (safe-mode).

Safe mode este starea în care intră sistemul în situația în care la nivelul sensorului bază apare un defect permanent. În acest caz, sensorul referință va acționa într-o manieră redundantă (ca element de rezervă), preluând toate atribuțiile elementului defect. La intrarea în starea de siguranță sensorul referință va acționa identic sensorului bază dinainte de defect. Problema care apare este că sistemul redundant rămas va fi de tip static, iar regulatorul încetează a mai fi tolerant la defecte din moment ce  $\alpha$  (factorul de corecție al regulatorului) va fi setat în permanență pe valoarea *red*. Pentru a evita unele avarii ale sistemului, care pot apărea în urma deteriorării sensorului rămas, viteza robotului mobil este redusă. În cazul în care se detectează un defect permanent la sensorul rămas în timpul evoluției sistemului în starea de siguranță, întregul sistem va fi oprit.

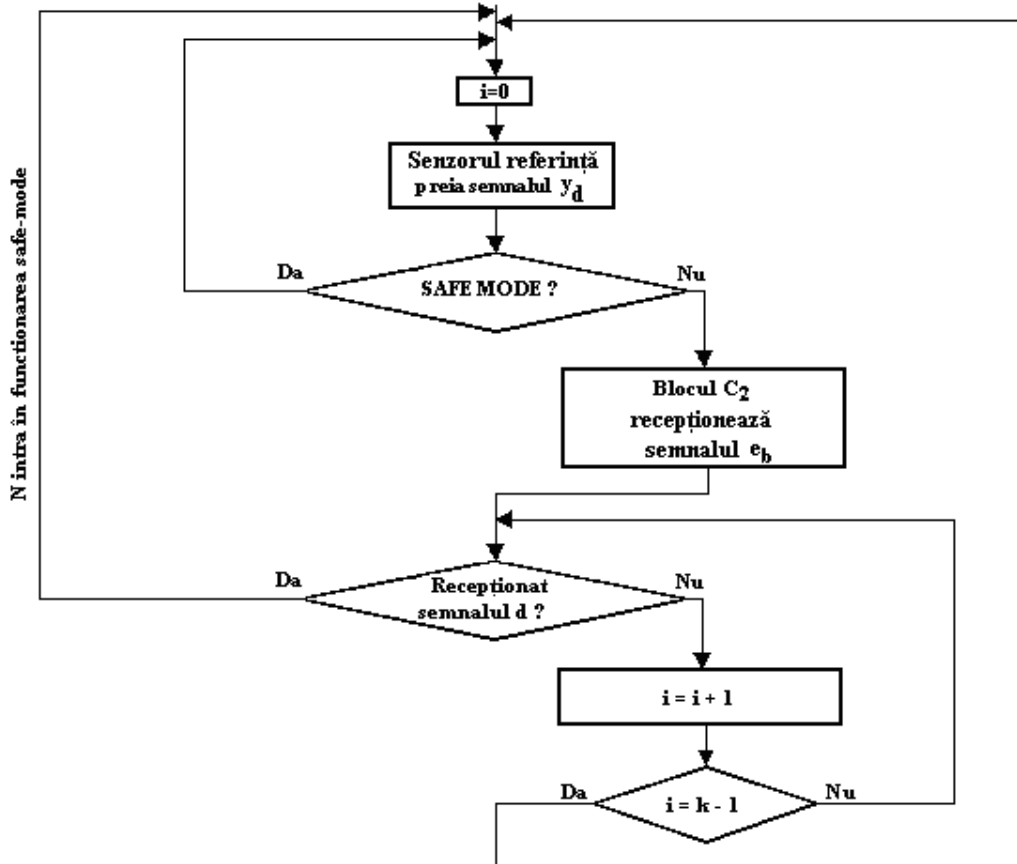


Fig. 4.3.17 Schema logică aferentă blocurilor N

- $C_1$  – blocul  $C_1$  se utilizează în cazul regimului normal de funcționare al sistemului. Acesta preia valoarea curentă a semnalului furnizat de sensorul bază și determină pe baza relației (9) dacă la nivelul acestui sensor a apărut un defect brusc. În intervalul de timp imediat următor detectării defectului brusc, defectul este catalogat de către blocul  $C_1$  drept defect temporar și se transmite regulatorului ultima valoare corectă a semnalului  $e_b$ , valoarea  $e_b[t-1]$ . În continuare, tot prin

redundanța de timp,  $C_1$  verifică pe baza relației (10) dacă defectul temporar devine permanent. Acest procedeu a fost de asemenea prezentat. Dacă  $C_1$  identifică un defect permanent la nivelul sensorului bază atunci el transmite semnalul  $d$  blocurilor  $N$  și  $C_2$ , prin care se comandă intrarea sistemului în regimul safe-mode.

Schema logică din figura 4.3.18 prezintă modul de funcționare al blocului  $C_1$ . În această figură:

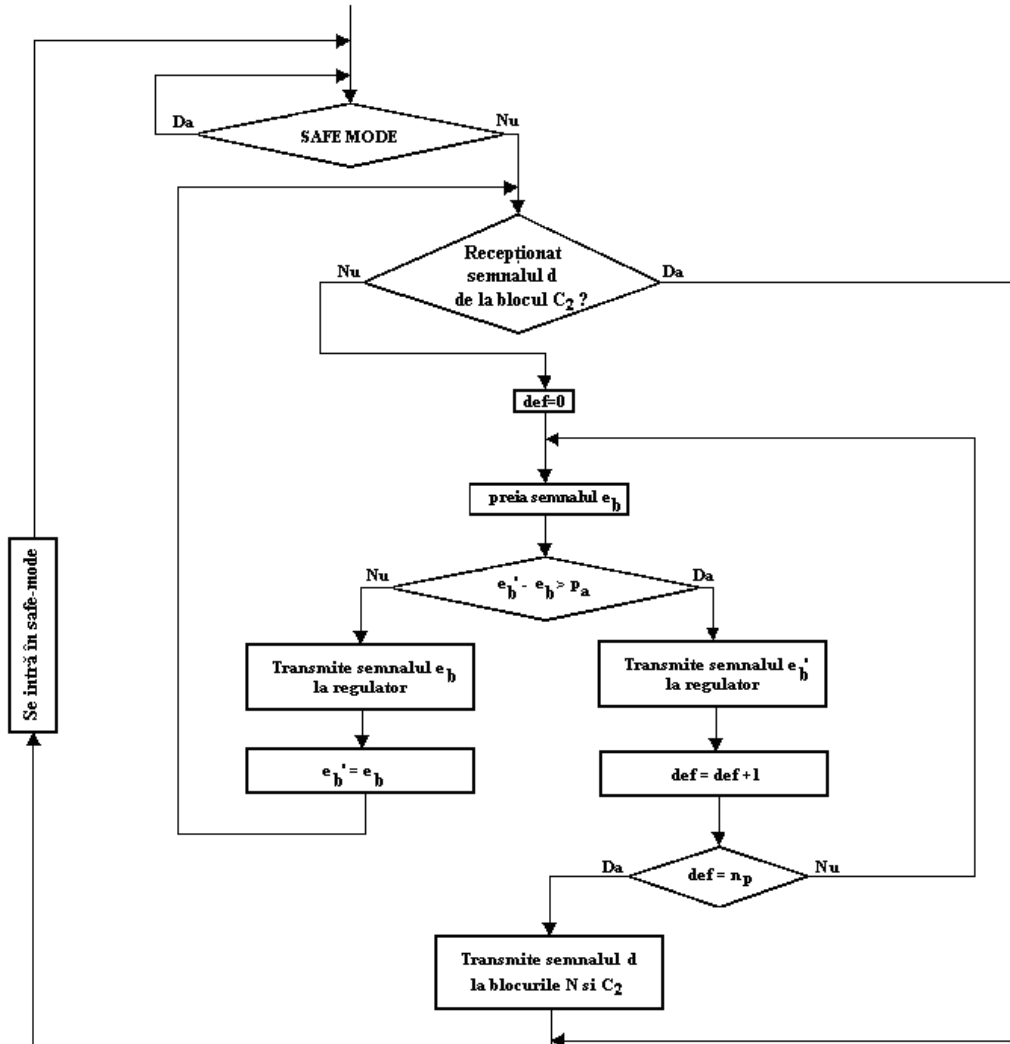


Fig. 4.3.18 Schema logică a funcționării blocului  $C_1$

- $e_b'$  - valoarea semnalului furnizat de sensorul bază și preluat de către blocul  $C_1$  în pasul anterior de eșantionare
- $def$  - variabilă folosită pentru a putea evita defectele temporare (un contor)

- $n_p$  – numărul de pași de eșantionare stabiliți pentru a putea determina un defect permanent (în cazul aplicației curente  $n_p$  va fi setat pe valoarea 2)
  - $p_a$  – pragul admisibil prestabilit pentru saltul brusc, de o amplitudine neacceptată, făcut de semnalul furnizat de senzor în caz de defect brusc
- $C_2$  – Blocul  $C_2$  acționează în două moduri:
- a) pentru regimul normal de funcționare blocul  $C_2$  intră în acțiune la fiecare  $k$  pas de eșantionare (vezi fig. 4.3.17). Preia semnalul  $e_b$  de la senzorul bază și semnalul  $e_r$  furnizat de senzorul referință. Calculează în funcție de cele două semnale factorul de corecție  $\alpha$  (vezi relația (8)). Dacă  $\alpha$  este diferit de 1 atunci se identifică un defect evolutiv. Se verifică dacă  $\alpha \in [\alpha_{lim_1}, \alpha_{lim_2}]$  (limitele stabilite inițial), iar în caz afirmativ, acesta se furnizează mecanismului de corecție. Dacă  $\alpha$  iese din limitele admise atunci se identifică un defect permanent la nivelul senzorului bază. În această situație se transmite semnalul numeric  $d$  blocurilor  $C_1$  și  $N$ , intrându-se în regimul de funcționare safe-mode.
  - b) pentru funcționarea în stare de siguranță, primul pas este de a modifica valoarea factorului de corecție  $\alpha$  și de a-l seta pe o valoare fixă. În safe-mode, viteza sistemului se va reduce pentru a prezenta siguranță sporită în funcționare, iar factorului  $\alpha$  i se atribuie valoarea *red*, care va fi un număr supraunitar în funcție de gradul de reducere a vitezei. Factorul de corecție astfel setat este transmis mecanismului de corecție cu scopul de a reconfigura regulatorul pentru funcționarea în safe-mode.

În funcționarea în stare de siguranță blocul  $C_2$  acționează pentru a identifica un defect brusc, respectiv de a determina dacă acel defect brusc devine un defect permanent (după algoritmul ilustrat pentru blocul  $C_1$ ). În regimul safe-mode, blocul  $C_2$  va furniza regulatorului în permanență semnalul  $e_r$ . În caz de defect permanent la senzorul rămas, sistemul se va opri.

Schema logică din fig. 4.3.19 ilustrează modul de funcționare al blocului  $C_2$ .

Aici:

- *red* – coeficient supraunitar de reducere a vitezei în safe-mode. Atribuindu-i factorului de corecție  $\alpha$  o valoare *red* supraunitară, se va putea realiza o reducere a vitezei robotului mobil. Dacă acest coeficient ar fi setat pe valoarea 1, atunci viteza nu ar fi redusă. Se va reveni asupra acestui subiect în paragraful următor în cadrul mecanismului de corecție a regulatorului.
- $lim_1, lim_2$  – limitele extreme acceptate pentru factorul de corecție  $\alpha$ . Dacă  $\alpha$  depășește aceste limite, înseamnă că la nivelul senzorului bază s-a produs un defect permanent și nu mai este fondată utilizarea lui.

Redundanța bazată pe referință, oferă regulatorului posibilitatea reconfigurării/corecției. Cu ajutorul elementelor furnizate de redundanța bazată pe referință împreună cu mecanismul de corecție al regulatorului, senzorul bază poate fi utilizat cu succes în pofida unui defect evolutiv la nivelul acestuia. Regulatorul este tolerant la defecte.

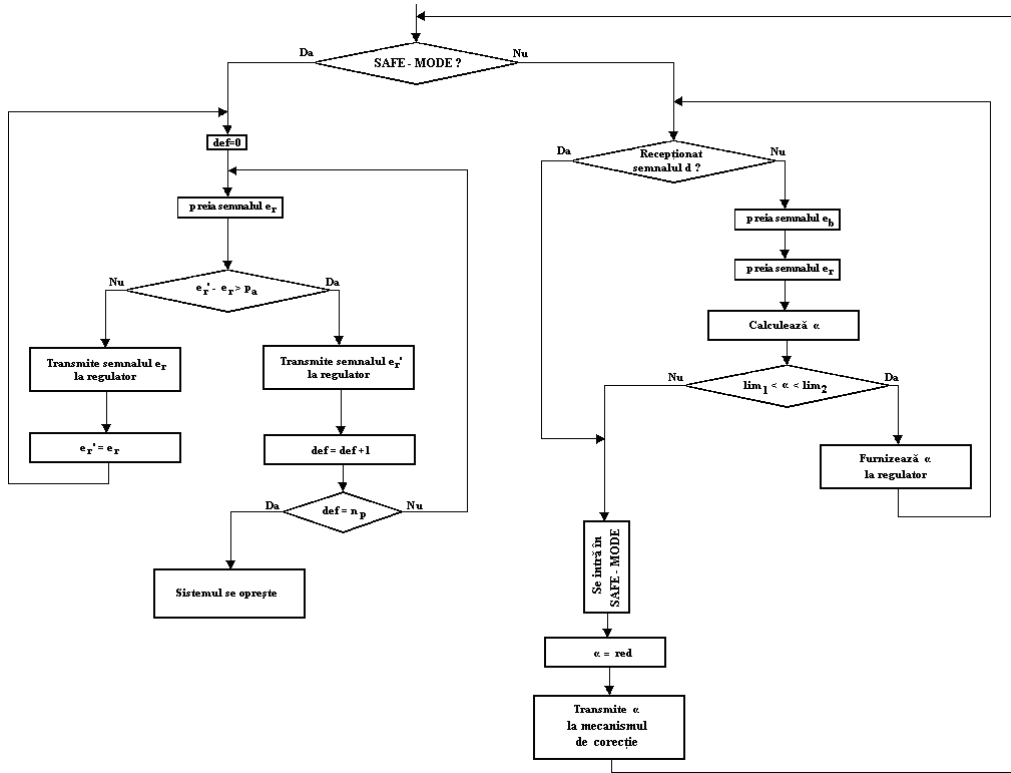


Fig. 4.3.19 Schema logică a funcționării blocului C<sub>2</sub>

### 4.3.3 Mecanismul de corecție al regulatorului

Conceperea mecanismului de corecție al regulatorului reprezintă ultima fază în obținerea regulatorului tolerant la defecte. Regulatorul tolerant la defecte este prezentat în fig. 4.3.20.

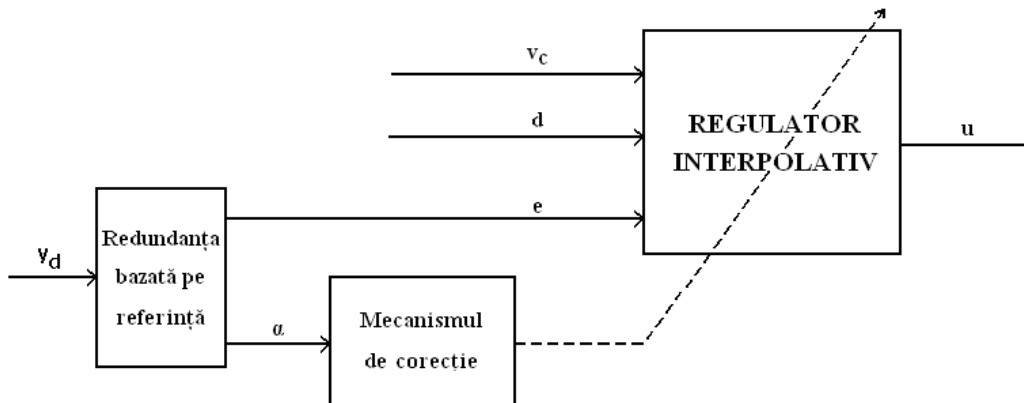


Fig. 4.3.20 Schema regulatorului tolerant la defecte

- $v_c$  – reprezintă viteza curentă
- $d$  – reprezintă distanța
- $e$  – este distanța față de obstacolul mobil
- $\alpha$  – factorul de corecție al regulatorului
- $y_d$  – intrarea structurii redundante
- $u$  – mărimea de ieșire a regulatorului interpolativ

Elemente teoretice referitoare la regulatorul interpolativ, metoda, notațiile și formulele de calcul pentru interpolarea bazată pe tabele tridimensionale, au fost prezentate în paragraful 4.1.2. Tabelul de interpolare utilizat în aplicația curentă este tridimensional (linii, coloane, fețe).

În fiecare pas de calcul se face, într-o primă etapă, o căutare între locurile de sprijin pentru a găsi locul de sprijin esențial în funcție de cele trei intrări  $v_c$ ,  $d$  și  $e$ . Cu ajutorul locului de sprijin esențial se determină valoarea de sprijin esențială  $R_{i,j,k}$ . Astfel, se identifică punctul de sprijin esențial  $P_{i,j,k} = (V_{1i}, V_{2j}, V_{3k}, R_{i,j,k})$ .

În urma căutării, într-o a doua etapă, se trece la calculele propriu-zise în concordanță cu formulele prezentate în cadrul paragrafului 4.1.2. După ce se realizează interpolările bidimensionale pe fețele  $k$  și  $k-1$ , cu rezultatele  $R_1$  și  $R_2$ , se trece la o interpolare a acestor două rezultate. Formula care realizează această ultimă operație este formula (7) din paragraful 4.1.2 și include locurile de sprijin pe fețe, locuri de sprijin alese din domeniul de intrare pe fețe al regulatorului interpolativ propriu-zis (corespunzător intrării a treia a regulatorului, intrarea  $e$ , distanța față de elementul perturbator furnizată de redundanța bazată pe referință). Prin adaptarea formulei (7) la notațiile din acest paragraf se obține relația (11) (se menționează că în regimul normal de funcționare  $e=e_b$ , iar în safe-mode  $e=e_r$ ):

$$R = R_2 + \frac{e - V_{3k-1}}{V_{3k} - V_{3k-1}} \cdot (R_1 - R_2) \quad (11)$$

Dacă senzorul bază suferă un defect evolutiv rezultă că distanța față de obstacolul mobil nu ar putea fi estimată cu exactitate de către acest element. Această eroare de estimare ar putea conduce la avarii majore ale sistemului, sau la o funcționare eronată (un timp de funcționare prea lung, sau chiar oprirea sistemului). Defectul evolutiv de la nivelul senzorului bază ar implica o viteză prescrisă impusă de regulator mai mare sau mai mică decât cea corectă, din cauză că regulatorul ar primi informația eronată că obstacolul mobil se află mai aproape (implicit, viteza prescrisă de regulator ar fi mai mică, mărindu-se astfel timpul în care robotul ar ajunge la destinație, robotul putându-se chiar opri) sau mai departe (implicit, viteza prescrisă de regulator nu va fi redusă iar sistemul riscă avarierea gravă prin coliziunea cu obstacolul dacă acesta se găsește pe direcția sa de deplasare) decât în realitate.

În regimul normal de funcționare, redundanța bazată pe referință stabilește relația dintre  $e_b$  și  $e_r$ ,  $e_r$  reprezentând mărimea corectă furnizată de senzorul referință, iar  $e_b$  mărimea furnizată de senzorul bază. Astfel, conform relației (8) se obține:

$$e_b = e_r \cdot \alpha \quad (12)$$

Se presupune că mecanismul de corecție nu acționează și că la nivelul senzorului bază apare un defect evolutiv. Atunci regulatorul furnizează un semnal eronat. Eroarea rezultatului interpolării poate avea două forme în funcție de hipercubul în care este „amplasat” rezultatul măsurării.

- dacă intrarea eronată  $e_b$  se situează în hipercubul în care s-ar situa și intrarea reală, atunci situația este ilustrată de figura 4.3.21.

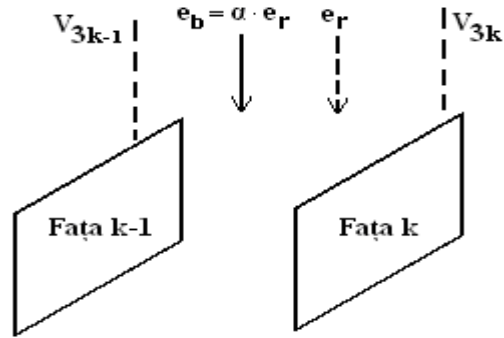


Fig. 4.3.21 Semnalul eronat și cel corect se folosesc de aceleași puncte de sprijin (indică spre același hipercub)

Eroarea rezultată în urma interpolării este:

$$R_{corect} - R_{eronat} = \frac{(R_{k-1} - R_k) \cdot e_r}{V_{3k} - V_{3k-1}} \cdot (1 - \alpha) \quad (13)$$

- dacă intrarea eronată  $e_b$  se situează într-un alt hipercub decât cel în care s-ar situa și intrarea reală, atunci situația este ilustrată de figura 4.3.22.

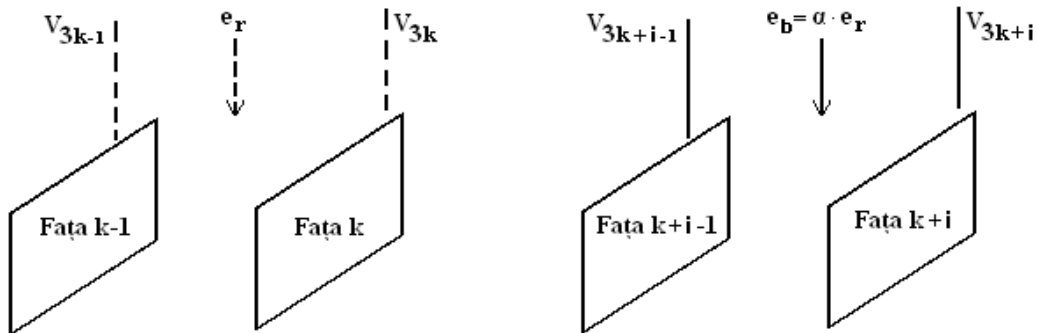


Fig. 4.3.22 Semnalul eronat și cel corect se folosesc de puncte de sprijin diferite

Eroarea rezultată în urma interpolării este:

$$R_{corect} - R_{eronat} = (R_k - R_{k+i}) + \frac{e_r - V_{3k-1}}{V_{3k} - V_{3k-1}} \cdot (R_{k-1} - R_k) - \frac{\alpha \cdot e_r - V_{3k+i-1}}{V_{3k+i} - V_{3k+i-1}} \cdot (R_{k+i-1} - R_{k+i}) \quad (14)$$

Factorul de corecție furnizat de către redundanța bazată pe referință mecanismului de corecție determină toleranța la defecte a regulatorului interpolativ. Regulatorul tolerant la defecte evită situațiile nedorite prin capacitatea sa de a impune viteza prescrisă corectă chiar și în situațiile în care senzorul bază suferă un defect evolutiv. Starea de funcționare corectă a regulatorului tolerant la defecte este dată de proprietatea sa de reconfigurare/corecție.



Corecția este interpretată ca o modificare a locului de sprijin curent în tabelul de interpolare. Modificând corespunzător locurile de sprijin în cadrul tabelului de interpolare se obține același rezultat al interpolării ca și în cazul unei intrări corecte. Astfel, viteza impusă de regulator este una corespunzătoare și sistemul funcționează la capacitate maximă chiar dacă apare un defect evolutiv la senzorul bază.

Figura 4.3.23 ilustrează interpretarea acțiunii de corecție (ajustarea realizată prin blocul de corecție este interpretată ca o modificare a locului de sprijin pe fețe) prin modificările care survin la tabelul de interpolare. Cu linie întreruptă este ilustrată situația în cazul unei funcționări corecte (raportată la senzorul referință), iar cu linie continuă este prezentată situația în care la nivelul senzorului bază apare un defect evolutiv.

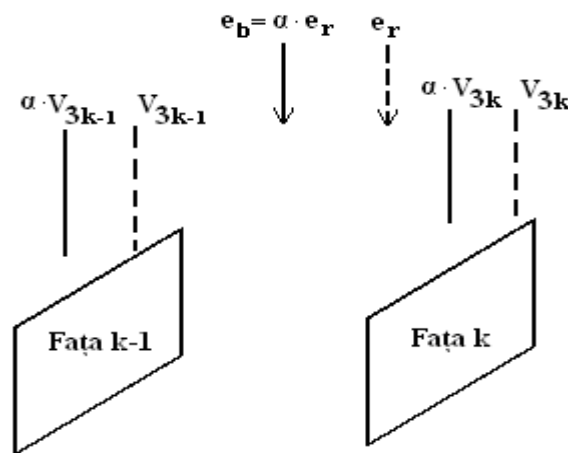


Fig. 4.3.23 Ajustările realizate asupra locurilor de sprijin din tabelul de interpolare în urma apariției unui defect evolutiv

În consecință, mecanismul de corecție nu acționează asupra valorilor de sprijin, ci doar asupra locului de sprijin.

Mecanismul de corecție este reprezentat în fig. 4.3.11 prin blocul BC. Corecția automată on-line a locului de sprijin pe fețe ale tabelului de interpolare se realizează prin factorului de corecție din cadrul blocului de corecție BC din fig. 4.3.11. Factorul de corecție al regulatorului fiind  $\alpha$ , în cadrul blocului de corecție BC din fig. 4.3.11 se introduce  $1/\alpha$ .

Regulatorul tolerant la defecte se încadrează între abordările active și anume apelează la metoda reconfigurării/corecției automate on-line a acestuia, implicând proceduri de corecție a parametrilor regulatorului la apariția defectului, respectiv reconfigurarea structurii redundante la apariția unui defect permanent.

Dacă defectul evolutiv nu se încadrează în limitele stabilite de acceptabilitate sistemul intră în funcționarea safe-mode.

După cum s-a mai specificat, funcționarea în starea de siguranță a sistemului s-a denumit *regim safe-mode*. Regimul a fost descris prin prisma redundanței bazate pe referință în subcapitolul anterior, sistemul ajungând în acest regim:

- în urma unui defect permanent rezultat dintr-un defect brusc la nivelul senzorului bază

- în urma unui defect permanent rezultat dintr-un defect evolutiv la nivelul sensorului bază (dacă factorul de corecție  $\alpha$  iese din limitele stabilite inițial)

În continuare se specifică funcționarea în starea de siguranță prin prisma mecanismului de corecție.

În safe-mode structura redundantă nu mai are sensorul suplimentar, deci factorul de corecție  $\alpha$  nu se mai poate calcula, respectiv regulatorul își pierde proprietatea de toleranță la defecte. După cum s-a specificat în figura 4.3.13 din subcapitolul anterior, în funcționarea în stare de siguranță, factorului de corecție  $\alpha$  i se atribuie valoarea *red*, pentru a reduce viteza sistemului. Tot în subcapitolul anterior s-au prezentat situațiile care pot apărea în cazul unui defect evolutiv la nivelul sensorului care furnizează distanța față de obstacolul mobil. În prima situație (dacă distanța furnizată față de elementul perturbator este prea mare) defectul sensorului poate duce la avarii grave, coliziunea cu obstacolul mobil. A doua situație nu este atât de gravă, defectul neafectând fizic robotul, doar timpul de realizare a sarcinii (timpul de execuție se poate mări considerabil, se poate ajunge la situația opririi întregului sistem înainte de a îndeplini sarcina). În funcționare în stare de siguranță un defect evolutiv nu mai poate fi determinat și pentru a evita în special prima situație menționată mai sus, trebuie redusă viteza robotului mobil. Astfel, în eventualitatea unui defect evolutiv la nivelul rămas, se poate evita avarierea robotului. Reducerea vitezei sistemului se poate realiza setându-l pe  $\alpha$  în mod corespunzător, atribuindu-i valoarea *red* supraunitară. Cu cât valoarea *red* este mai mare, cu atât viteza sistemului se va reduce în mai mare măsură.

Oprirea forțată a sistemului în cazul funcționării în stare de siguranță se poate produce în cazul unui defect brusc la nivelul sensorului rămas, care se cataloghează drept defect permanent.

#### 4.3.4 Experimente

Regulatorul interpolativ propriu-zis cu trei intrări al robotului mobil dezvoltat în paragraful 4.3.1 (rezultatele experimentale au fost prezentate în figurile 4.3.6, 4.3.7, 4.3.8), împreună cu redundanța bazată pe referință și mecanismul de corecție, prezentate în paragrafele 4.3.2 și 4.3.3, configurează împreună un regulator interpolativ tolerant la defecte (vezi fig. 4.3.1).

În continuare, prezentarea se va referi la experimente efectuate cu regulatorul interpolativ tolerant la defecte, iar rezultatele se vor ilustra sub formă de grafice. Parțial, problema a făcut obiectul lucrării [66]. Studiul are menirea de a demonstra eficiența regulatorului interpolativ tolerant la defecte. Prin eficiență se înțelege toleranță la defecte, disponibilitate ridicată și siguranță în funcționare, caracterizate prin următoarele:

- robotul mobil funcționează la capacitate maximă în pofida unui defect evolutiv sau brusc apărut la nivelul sensorului bază prin metoda de corecție asupra regulatorului interpolativ propriu-zis,
- robotul mobil are o evoluție sigură în pofida unui defect permanent la nivelul sensorului bază prin reconfigurarea structurii redundante bazată pe referință,
- robotul mobil se oprește în momentul unui defect permanent apărut în urma unui defect brusc la nivelul sensorului referință.

Scenariile concepute urmăresc să demonstreze evoluția robotului cu regulator interpolativ tolerant la defecte într-un spațiu semicunoscut unde pot apărea obstacole mobile.

Sistemul va fi supus la cinci experimente :

- primele două testează situațiile în care apare un defect evolutiv la senzorul bază dar nu se produce un defect permanent (factorul de corecție  $\alpha$  se află în plaja acceptabilă, situându-se între limitele stabilite)
- următoarele două experimente analizează situația în care senzorul bază suferă un defect permanent provenit în urma:
  - a) unui defect brusc
  - b) unui defect evolutiv

În această situație se intră în safe-mode și implicit senzorul referință asigură singur funcționarea sistemului.

- ultimul experiment urmărește defectarea permanentă a ambilor senzori, ceea ce implică oprirea întregului sistem.

Ipotezele experimentale prezentate în subcapitolul 4.3.1 se păstrează pentru a putea ilustra prin comparație eficiența regulatorului interpolativ tolerant la defecte.

1) Primul experiment urmărește un *defect evolutiv la nivelul senzorului bază, iar în consecință factorului de corecție  $\alpha$  se micșorează, dar rămâne în limitele specificate (defectul nu este permanent).*

Fig. 4.3.24 ilustrează distanța reală a robotului față de obstacolul mobil în funcție de timp. Se observă că la momentul  $t=4,7s$  în raza de vizibilitate a robotului ( $<3m$ ) intră un obstacol mobil care se apropie de acesta până la  $1m$  (momentul  $t=6,5s$ ), iar apoi, începând cu momentul  $t=10s$ , se îndepărtează dispărând din raza de vizibilitate la momentul  $t=12s$ . Figura 4.3.25 ilustrează modul în care senzorul bază, defect de la momentul  $t=5s$ , redă această situație, iar figura 4.3.26 modul în care mecanismul de corecție acționează prin modificarea valorii factorului  $\alpha$ .

Prin redundanța bazată pe referință și prin mecanismul de corecție automată on-line, se realizează o corecție asupra regulatorului tolerant la defecte și sistemul de reglare își îndeplinește funcția, în continuare, în mod corect. Acest lucru se observă comparând figurile 4.3.27 și 4.3.28, care reprezintă evoluția vitezei robotului mobil și a distanței parcurse de acesta în timp, cu figurile 4.3.7 și 4.3.8 din paragraful 4.3.1. În urma comparațiilor rezultă că defectul evolutiv nu afectează evoluția robotului mobil.

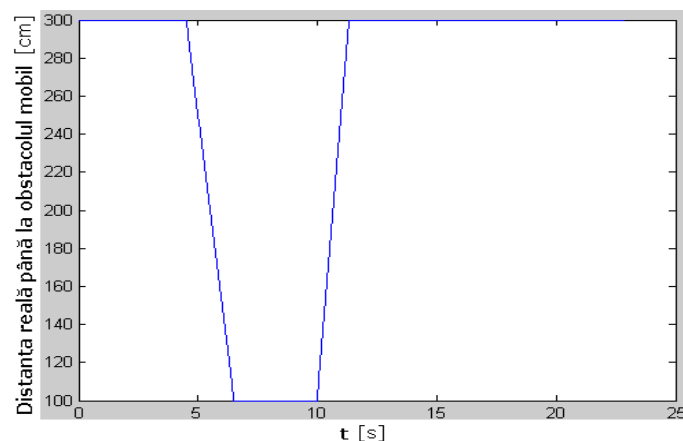


Fig. 4.3.24 Distanța reală a robotului față de obstacolul mobil

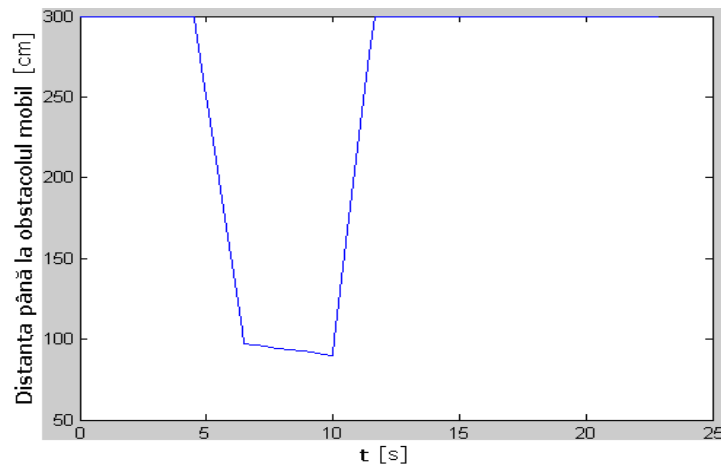


Fig. 4.3.25 Distanța față de obstacolul mobil furnizată de senzorul bază

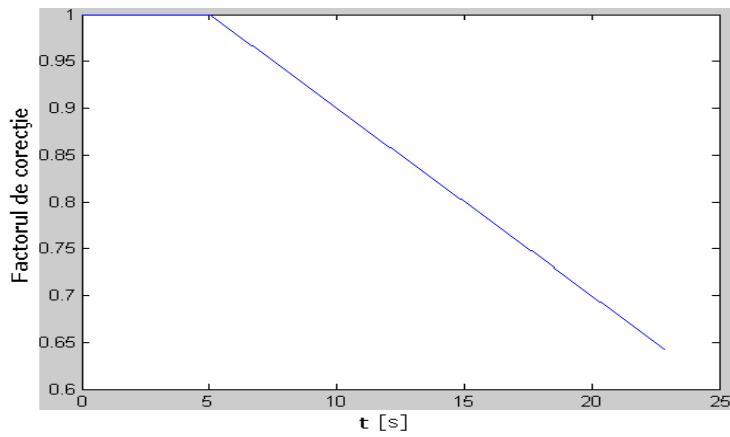
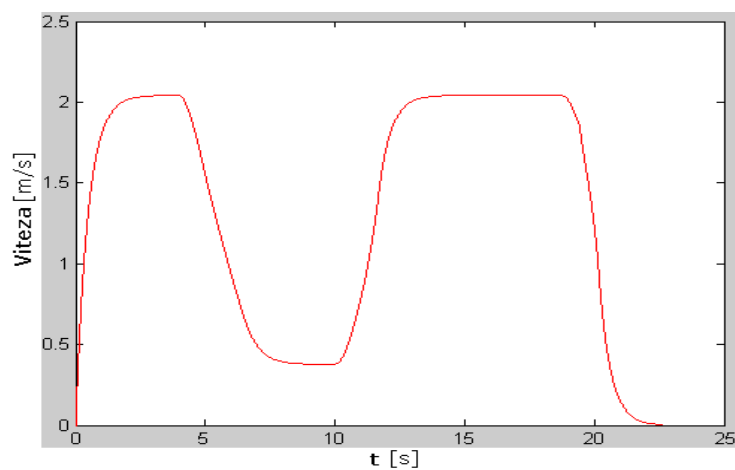
Fig. 4.3.26 Evoluția factorului de corecție  $\alpha$ 

Fig. 4.3.27 Evoluția vitezei robotului mobil în timp

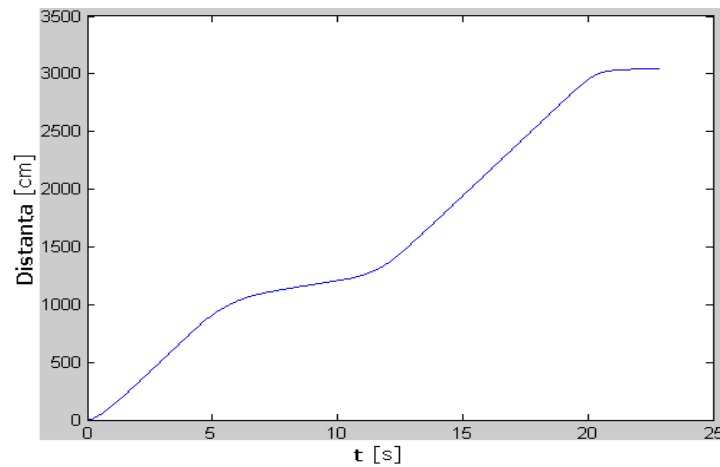


Fig. 4.3.28 Distanța parcursă de robotul mobil în timp

2) Al doilea experiment analizează situația în care *senzorul bază suferă un defect evolutiv în sensul măririi factorului de corecție  $\alpha$  până la o valoare superioară unde se va stabiliza*. Pe tot parcursul experimentului  $\alpha$  rămâne în limitele specificate pentru ca defectul de la senzorul bază să nu fie catalogat permanent.

Evoluția obstacolului mobil față de robot este aceeași ca în experimentul anterior, fiind ilustrată în figura 4.3.24. Fig. 4.3.29 ilustrează distanța față de obstacolul mobil furnizată de senzorul bază (se observă că după 5 secunde distanța furnizată începe să nu mai fie corectă, deoarece la nivelul senzorului bază apare defectul evolutiv). Evoluția factorului de corecție  $\alpha$  este ilustrată în fig. 4.3.30 de unde se distinge faptul că de la momentul  $t=5s$  factorul devine supraunitar și se stabilizează la  $t=20s$ .

Regulatorul interpolativ tolerant la defecte asigură o funcționare continuă și sigură a sistemului în pofida defectului evolutiv apărut la senzorul bază. Acest lucru rezultă comparând figurile 4.3.31 și 4.3.32, care reprezintă evoluția vitezei robotului mobil și a distanței parcurse de acesta în timp, cu figurile 4.3.7 și 4.3.8 din cadrul paragrafului 4.3.1.

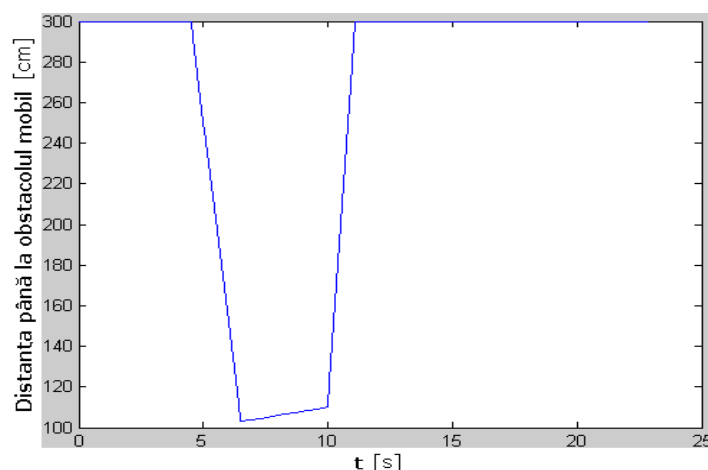


Fig. 4.3.29 Distanța față de obstacolul mobil furnizată de senzorul bază

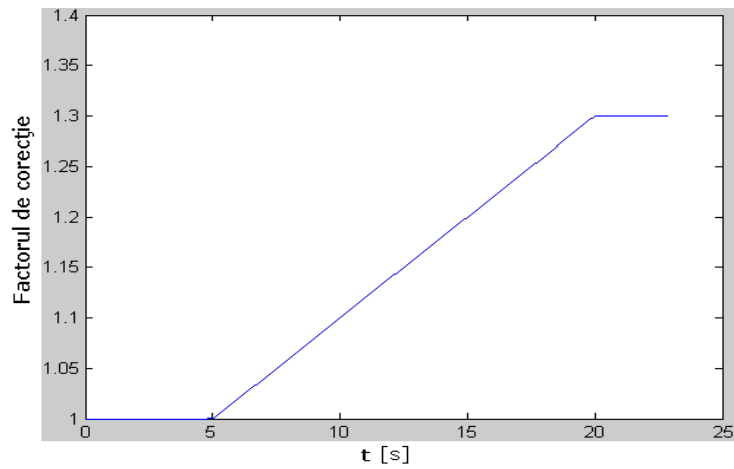
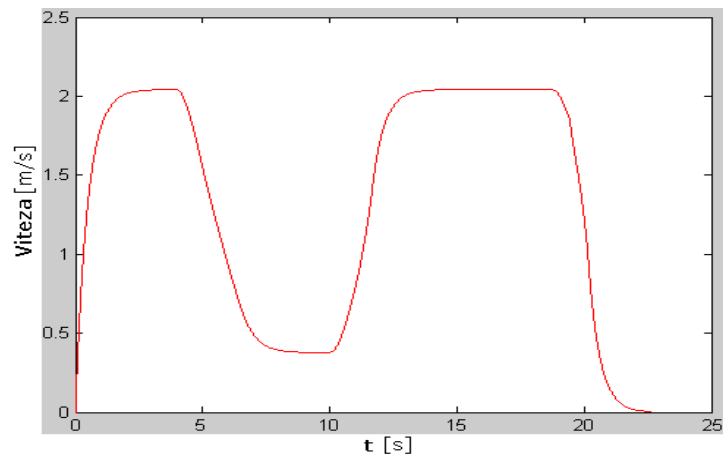
Fig. 4.3.30 Evoluția factorului de corecție  $\alpha$ 

Fig. 4.3.31 Evoluția vitezei robotului mobil în timp

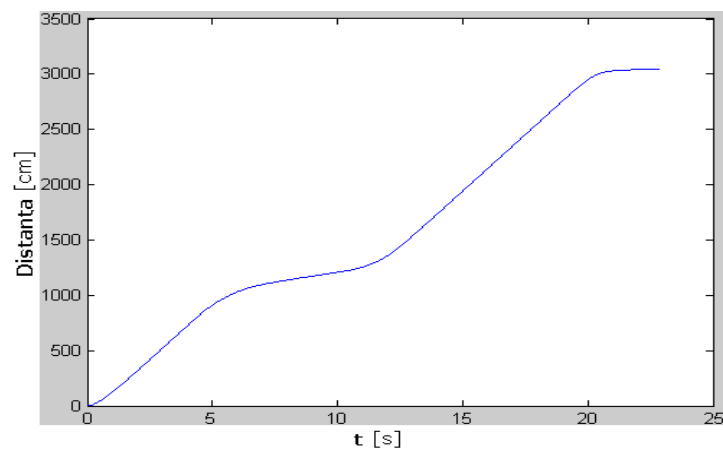


Fig. 4.3.32 Distanța parcursă de robotul mobil în timp

3) Scopul acestui experiment a fost de a ilustra situația în care *senzorul bază al sistemul de achiziție se defectează gradat (defect evolutiv) până când trece de limita admisă*. După cum s-a precizat, starea de defectare a senzorului bază se poate analiza cu ajutorul factorului de corecție  $\alpha$ . S-a fixat limita inferioară pentru  $\alpha$ ,  $\alpha_{lim1}=0.5$ , iar limita superioară neconducând la o situație atât de periclitantă s-a fixat la  $\alpha_{lim2}=1.8$ .

Scenariul stabilește că senzorul bază suferă un defect evolutiv, începând cu momentul  $t=3$  secunde, observabil pe evoluția factorului de corecție  $\alpha$  din fig. 4.3.36. Din aceeași figură se poate observa că la momentul  $t=16s$  factorul de corecție depășește limita inferioară admisă, însemnând că după 16s defectul evolutiv este catalogat ca defect permanent. În consecință, se intră în funcționarea în regimul safe-mode. În safe-mode, tot sistemul se bazează pe informațiile furnizate de către senzorul referință, iar factorul de corecție se fixează pe valoarea  $red=1.3$  pentru a reduce viteza robotului.

Primul element perturbator intră în plaja de observabilitate a robotului la momentul  $t=4,7$  secunde, având evoluția descrisă în cadrul scenariului 1, evoluție ilustrată și în figura 4.3.34 până la momentul  $t=12s$ , când iese din plaja de observabilitate. Evoluția obsacolului mobil furnizată de senzorul bază este ilustrată în figura 4.3.33 până la momentul  $t=12s$ . După cum s-a menționat mai sus, senzorul bază suferă un defect permanent din momentul  $t=16s$ . Până în acest moment se poate observa că sistemul a tolerat defectul evolutiv la nivelul senzorului bază comparând evoluțiile vitezei și distanței din figurile 4.3.37 și 4.3.38 cu evoluțiile din figurile 4.3.7 și 4.3.8 din paragraful 4.3.1.

Din momentul  $t=16s$ , robotul evoluează în regimul safe-mode. În acest context, regulatorul încetează să mai fie tolerant la defecte, iar viteza sistemului este redusă, după cum se poate observa în figura 4.3.37. Sistemul în safe-mode se bazează pe informațiile furnizate de senzorul referință. Un al doilea obstacol intră în plaja de observabilitate a robotului mobil în momentul  $t=16s$ , se apropie până la 2,7m și iese din plaja de observabilitate la momentul  $t=21s$ . Evoluția celui de-al doilea obstacol mobil este ilustrată în figura 4.3.34 (după  $t=16s$ ), iar evoluția acestuia furnizată de senzorul referință este redată în figura 4.3.33 (după  $t=16s$ ). După cum reiese din cele două figuri, senzorul referință redă corect distanța, dar din cauză că sistemul este în safe-mode, elementul perturbator este văzut de către regulator mai aproape cu 0,55m, după cum se poate observa în figura 4.3.35. Datorită posibilității de reconfigurare a structurii de măsurare defectul permanent este astfel tolerat, dar robotul parcurge distanța prescrisă într-un timp mai lung, după cum se poate observa în figura 4.3.38.

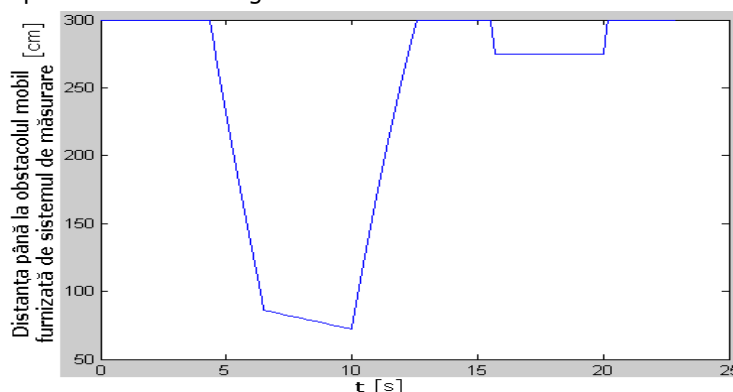


Fig. 4.3.33 Distanța față de obstacolul mobil furnizată de sistemul de achiziție

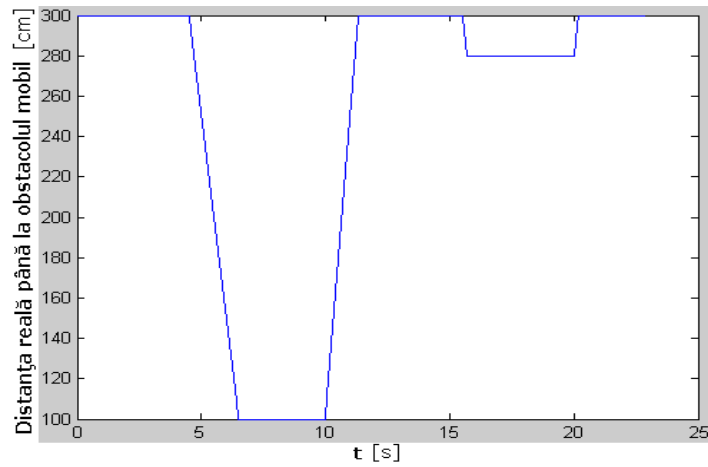


Fig. 4.3.34 Distanța reală față de elementul perturbator

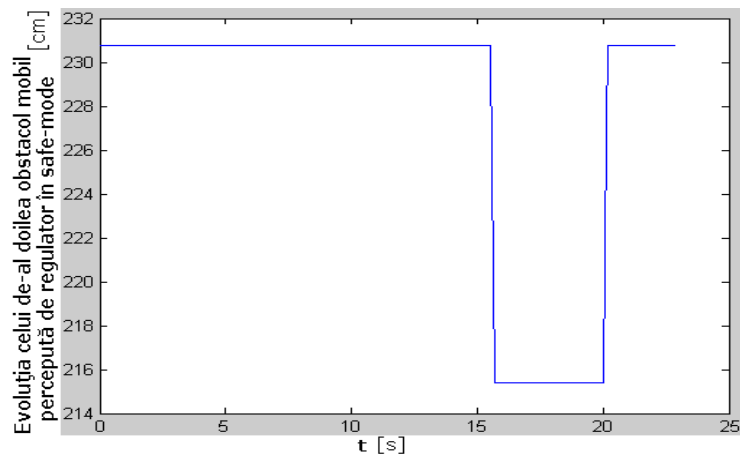


Fig. 4.3.35 Distanța față de al doilea obiect mobil procesată de regulator în safe-mode

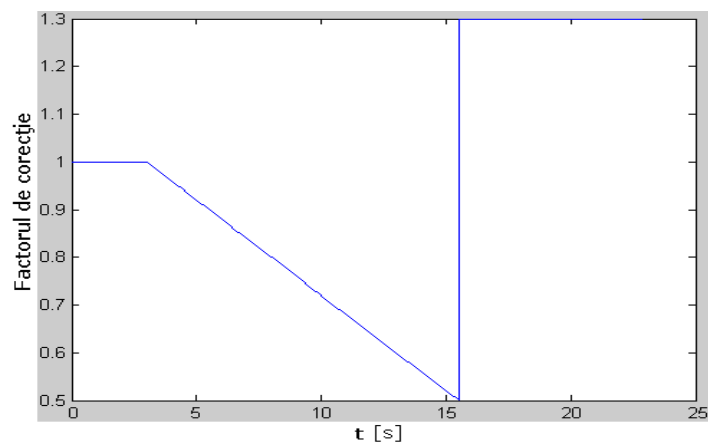


Fig. 4.3.36 Evoluția factorului de corecție



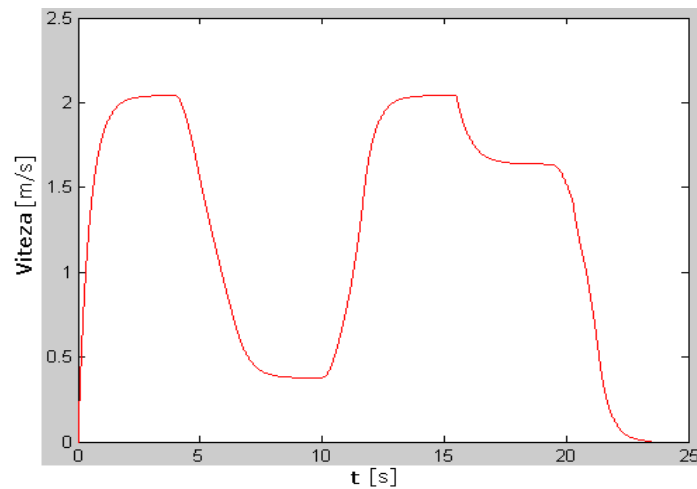


Fig. 4.3.37 Evoluția vitezei robotului mobil

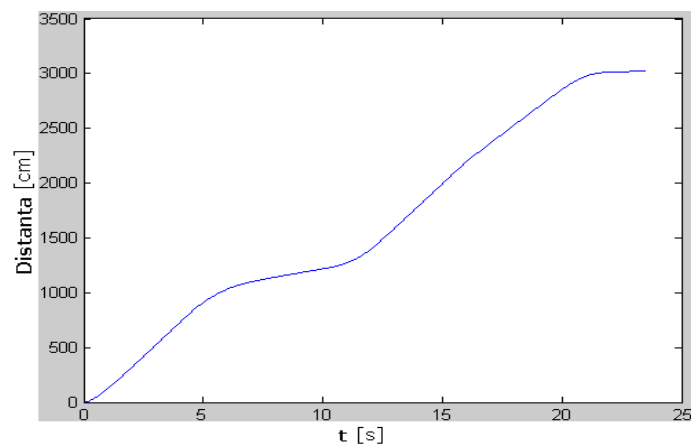


Fig. 4.3.38 Distanța parcursă de robotul mobil

4) Acest experiment analizează situația în care *la nivelul sensorului bază apare un defect permanent identificat în urma unui defect brusc*. Redundanța bazată pe referință identifică defectul permanent și astfel sistemul intră în safe-mode. La nivelul sensorului referință nu se identifică nici un defect pe parcursul experimentului, iar robotul își execută sarcina în safe-mode.

La nivelul sensorului bază apare un defect brusc identificat ca fiind defect permanent la momentul  $t=2s$ . Din acest moment sistemul intră în safe-mode. Se poate observa din figurile 4.3.40 (evoluția obstacolului mobil) și 4.3.41 (evoluția obstacolului mobil percepută de sistemul de reglare) că după intrarea în safe-mode (momentul  $t=2s$ ), obstacolul este perceput mai aproape decât în realitate.

Evoluția factorului de corecție este ilustrată în figura 4.3.39 (după 2 secunde factorul de corecție este setat la valoarea  $red=1.3$ ). Fig. 4.3.42 ilustrează distanța parcursă de robotul mobil în timp, iar fig. 4.3.43 prezintă evoluția vitezei sistemului. Se poate observa că în urma defectării sensorului bază, viteza sistemului se reduce,

iar viteza maximă (viteza de deplasare când nu există nici un obstacol mobil în plaja de observație) este micșorată la 1,65 m/s.

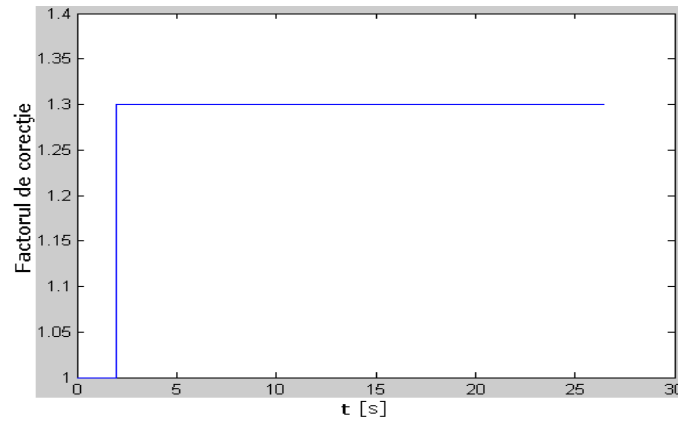


Fig. 4.3.39 Evoluția factorului de corecție

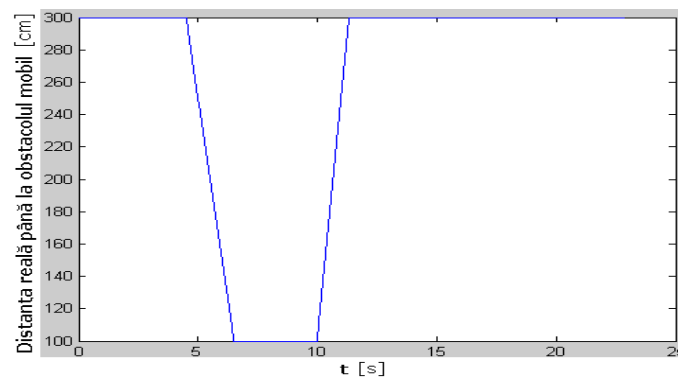


Fig. 4.3.40 Distanța reală și distanța furnizată de sistemul de achiziție față de obstacolul mobil

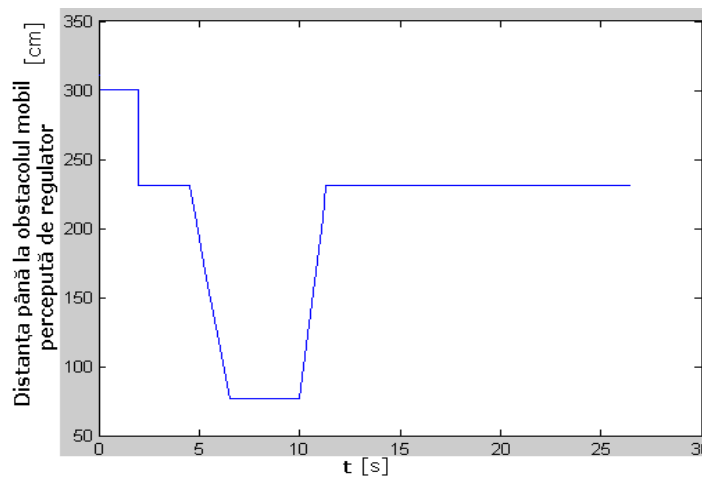


Fig. 4.3.41 Distanța față de obstacolul mobil considerată de sistemul de reglare

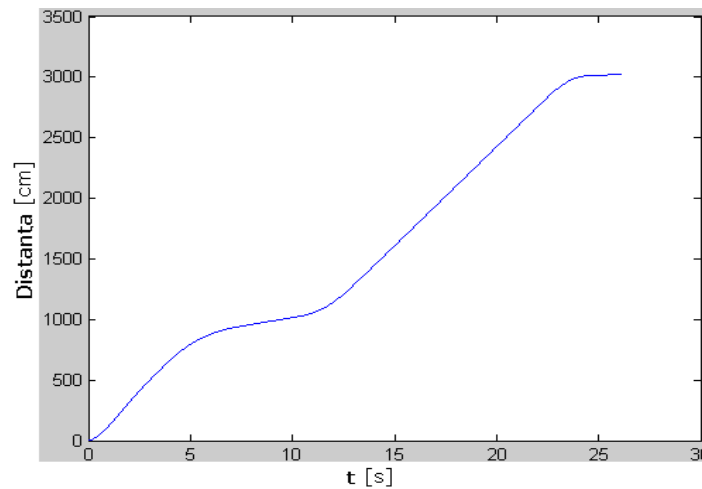


Fig. 4.3.42 Distanța parcursă de robotul mobil

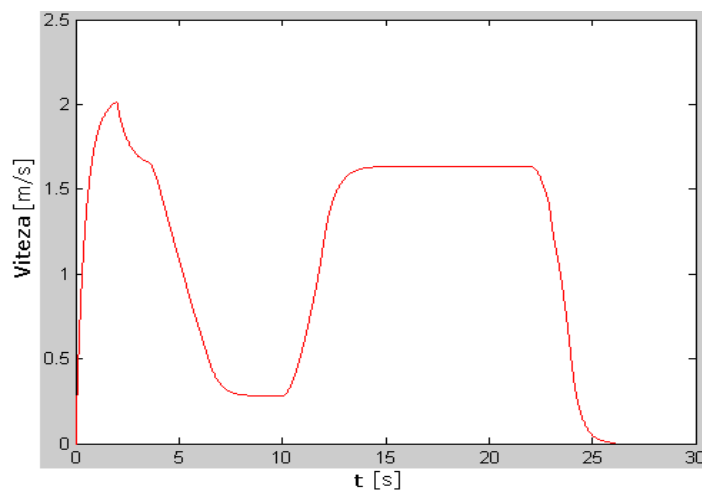


Fig. 4.3.43 Evoluția vitezei în funcție de timp

5) Ultimul experiment urmărește situația în care *cei doi senzori ai redundanței bazate pe referință se defectează permanent și sistemul este nevoit să se oprească*.

Figurile 4.3.44 și 4.3.45 ilustrează distanța reală față de elementul perturbator, respectiv distanța față de obiectul mobil în urma căreia sistemul de reglare stabilește viteza robotului mobil (distanța percepută de către sistemul de reglare. Defectul brusc la nivelul sensorului bază se produce la momentul  $t=2s$ . Din acest moment se intră în safe-mode.

Un al doilea eveniment se produce la momentul  $t=12,5$  secunde, când senzorul referință suferă un defect permanent iar sistemul este nevoit să se oprească. Se observă în fig. 4.3.45 (momentul  $t=12,5s$ ) că regulatorul determină oprirea sistemului considerând distanța față de obstacolul mobil fiind nulă.

Distanța parcursă de robotul mobil este de aproximativ 13 m (fig. 4.3.46), iar evoluția vitezei este ilustrată în fig. 4.3.47.

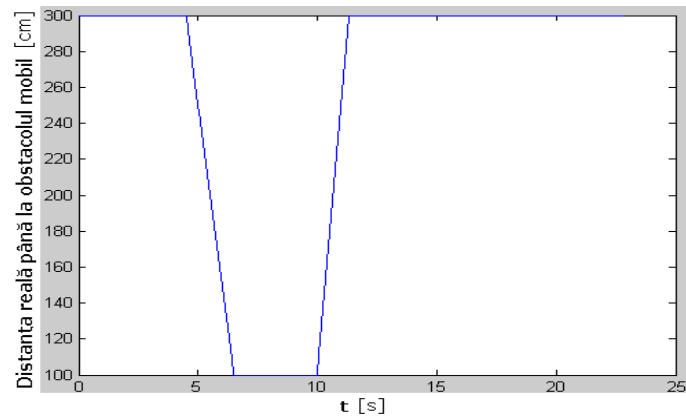


Fig. 4.3.44 Distanța reală față de obiectul mobil

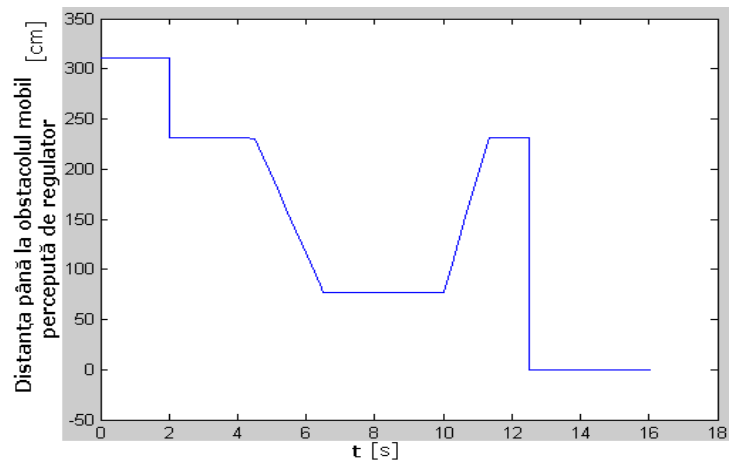


Fig. 4.3.45 Distanța față de obstacolul mobil considerată de sistemul de reglare

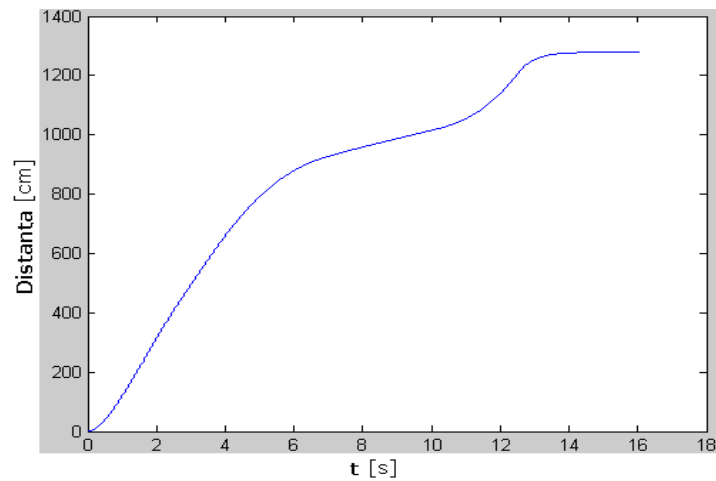


Fig. 4.3.46 Distanța parcursă de robotul mobil

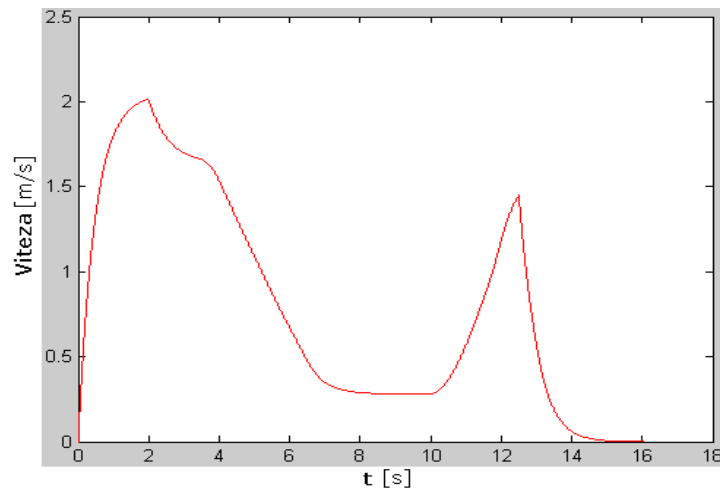


Fig. 4.3.47 Evoluția vitezei robotului mobil în timp

#### 4.3.5 Analiză de disponibilitate

Paragraful își propune să evidențieze printr-o analiză comparativă de disponibilitate, influența regulatorului tolerant la defecte realizat. Regulatorul tolerant la defecte a avut ca obiect central sensorul de distanță care furnizează distanța până la obstacolul mobil. Analiza disponibilității, realizată în cadrul acestui paragraf, se referă la două situații corespunzătoare unor date de fiabilitate diferite referitoare la sensorul de distanță.

În prima situație, se consideră că sensorul de distanță are un MTBF de 180 zile, respectiv rata de defectare  $\lambda_i=0.000231$ . Examinând acest MTBF, în urma analizării datelor și informațiilor primite de la diverși producători de senzori, autorul apreciază că defectelor evolutive, bruște și permanente, le corespund ratele de defectare  $\lambda_e=0.000124$ ,  $\lambda_b=0.000077$ , respectiv  $\lambda_p=0.000086$ .

A doua situație corespunde cazului când s-a considerat că sensorul de distanță are un MTBF de 150 zile, rezultând astfel rata de defectare  $\lambda_i=0.000277$ . Pe aceeași bază ca și situația 1, au rezultat ratele de defectare:  $\lambda_e=0.000123$  pentru defectele evolutive,  $\lambda_b=0.000079$  pentru defectele bruște, respectiv  $\lambda_p=0.000125$  pentru defectele permanente.

Referitor la repararea senzorilor, în urma afirmațiilor făcute de marii producători de senzori (Schneider, Massa, etc.), se precizează că senzorii trebuie înlocuiți aproape întotdeauna în urma defectării, reparația fiind imposibilă. Astfel, în urma defectării unui sensor de distanță, fie el ultrasonic sau infraroșu, se estimează un timp de cădere (MDT) de minimum 2 săptămâni, timp necesar achiziționării și transportului noului sensor, precum și înlocuirii sensorului defect. Ca urmare, rata de reparare considerată pentru calculul disponibilității este  $\mu=0.0029$ .

Pentru a face o analiză a disponibilității se va utiliza modelarea Markov (vezi subcapitolul 1.2). Stările sistemului sunt:

- starea  $\textcircled{0}$  - starea de succes a sistemului, neafectată de nici un defect la nivelul sensorului de distanță

- starea ① – starea în care senzorul de distanță suferă un defect evolutiv
  - starea ② – starea în care senzorul de distanță suferă un defect brusc
  - starea ③ – senzorul de distanță este afectat de un defect permanent
- Modelul construit este ilustrat în figura 4.3.48:

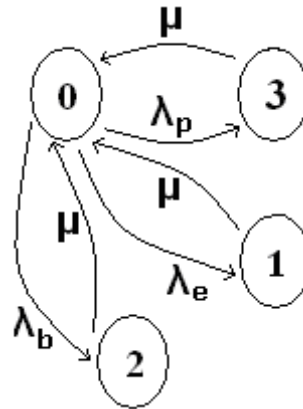


Fig. 4.3.48 Modelul stabilit pentru analiza disponibilității

Procesele tranzitorii asociate probabilităților celor patru stări ale modelului prezentat în fig. 4.3.48 pot fi descrise de ecuațiile:

$$\begin{aligned}
 P_0(t + \Delta t) &= P_0(t)(1 - \lambda_p \Delta t - \lambda_b \Delta t - \lambda_e \Delta t) + P_1(t)\mu \Delta t + P_2(t)\mu \Delta t + P_3(t)\mu \Delta t \\
 P_1(t + \Delta t) &= P_0(t)\lambda_e \Delta t + P_1(t)(1 - \mu \Delta t) \\
 P_2(t + \Delta t) &= P_0(t)\lambda_b \Delta t + P_2(t)(1 - \mu \Delta t) \\
 P_3(t + \Delta t) &= P_0(t)\lambda_p \Delta t + P_3(t)(1 - \mu \Delta t)
 \end{aligned}
 \tag{15}$$

Prin trecere la limită pentru  $\Delta t \rightarrow 0$  se obțin ecuațiile de stare cu ajutorul cărora se poate studia dinamica tranzițiilor și evoluția disponibilității:

$$\begin{cases}
 \dot{P}_0(t) = -(\lambda_e + \lambda_b + \lambda_p) \cdot P_0(t) + \mu \cdot (P_1(t) + P_2(t) + P_3(t)) \\
 \dot{P}_1(t) = \lambda_e \cdot P_0(t) - \mu \cdot P_1(t) \\
 \dot{P}_2(t) = \lambda_b \cdot P_0(t) - \mu \cdot P_2(t) \\
 \dot{P}_3(t) = \lambda_p \cdot P_0(t) - \mu \cdot P_3(t)
 \end{cases}
 \tag{16}$$

În figura 4.3.49, este ilustrată comparativ evoluția disponibilităților.  $A_{\text{initial}}$  reprezintă disponibilitatea sistemului inițial, iar  $A_{\text{final}}$ , disponibilitatea sistemului îmbunătățit.

Pentru calculul disponibilității  $A_{\text{initial}}$  s-a avut în vedere că starea de succes a sistemului este starea ① (deoarece defectele bruște și permanente nu sunt tolerate în cazul sistemului inițial), iar pentru calculul disponibilității  $A_{\text{final}}$  faptul că stările de succes ale sistemului sunt ①, ② și ③.

Schema Simulink corespunzătoare formulei (16) cu ajutorul căreia s-au realizat simulările este ilustrată în figura 4.3.50.

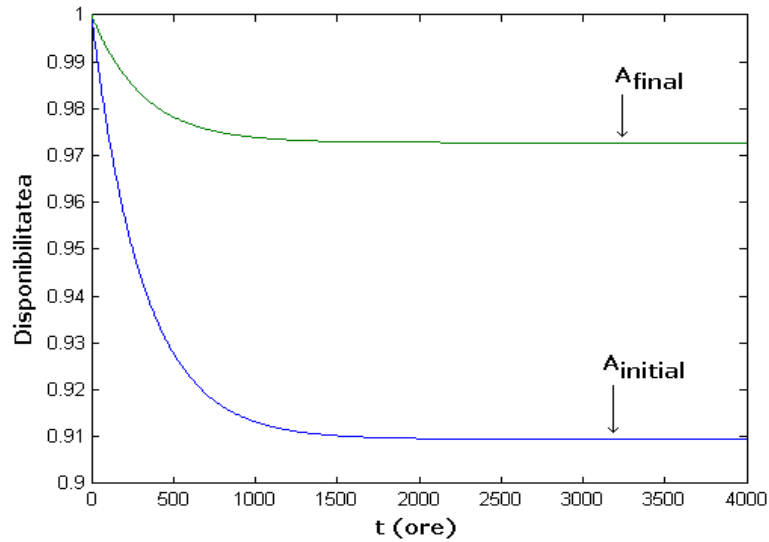


Fig. 4.3.49 Evoluția disponibilităților pentru analiza realizată în situația 1.

După cum se poate observa din figura 4.3.49, disponibilitatea se îmbunătățește în urma implementării regulatorului tolerant la defecte cu 6.5%.

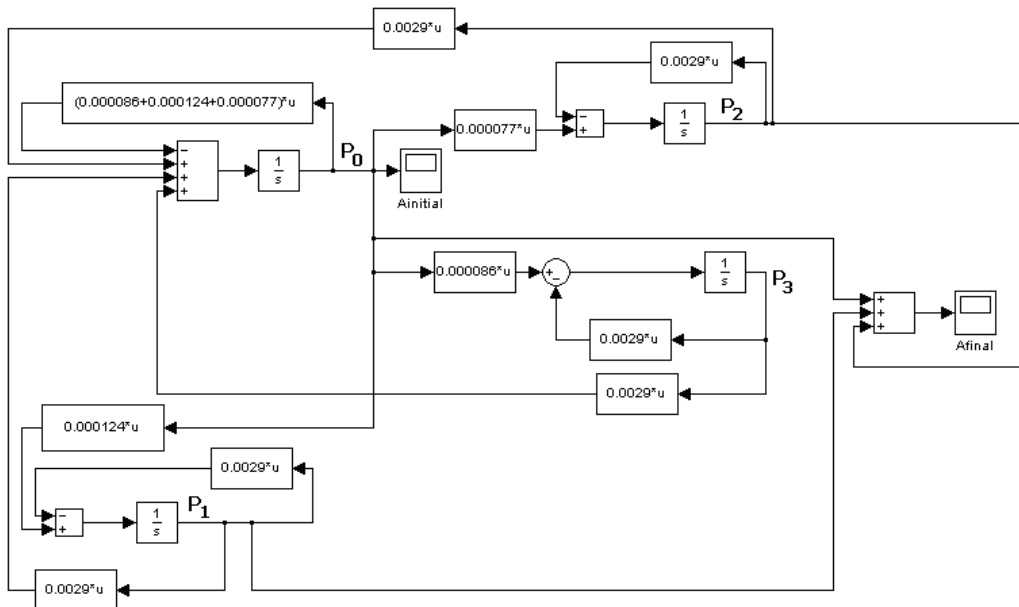


Fig. 4.3.50 Schema Simulink asociată analizei disponibilității

Pentru situația a doua, similar situației 1, în urma simulărilor de obțin evoluțiile disponibilităților  $A_{\text{initial}}$  și  $A_{\text{final}}$  care sunt ilustrate comparativ în figura 4.3.51.

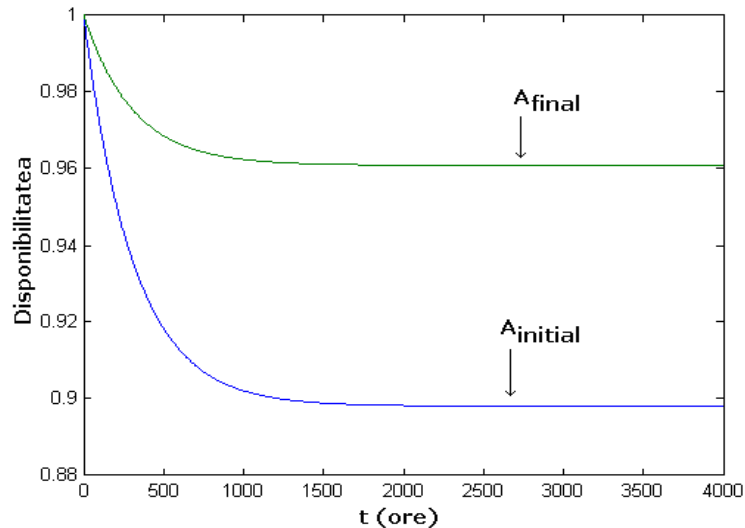


Fig. 4.3.51 Evoluția disponibilităților pentru analiza realizată în situația 2.

După cum se poate observa din figura 4.3.51, în urma implementării structurii tolerante la defecte, disponibilitatea se îmbunătățește cu 6-6.5%.

#### 4.4 Aspecte referitoare la creșterea disponibilității și siguranței în deplasarea roboților mobili cu roți

În cadrul aplicațiilor cu roboți mobili studiile recente se concentrează pe deplasarea acestora în medii semicunoscute (spații dinamice cunoscute) [88] sau în medii necunoscute (spații cu hazard) [40]. Paragrafele 4.2.1 și 4.2.2 prezintă tipurile de defecte care pot apărea la aceste tipuri de roboți. Evoluția roboților mobili trebuie să fie sigură, iar în acest sens studiul siguranței în funcționare și al fiabilității prezintă interes [12], [23], [75].

Funcționarea corectă a unui robot mobil înseamnă o funcționare fără erori și defecte. Apariția erorilor pe parcursul mișcării robotului conduce la deviații de diverse amplitudini (în funcție de gravitatea erorilor) în evoluției acestuia. Aceste deviații se pot solda cu incapacitatea sistemului de a-și îndeplini sarcinile, ceea ce se cataloghează drept defect. Astfel, creșterea siguranței în funcționare pentru roboții mobili înseamnă impunerea și obținerea unei funcționări corecte.

Subcapitolul își propune să prezinte o soluție de creștere a disponibilității pentru roboții mobili prin detecția și eliminarea erorilor de odometrie. Exemple de studii care urmăresc eliminarea erorilor de odometrie sunt prezentate în [103], [17]. Studiile prezintă diverse neajunsuri: majoritatea sunt bazate pe erorile sistematice, iar de cele mai multe ori nu se ține seamă de evoluția robotului mobil în spații dinamice. Aceste neajunsuri (probleme) sunt analizate mai în detaliu în paragraful 4.4.3. Eliminarea erorilor de odometrie se realizează prin corecția orientării robotului prin preluarea și procesarea de imagini. Localizarea pe baza informațiilor vizuale provenite din mediul înconjurător [44], [10] (procesarea de imagini reprezentând un domeniu de actualitate [95], [28]), precum și utilizarea informațiilor vizuale în



vederea corecției de poziție și orientare [128], reprezintă subiectele unor studii intense și de actualitate.

Subcapitolul se divide în cinci părți: întâi se face o scurtă descriere a robotului mobil, apoi se prezintă evoluția autonomă a robotului mobil cu roți într-un spațiu semicunoscut (spații cunoscute dinamice), se ilustrează problemele care pot să apară pe parcursul deplasării (posibilele surse ale erorilor de orientare), se dezvoltă modalitatea de creștere a disponibilității prin utilizarea informației redundante provenite de la anumite puncte din mediu (prin procesarea imaginii furnizate de camera de luat vederi a robotului) pentru a corecta erorile de orientare, iar în final se prezintă studiul practic.

#### 4.4.1 Robotul X80

Robotul mobil X80 are în componență: un controller de mișcare și interacțiune, un controller multimedia, un modul de imagine color cu cameră de luat vederi, un modul senzorial ultrasonic, un modul senzorial de mișcare umană pyroelectric, un modul senzorial de măsură a distanței infraroșu Sharp și un modul wireless WiFi 802.11. Robotul mobil X80 este ilustrat în figura 4.4.1:

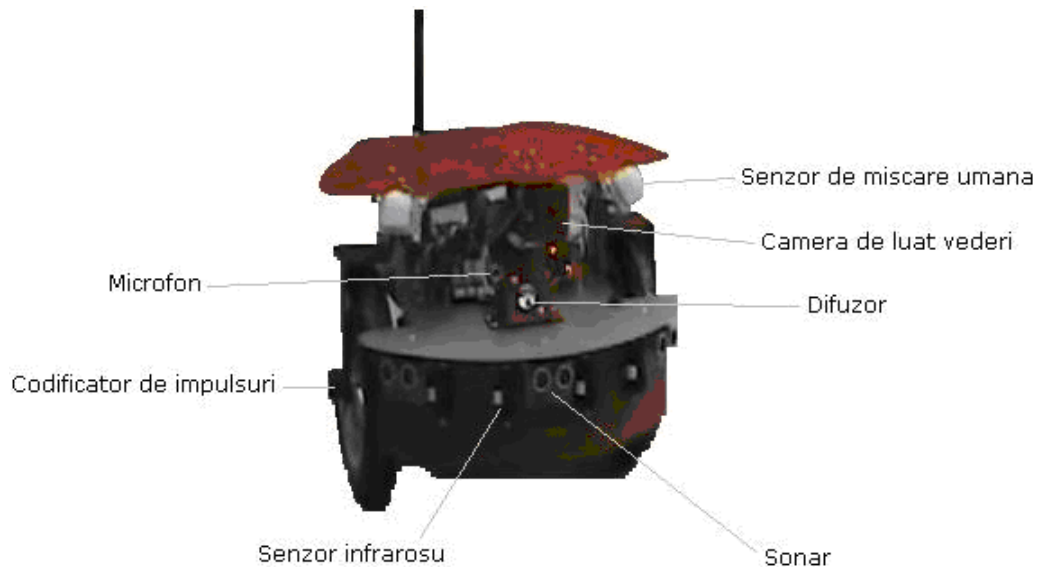


Fig. 4.4.1 Robotul mobil X80

Mediul de programare utilizat pentru implementarea aplicațiilor este Visual C++ 6.

Platforma cu robot mobil X80 [119] este proiectată pentru a dezvolta aplicații robotice precum: evoluție autonomă, monitorizare la distanță etc. Cu ajutorul modulului wireless WiFi 802.11, sistemul trimite informațiile de la senzori spre un PC cu o viteză de peste 10Hz. Robotul trimite de asemenea informații video și audio spre server, oferind astfel o gamă variată de posibilități de dezvoltare. Comenzile și instrucțiunile sunt transmise tot cu o viteză de peste 10Hz oferind astfel acces și control în timp real. Figura 4.4.2 ilustrează modul în care se realizează comunicația Wireless între robot și PC.

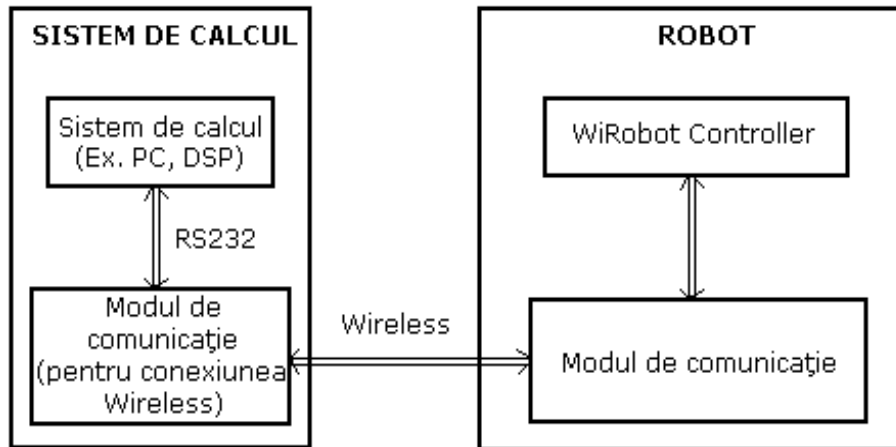


Fig. 4.4.2 Comunicația robot-calculator

Sistemul X80 oferă cercetătorilor posibilitatea: de dezvoltare a aplicațiilor de asistență robotică specializată, de uzitare ca robot cu sarcini de securitate sau pur și simplu de întrebuințare ca platformă de implementare a unor variate tipuri de proiecte (legate de interacțiunea om-robot, de navigația în sistemele mobile, de comportamentul roboților, de procesarea de imagini, de recunoașterea de obiecte, de recunoașterea vocii, de teleoperare, de preluarea informațiilor la distanță, de localizare, de realizare de hărți în deplasarea mobilă etc.).

Pentru dezvoltările din cadrul subcapitolului, resursele utilizate ale robotului sunt: senzorii de tip sonar (folosiți pentru determinarea distanțelor până la obstacole), senzorii de tip infraroșu (folosiți la procedura de ocolire a unui obstacol static), senzorii de mișcare umană (folosiți pentru detecția mișcării umane), modulul Wireless (folosit pentru comunicație între robot și sistemul de calcul), codificatoarele de impulsuri (folosite pentru localizare, determinarea poziției robotului mobil în spațiu), camera de luat vederi (pentru preluarea și procesarea de imagine în vederea corecției orientării, direcției de deplasare a robotului mobil).

#### 4.4.2 Deplasarea autonomă a robotului

Scopul primei părți a studiului întreprins de autor a fost obținerea unei deplasări autonome a robotului care evoluează în spații semicunoscute, caracterizată prin menținerea traiectoriei dorite în cazul apariției obstacolelor statice, respectiv dinamice, precum și în situația în care apar diferite defecte sau erori care modifică orientarea robotului.

Organigrama care ilustrează conceptul pe baza căruia se asigură deplasarea robotului este prezentată în fig. 4.4.3.

Robotul este echipat cu două motoare (unul la fiecare roată), iar la nivelul fiecărei roți există un codificator de impulsuri. Cu ajutorul acestor elemente împreună cu o aplicație software dezvoltată în Visual C++ și destinată pentru determinarea poziției robotului mobil (localizării) se realizează deplasarea robotului de la un punct inițial până la un punct final.

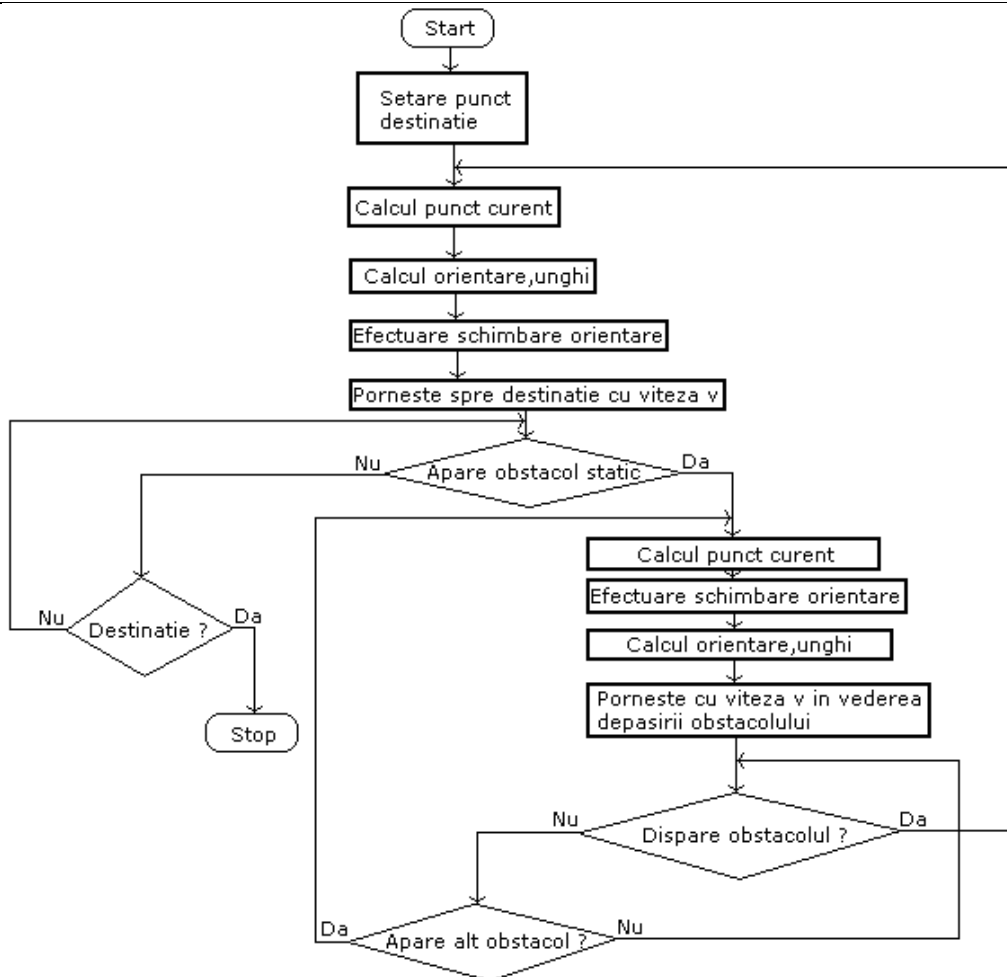


Fig. 4.4.3 Organigrama care ilustrează mișcarea robotului

Următoarele secvențe de cod ilustrează modul de lucru pentru a determina distanța care urmează să fie parcursă, precum și unghiul după care se face deplasarea.

Variabilele sunt: coordonatele punctului inițial ( $x_i$ ,  $y_i$ ), coordonatele punctului final ( $x_{f\_cm}$ ,  $y_{f\_cm}$ ), distanța obținută în centimetri ( $dist\_cm$ ), numărul de impulsuri pentru o rotație completă a roții (ROTAȚIE\_COMPLETĂ) și diametrul roții (DIAMETRU\_ROATĂ). Astfel, întâi se determină distanța care trebuie parcursă în centimetri,

$$dist\_cm = \sqrt{((x_{f\_cm} - x_i) * (x_{f\_cm} - x_i)) + ((y_{f\_cm} - y_i) * (y_{f\_cm} - y_i))};$$

urmând ca rezultatul să fie transformat în unitățile de măsură utilizate de codificatoarele de impulsuri prin formula:

$$distanța = (dist\_cm * ROTAȚIE_COMPLETĂ) / (\pi * DIAMETRU_ROATĂ);$$

În continuare se prezintă calculul unghiului față de punctul final și modificarea orientării în vederea deplasării spre destinație. Pentru situația în care coordonatele destinației se găsesc parcurgând în sens pozitiv ambele axe din punctul inițial, calculul unghiului se desfășoară potrivit secvenței de cod:

```

if (((xf_cm-xi)>=0)&&((yf_cm-yi)>=0))
{
  if ((yf_cm-yi)>(xf_cm-xi)) {
    unghi_nou=((double (pi/2))-(atan2((xf_cm-xi),(yf_cm-yi))));
  }
  else {
    unghi_nou=atan2((yf_cm-yi),(xf_cm-xi)); //fata de orizontala
  }
}

```

Unghiul calculat în grade se transformă în impulsuri (numărul de impulsuri necesare pentru ca orientarea robotului să se modifice după unghiul respectiv). Deplasarea robotului se declanșează cu ajutorul funcției (vezi [119]):

```

m_MOTSDK.DcMotorPositionTimeCtrAll
(cmd1,cmd2,NO_CONTROL,NO_CONTROL,NO_CONTROL,NO_CONTROL,timp);

```

unde: *cmd1* și *cmd2* reprezintă numărul de impulsuri corespunzător poziției finale a robotului (se setează cu ajutorul valorilor *distanța* și *orientare* determinate anterior), iar variabila *timp* reprezintă perioada de timp în care se va face deplasarea.

În permanență, cu ajutorul modulelor software se cunoaște orientarea robotului și poziția acestuia în spațiu (respectiv distanța până la punctul final).

Dacă robotul nu întâlnește nici un obstacol care interferează cu traiectoria lui, atunci se va deplasa spre punctul final și se va opri în dreptul acestuia.

Dacă robotul întâlnește pe parcursul evoluției sale:

- un obstacol dinamic – se oprește și așteaptă ca acesta să dispară din aria sa
- un obstacol static – îl va ocoli în modalitatea cea mai convenabilă în raport cu poziția acestuia.

Obstacolele se detectează cu ajutorul senzorilor de tip sonar, respectiv de tip infraroșu cu care este echipat robotul. În cazul întâlnirii unui obstacol static se trece la operația de ocolire (fig. 4.4.4). Acest procedeu implică modificarea orientării robotului pentru a putea ocoli obstacolul și deplasarea cu o viteză constantă în vederea depășirii acestuia.

După ce obstacolul a fost detectat cu ajutorul sonarelor, se va modifica orientarea robotului în vederea ocolirii utilizându-se senzorii infraroșu. Prima funcție apelată în momentul detectării unui obstacol este memorarea poziției curente. Se determină întâi numărul de impulsuri parcurse de la punctul anterior memorat (variabila *parcurs*). Pentru a obține coordonatele punctului curent se calculează distanța parcursă în centimetri

```

dianta_cm= double(((parcurs*DIAMETRU_ROATĂ)*pi)/ROTAȚIE_COMPLETĂ);

```

și unghiul în grade. Apoi, cu ajutorul unor funcții trigonometrice se determină coordonatele căutate.

Pentru depășirea obstacolului robotul evoluează cu viteză constantă până când elementul dispare (funcția este definită în [119]):

m\_MOTSDK.DcMotorVelocityNonTimeCtrAll (-200, 200, NO\_CONTROL, NO\_CONTROL, NO\_CONTROL, NO\_CONTROL);

În momentul în care obstacolul dispăre, robotul își va determina poziția și orientarea curentă (în prealabil robotul memorează toate schimbările de orientare, respectiv distanțele parcurse după un anumit unghi pentru a putea stabili gradul de deviere de la traiectoria inițială) pentru a putea reîncepe deplasarea spre punctul final.

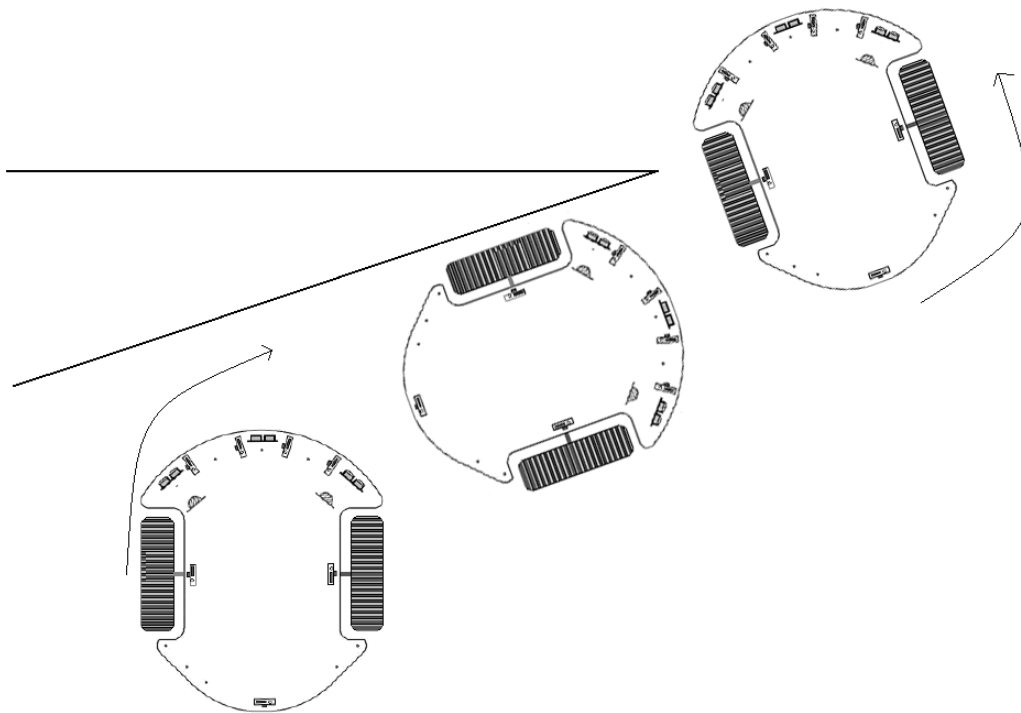


Fig. 4.4.4 Ocolirea unui obstacol static

Ideile prezentate în acest paragraf descriu algoritmul deplasării autonome a robotului mobil. Pentru ca deplasarea să se realizeze cu succes, robotul trebuie să-și cunoască în permanență poziția curentă și orientarea. Întotdeauna însă, apar defecte și erori care vor conduce la abateri de la traiectorie. O astfel de defecțiune apare în urma erorilor de odometrie. La o deplasare pe distanțe scurte, erorile pot fi minore, dar la distanțe mai mari erorile se amplifică rezultând o funcționare eronată, adică un defect.

#### 4.4.3 Erori de odometrie

Pe parcursul evoluției robotului mobil de la punctul inițial către punctul final pot să apară diverse erori sau defecțiuni care pot conduce la imposibilitatea îndeplinirii sarcinii impuse. În acest sens pot să apară erori de odometrie. Erorile de odometrie pot fi de două feluri: erori sistematice și erori nesistematice. Ambele au efect asupra orientării robotului și implicit asupra traiectoriei acestuia.

Erorile sistematice cele mai probabile sunt consecința următoarelor cauze:

- roțile robotului au diametre diferite,
- valoarea medie a diametrelor celor două roți diferă de valoarea nominală,
- alinierea greșită a roților,
- incertitudini la nivelul bazei efective a roților,
- rezoluție limitată superior a codificatorului de impulsuri,
- rată de eșantionare limitată superior a codificatorului de impulsuri

Erorile nesistematice au de regulă următoarele surse:

- deplasarea pe un teren nenivelat,
- deplasarea peste diverse obiecte neașteptate pe podea
- alunecarea roților datorită:
  - podelei alunecoase,
  - accelerării prea mari
  - ajustării rapide (ex. curbe)
  - acțiunii unor forțe externe (interacțiunea cu corpuri externe)
  - acțiunii unor forțe interne (ex. roata de echilibru)
  - contactului nului cu podeaua

Atât erorile sistematice cât și cele nesistematice pot avea un efect grav asupra funcționării robotului mobil.

Cele mai multe studii tratează doar partea de erori de odometrie sistematice, bazându-se pe o calibrare apriorică [17], [103]. Există foarte puține studii care să surprindă și aspectul erorilor nesistematice, un exemplu fiind [26] unde nu se ține seama de faptul că robotul evoluează în spații dinamice.

#### **4.4.4 Procedeu de corecție a direcției de deplasare a robotului prin preluarea și procesarea de imagine**

Cercetarea din acest paragraf se axează pe corectarea erorilor de orientare în deplasarea robotului prin introducerea unor informații redundante provenite de la obiecte din mediul inconjurător (în cazul implementat în lucrare informația redundantă este furnizată din punctul țintă). Informația suplimentară se transmite prin imagine, iar prin modulul de procesare a imaginii se obține factorul de corecție.

Principiul de procesare al imaginii se bazează pe:

- i) descompunerea imaginii sursă (reprezentând obiectul care urmează să fie identificat) în pixeli care se grupează în cadrul unor familii de culori (care reprezintă grupuri de pixeli înrudiți – aflați în interiorul unui prag admisibil al valorilor roșu, albastru și verde, unul față de celălalt),
- ii) eliminarea tuturor pixelilor din cadrul imaginii destinație (imaginea în care se caută obiectul din imaginea sursă) care nu aparțin familiilor de culori identificate în imaginea sursă,
- iii) eliminarea obiectelor suplimentare din cadrul imaginii destinație în vederea identificării obiectului căutat. Criteriul de eliminare se bazează pe numărul de familii de culori regăsit în fiecare obiect, precum și pe numărul de pixeli conținut de fiecare familie de culori.

Amplasarea punctelor de reper care să furnizeze informația redundantă pentru corecția orientării trebuie amplasate la o distanță optimă unul față de celălalt din două motive:

- dacă distanța dintre ele este prea mică atunci se mărește numărul de procesări de imagine și de corecții este prea mare
- dacă distanța este prea mare atunci se poate ajunge la situația în care eroarea de orientare se amplifică și implicit apar și erori mari de distanță. Acest fapt conduce la pierderea din cadru a țintei. Implicit, situația apărută creează pierderi de timp și întreruperea mișcării continue pentru a apela modulul de căutare a țintei. Se poate ajunge și la pierderea totală a țintei, fapt ce implică o pierdere totală a coordonatelor curente de către robot.

Robotul este dotat cu o cameră de luat vederi poziționată frontal (vezi fig. 4.4.1). Camera de luat vederi realizează captura și salvarea de imagini în format .bmp cu ajutorul funcțiilor furnizate în [119].

```
m_MOTSDK.TakePhoto ();
m_MOTSDK.SavePhotoAsBMP ("C:\\delarobot.bmp");
```

Scopul utilizării camerei de luat vederi este acela de a identifica obiectul căutat (OB1) în cadrul imaginii destinație preluate (IM2).

#### 4.4.4.1 Identificarea caracteristicilor obiectului căutat

Imaginea sursă (IM1) (fig. 4.4.5) conține OB1 care descompus și pixelii acestuia sunt memorați într-o matrice bidimensională (MAT1). Fiecare element al MAT1 furnizează trei valori corespunzătoare fiecărui pixel: roșu, verde și albastru (RGB). Dimensiunea imaginii căutate ( $m1 \times n1$ ) corespunde dimensiunii obiectului care ar urma să fie identificat de robot dacă acesta s-ar afla în poziția corectă. Pasul următor îl reprezintă descompunerea și gruparea pixelilor obiectului OB1 în familii de culori. Adaugarea unui pixel în cadrul unei familii de culori se realizează după secvența următoare de cod:

```
if ((gasit==0) && (((GetBValue(matrice[i][j]))<=(fam[k].supb+plaja) &&
((GetBValue(matrice[i][j]))>=(fam[k].infb-plaja))) &&
(((GetRValue(matrice[i][j]))<=(fam[k].supr+plaja) &&
((GetRValue(matrice[i][j]))>=(fam[k].infr-plaja))) &&
(((GetGValue(matrice[i][j]))<=(fam[k].supg+plaja) &&
((GetGValue(matrice[i][j]))>=(fam[k].infg-plaja))))))
{ fam[k].nr_pixelii++;
if ((GetBValue(matrice[i][j]))<fam[k].infb)
    fam[k].infb= (GetBValue(matrice[i][j]));
if ((GetBValue(matrice[i][j]))>fam[k].supb)
    fam[k].supb= (GetBValue(matrice[i][j]));
if ((GetRValue(matrice[i][j]))<fam[k].infr)
    fam[k].infr= (GetRValue(matrice[i][j]));
if ((GetRValue(matrice[i][j]))>fam[k].supr)
    fam[k].supr= (GetRValue(matrice[i][j]));
if ((GetGValue(matrice[i][j]))<fam[k].infg)
    fam[k].infg= (GetGValue(matrice[i][j]));
if ((GetGValue(matrice[i][j]))>fam[k].supg)
    fam[k].supg= (GetGValue(matrice[i][j]));
gasit=1;
}
```

Notațiile folosite au următoarea semnificație:

- fam[k] – familia de culori,
- nr\_pixeli – numărul de pixeli din familia de culori,
- plaja – plaja setată admisibilă pentru a introduce un pixel în cadrul familiei de culori,
- matrice[i][j] – pixelul curent,
- supb, infb, subr, infr, supg, infg – limitele superioare și inferioare ale familiei de culori,
- GetBValue, GetRValue, GetGValue – funcții care extrag valoarea albastru, roșu și verde din pixelul respectiv.

În final se memorează numărul de familii de culori, respectiv numărul de pixeli conținut de fiecare familie în parte (de exemplu, pentru obiectul din figura 4.4.5 se vor identifica patru familii de culori).



Fig. 4.4.5 Obiectul căutat

#### 4.4.4.2 Identificarea obiectului căutat în cadrul imaginii preluate de cameră

După preluarea imaginii IM2 urmează să se facă identificarea OB1 în cadrul acesteia. Pixelii din IM2 se memorează în matricea MAT2. Procesarea imaginii își propune să elimine pas cu pas surplusul din imaginea IM2 pentru ca în final IM2 să conțină doar obiectul căutat.

Acest lucru se realizează pe parcursul a doi pași:

- se elimină din IM2 pixelii care nu fac parte din familia de culori a OB1
- se identifică și se compară obiectele rămase în imagine eliminându-se unul câte unul, până când în imagine rămâne doar OB1.

Identificarea obiectelor rămase în IM2 după ce s-au eliminat pixelii care nu făceau parte din familia de culori a OB1 se realizează grupând pixelii învecinați rămași. În continuare se vor elimina obiectele unul câte unul pentru a obține OB1. În fiecare pas se compară două obiecte din cadrul imaginii IM2, analizând numărul de familii de culori (incluse un familia de culori a OB1). Numărul de familii de culori este primul criteriu de departajare a celor două obiecte. Dacă pentru cele două obiecte numărul de familii de culori este egal, atunci se trece la al doilea criteriu de departajare care se bazează pe numărul de pixeli conținut de fiecare familie de culori.

În figura 4.4.6 se ilustrează un exemplu pentru IM2 în care apare OB1. Figura 4.4.7 prezintă cele două obiecte rămase după ce se elimină pixelii care nu fac



parte din familia de culori a OB1, urmând ca în fig. 4.4.8 să rămână doar OB1 (după ce s-a analizat numărul de familii de culori).

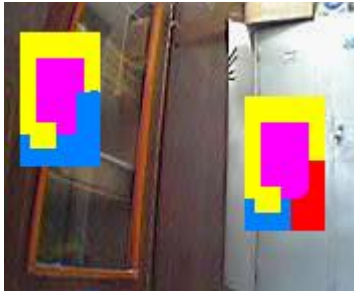


Fig. 4.4.6 Imaginea inițială (IM2)



Fig. 4.4.7 Imaginea după prima prelucrare



Fig. 4.4.8 Obiectul identificat în imagine

#### 4.4.4.3 Determinarea poziției obiectului căutat în imagine și furnizarea informației necesare pentru corecție

După identificarea obiectului căutat în cadrul imaginii se trece la calculul poziției obiectului în imagine (în funcție de coordonatele obiectului în imagine). Modul de procesare a imaginii are ca ieșire deviația ( $\Delta x$ ) față de poziția corectă așteptată a robotului față de obiectul căutat. Deviația ( $\Delta x$ ) se transformă într-un unghi, respectiv un număr de impulsuri de corecție a orientării robotului (fig. 4.4.9).

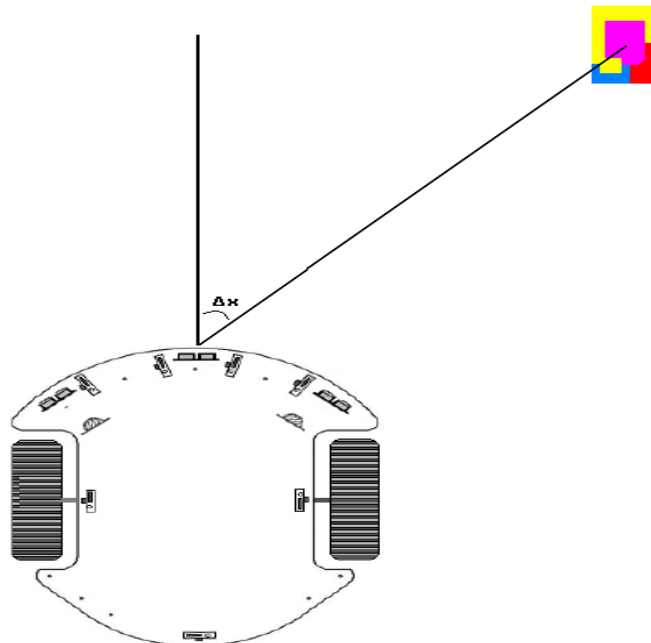


Fig. 4.4.9 Orientarea se modifică în funcție de orientarea curentă și de orientarea nouă identificată

Transmiterea datelor între programul principal și modulul de procesare a imaginii se realizează prin intermediul fișierelor text (variabila `fault_orientare` este ieșirea modulului de procesare a imaginii și este scrisă în fișierul text):

```
f=fopen("C:\\orientare.txt","wt");
fprintf(f,"%d",fault_orientare);
fclose(f);
```

Având în vedere cele menționate în paragraful 4.4.2 deplasarea robotului de la un punct la altul se va face utilizând cele două codificatoare de impulsuri de la nivelul celor două roți iar pentru a identifica obstacolele dinamice, respectiv pentru a ocoli obstacolele statice se vor utiliza cele trei sonare și senzorii infraroșu. Pe parcursul deplasării robotului, pot apărea erori de orientare și de distanță cauzate de diverși factori (ex. erori de odometrie). Dacă nu există o informație care să corecteze aceste erori, robotul va avea o evoluție eronată catalogată ca defect.

Una dintre problemele care apar este aceea că intervalul de timp necesar pentru procesarea imaginii este mai mare decât pasul de eșantionare al sistemului. Acest fapt conduce la eliminarea posibilității de a realiza reglarea orientării procesând imaginea la fiecare pas de eșantionare, având ca referință poziția dorită a obiectului identificat în imagine. Cele menționate conduc la următoarea ipoteză de lucru: verificarea redundantă a orientării robotului prin procesarea de imagine trebuie realizată suficient de des pentru a se evita erori mari de orientare care rezultă în erori considerabile de distanță.

#### 4.4.5 Studiu practic

În cele ce urmează, se descriu două experimente care să ilustreze eficiența celor prezentate mai sus. Robotul mobil trebuie să evolueze într-un spațiu semicunoscut și să parcurgă distanța de la un punct inițial până la un punct final.

La pornire, robotul trebuie să-și calculeze distanța pe care o are de parcurs, unghiul după care se va începe deplasarea și să-și schimbe orientarea pentru a evolua pe distanța cea mai scurtă spre destinație.

Pe parcursul deplasării, robotul mobil întâlnește un obstacol static pe care trebuie să-l ocolească. Procedeu trebuie să aibă trei caracteristici: modificarea orientării robotului astfel încât deplasarea să se facă în paralel cu obstacolul, calculul poziției și orientării curente în momentul în care obstacolul dispăre, respectiv modificarea orientării și deplasarea spre punctul destinație.

Dacă pe parcursul deplasării apar erori de orientare atunci acestea se corectează cu ajutorul modulului de preluare și procesare a imaginii.

În primul grup de figuri se prezintă primul experiment în care:

- robotul trebuie să meargă de la punctul  $(x_0, y_0)$  la punctul  $(x_f, y_f)$
- pe parcursul evoluției întâlnește un obstacol static pe care îl ocolește
- robotul ajunge la țintă

Figurile descriu: momentul pornirii robotului (fig. 4.4.10 - poziția inițială, fig. 4.4.11 - după modificarea orientării în vederea obținerii traiectoriei dorite), momentul întâlnirii și ocolirii obstacolului static (fig. 4.4.12 - observarea obstacolului, fig. 4.4.13 - reorientarea pentru ocolire, fig. 4.4.14 - detectarea dispariției obstacolului, fig. 4.4.15 - reorientarea pentru a evolua pe traiectoria optimă spre punctul final, fig. 4.4.16 - evoluția spre țintă), drumul spre destinație (fig. 4.4.17 - evoluția spre țintă dintr-un alt unghi, fig. 4.4.18 - atingerea punctului final).

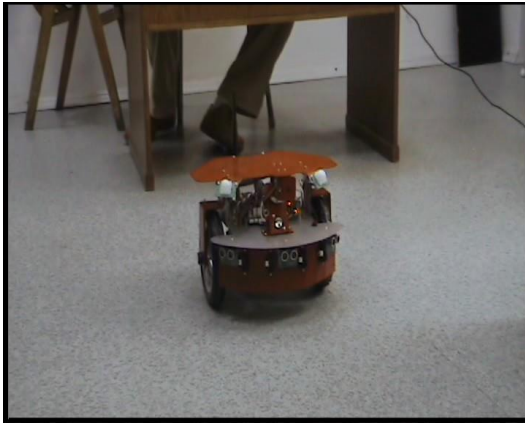


Fig. 4.4.10 Robotul în poziția inițială

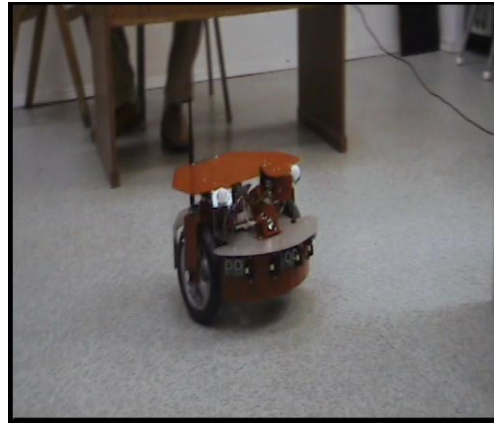


Fig. 4.4.11 După modificarea orientării



Fig. 4.4.12 Observarea obstacolului



Fig. 4.4.13 Reorientarea în vederea ocolirii obstacolului



Fig. 4.4.14 Detectarea dispariției obstacolului



Fig. 4.4.15 Reorientarea pentru a evolua pe traiectoria optimă spre punctul final



Fig. 4.4.16 Evoluția spre punctul țintă



Fig. 4.4.17 Evoluția spre țintă



Fig. 4.4.18 Atingerea punctului final

Al doilea grup de figuri descrie al doilea experiment care până la un punct este similar cu primul (până în momentul în care se face preluarea informației de la țintă prin intermediul imaginii furnizate de cameră).

- la momentul  $t_x$  se face preluarea și procesarea de imagine de la țintă, iar în urma acestui procedeu se corectează orientarea robotului mobil. Punctul final (ținta) este un obiect ușor identificabil în peisaj datorită configurației speciale de culori (în scenariul curent ținta are trei culori: roșu, galben, verde).

Astfel evoluția robotului între figurile 4.4.19 și 4.4.24 este similară primului experiment. În situația curentă ținta a fost deplasată pentru a sublinia modalitatea de corecție a orientării cu ajutorul informației preluate de la țintă prin intermediul camerei de luat vederi. S-a recurs la această procedură pentru că a părut eficientă în condițiile în care pe spații foarte mici erorile și mai ales corecția nu se pot observa în mod evident. Figurile descriu preluarea și procesarea informației redundante furnizate de țintă (fig. 4.4.25), corecția orientării în vederea obținerii traiectoriei

corecte (fig. 4.4.26), drumul spre destinație și atingerea punctului final (fig. 4.4.27 și 4.4.29).



Fig. 4.4.19 Robotul în poziția inițială



Fig. 4.4.20 După modificarea orientării



Fig. 4.4.21 Observarea obstacolului



Fig. 4.4.22 Reorientarea în vederea ocolirii obstacolului



Fig. 4.4.23 Detectarea dispariției obstacolului



Fig. 4.4.24 Reorientarea pentru a evolua pe traiectoria optimă spre punctul final



Fig. 4.4.25 Momentul preluării și procesării informației furnizate de țintă



Fig. 4.4.26 Corecția orientării în vederea obținerii traiectoriei corecte



Fig. 4.4.27 Drumul spre destinație



Fig. 4.4.28 Atingerea punctului final

În concluzie, semnalul furnizat de către țintă sub forma unei imagini reprezintă informația redundantă cu ajutorul căreia se corectează erorile de odometrie care apar pe parcursul deplasării.

#### 4.4.6 Analiză de fiabilitate și disponibilitate

Acest paragraf realizează un studiu de fiabilitate și disponibilitate cu privire la influența corecției erorilor de localizare pe parcursul mișcării unui robot mobil cu roți. Modul de corecție a orientării prin preluare și procesare de imagine realizează implicit corecția localizării robotului mobil și astfel fiabilitatea și disponibilitatea sistemului crește. Eficiența modului de corecție a orientării prin preluare și procesare de imagine este demonstrată prin simulări care vor viza urmărirea comparativă a evoluției fiabilității și disponibilității.

În cadrul paragrafului 4.2.2 s-au prezentat diverse date referitoare la fiabilitatea și disponibilitatea roboților mobili care evoluează în spații semicunoscute. Obținerea datelor specifice pentru diverse tipuri de roboți se poate realiza doar experimental în urma monitorizării de durată semnificativă. Literatura de specialitate prezintă un număr redus de astfel de studii, dintre care utilizabile pentru cercetarea din cadrul acestui paragraf au fost doar [5], [21], [113].

După cum s-a menționat în paragraful 4.2.2, indicatorul MTBF pentru roboții mobili cu roți care evoluează în spații semicunoscute (de tipul robotului X80, element de studiu în cadrul acestui subcapitol) este: după cercetarea efectuată în [113] de 7 ore, iar după [21] de 19.5 ore. Din studiul efectuat în [5] se poate determina un MTBF de 17.5 ore (robotul este în stare de funcționare o perioadă de 140 de ore și în această perioadă de timp au apărut 8 defecte).

Pentru realizarea studiului a fost necesară aflarea procentului de defecte cauzate de o localizare eronată. Din [113] și [5] se poate extrage următoarea concluzie: aproximativ 25% din totalul defectelor sunt defecte de localizare care pot fi corectate cu un modul de corecție de genul celui prezentat în cadrul acestui subcapitol.

Informațiile cumulate până în acest punct permit efectuarea unei analize a fiabilității. Dar, robotul mobil este un sistem reparabil iar un studiu complet trebuie să prezinte și o analiză a disponibilității. În acest scop interesează valoarea indicatorului MDT. În lucrarea [21] se furnizează valoarea MDT=60.5 ore. [21], [22] analizează doar disponibilitatea inerentă (vezi subcapitolul 1.2) cu toate că disponibilitatea operațională este cea care prezintă relevanță. Din calculul disponibilității inerente se extrage MTTR=1.3 ore.

Studiul fiabilității și disponibilității se bazează pe modelele Markov (vezi subcapitolul 1.2), reținându-se pentru o analiză minimală doar două stări ale sistemului: starea **0** stare de succes și starea **1** stare de defectare. Se consideră astfel modelul ilustrat în figura 4.4.29:

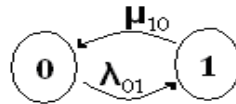


Fig. 4.4.29 Modelul stabilit pentru analiza fiabilității și disponibilității

Procesele tranzitorii asociate probabilităților celor două stări prezentate în fig. 4.4.29 pot fi descrise de ecuațiile următoare care țin cont de intensitățile de defectare și reparare corespunzătoare:

$$\begin{aligned} P_0(t + \Delta t) &= P_0(t)(1 - \lambda_{01}\Delta t) + P_1(t)\mu_{10}\Delta t \\ P_1(t + \Delta t) &= P_0(t)\lambda_{01}\Delta t + P_1(t)(1 - \mu_{10}\Delta t) \end{aligned} \quad (17)$$

Prin trecere la limită pentru  $\Delta t \rightarrow 0$  se obțin ecuațiile cu ajutorul cărora se poate studia dinamica tranzițiilor și evoluția fiabilității și disponibilității:

$$\begin{cases} \dot{P}_0(t) = -\lambda_{01}P_0(t) + \mu_{10}P_1(t) \\ \dot{P}_1(t) = \lambda_{01}P_0(t) - \mu_{10}P_1(t) \end{cases} \quad (18)$$

Se realizează un număr de cinci experimente care constau în: două analize de fiabilitate, două analize de disponibilitate și o analiză de disponibilitate inerentă.

Pentru efectuarea experimentelor, s-au implementat tranzițiile de forma (18), (19) cu ajutorul schemei Simulink ilustrată în figura 4.4.30.

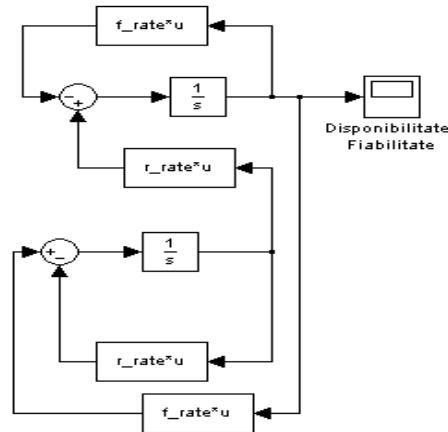


Fig. 4.4.30 Schema Simulink utilizată pentru analiza de fiabilitate și disponibilitate

Primele două experimente realizează o analiză minimală a fiabilității robotului mobil în două situații (situații bazate pe datele furnizate de studiile [113] și [21]): pentru un MTTF de 7 ore, respectiv pentru un MTTF de 19.5 ore. Desigur, pentru o analiză a fiabilității, rata de reparare din ecuațiile (18) se consideră nulă. Practic, ecuația după care se ghidează studiul este:

$$\dot{P}_0(t) = -\lambda_{01}P_0(t) \quad (19)$$

Figurile 4.4.31 și 4.4.32 prezintă evoluția comparativă a fiabilității pentru cele două situații de mai sus. Analiza fiabilității se realizează în condițiile în care robotul mobil este, respectiv nu este prevăzut cu un modul redundant de corecție a orientării bazat pe preluarea și procesarea de imagini.

Notățiile din figuri sunt următoarele:

- analiza referitoare la sistemul inițial -  $S_{\text{initial}}$
- analiza referitoare la sistemul prevăzut cu modulul redundant -  $S_{\text{final}}$

În prima situație (situația 1) având un MTTF de 7 ore, ratele de defectare vor avea următoarele valori:  $\lambda_{01}=0.14285$  pentru sistemul inițial, iar pentru sistemul prevăzut cu modulul redundant  $\lambda_{01}=0.10714$ .

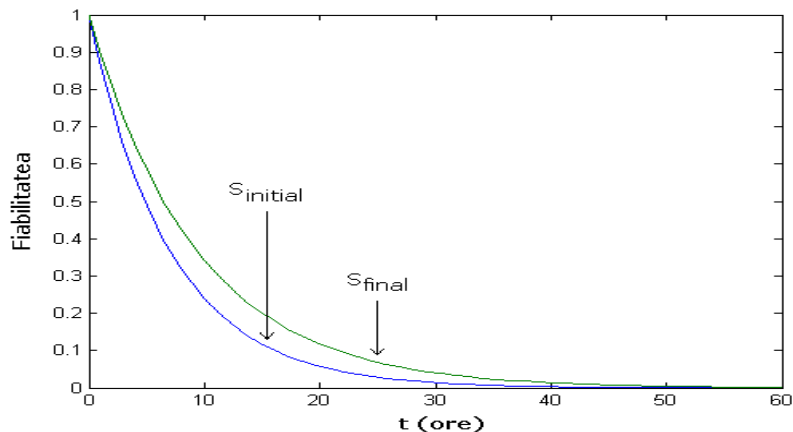


Fig. 4.4.31 Evoluția fiabilității în prima situație analizată (situația 1)



4.4 Aspecte referitoare la creșterea disponibilității și siguranței în deplasarea roboților mobili cu roți 187

A doua situație (situația 2) prezintă un  $MTTF=19.5$  ore, iar ratele de defectare vor avea următoarele valori:  $\lambda_{01}=0.05128$  pentru  $S_{\text{inițial}}$ , iar  $\lambda_{01}=0.03846$  pentru  $S_{\text{final}}$ .

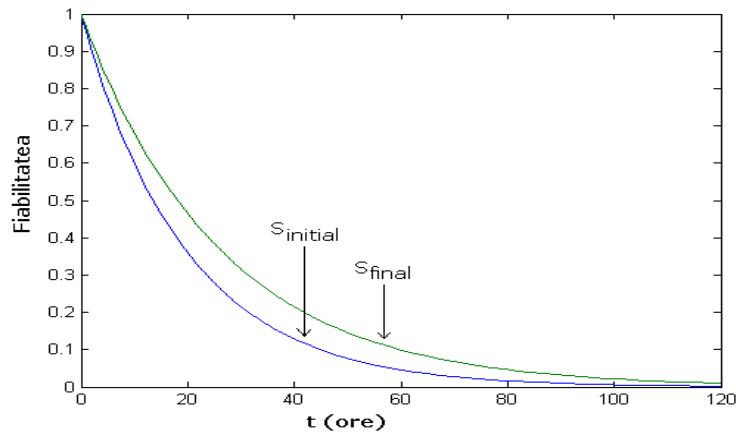


Fig. 4.4.32 Evoluția fiabilității pentru a doua situație analizată (situația 2)

Următoarele două experimente au ca obiectiv studiul comparativ al evoluției disponibilității robotului mobil pentru cele două variante de implementare,  $S_{\text{inițial}}$  și  $S_{\text{final}}$ .

Prima analiză (situația 3) a disponibilității prezintă un  $MTBF$  de 19.5 ore pentru  $S_{\text{inițial}}$ , și în consecință o rată de defectare  $\lambda_{01}=0.05128$ . De asemenea, această situație surprinde cazul cel mai relevant, care realizează analiza disponibilității operaționale considerând indicatorul  $MDT$  de 60.5 ore (așa cum se prezintă în [21]) și în consecință rata de reparare  $\mu_{10}=0.01652$ . Pentru  $S_{\text{final}}$  rata de defectare  $\lambda_{01}=0.03846$ , iar rata de reparare se păstrează constantă cu toate că în mod cert ea va crește în condițiile introducerii modului redundant. Fig. 4.4.33 prezintă evoluția comparativă a disponibilităților pentru această situație, utilizând ecuațiile (18).

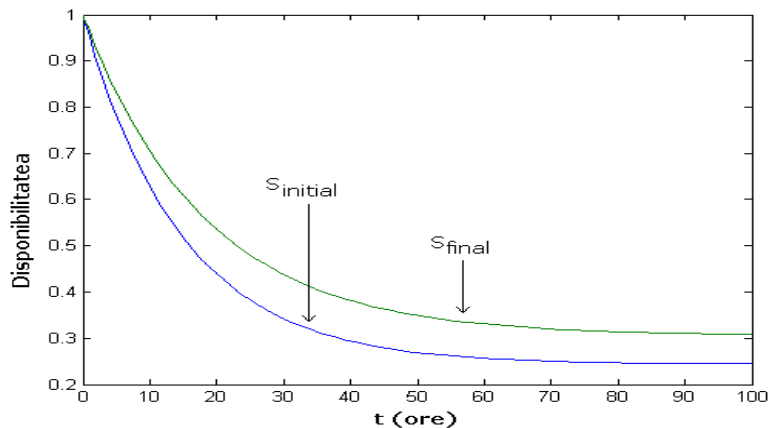


Fig. 4.4.33 Evoluția disponibilității operaționale pentru situația 3

După cum se poate observa din figura 4.4.33, consecința implementării modului redundant pentru corecția orientării prin preluarea și procesarea de imagine este o creștere a disponibilității de aproximativ 7%.

Următoarea analiză (situația 4) prezintă aceleași ipoteze ca și în situația precedentă cu excepția faptului că se consideră rata de reparare influențată de scăderea MTBF. Influența este certă, dar proporția cu care MDT se va modifica este doar presupusă. Astfel se consideră în vederea analizei că MDT va coborî până la 45 de ore și astfel rata de reparare va deveni  $\mu_{10}=0.02222$ . Figura 4.4.34 ilustrează rezultatele analizei:

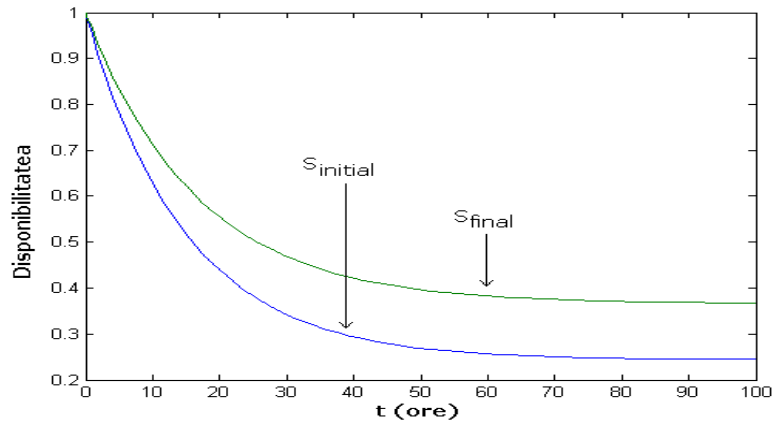


Fig. 4.4.34 Evoluția disponibilității operaționale pentru situația 4

Din figura 4.4.34 se poate observa că disponibilitatea corespunzătoare lui  $S_{final}$  crește față de disponibilitatea pentru  $S_{initial}$  cu aproximativ 13%.

Ultimul experiment (situația 5) realizată se axează pe evoluția disponibilității inerente în contextul unui MTTR de 1.3 ore (valoare furnizată în [21]). Valoarea ratei de reparare va fi în aceste condiții  $\mu_{10}=0.76923$ . Valorile corespunzătoare ratelor de defectare se consideră cele de la situația 3. Se consideră, de asemenea că rata de reparare nu se va modifica în urma implementării modulului redundant de corecție a orientării, fapt ce implică o influență minimă asupra disponibilității (cu toate că în mod cert este influențată și rata de reparare). Figura 4.4.35 ilustrează evoluția comparativă a disponibilităților pentru  $S_{initial}$  și  $S_{final}$ .

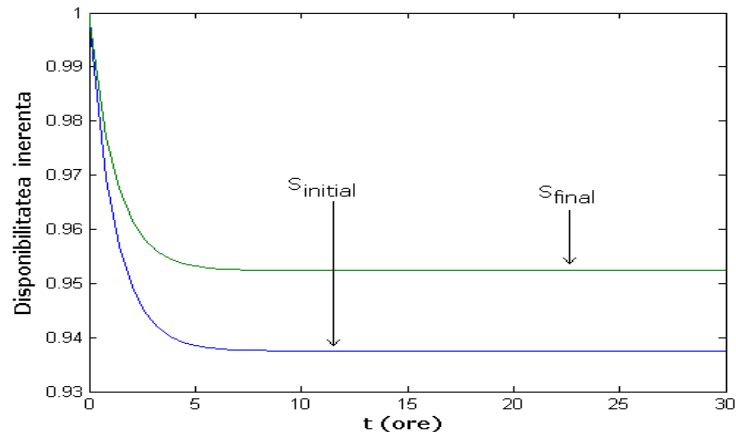


Fig. 4.4.35 Evoluția disponibilității inerente pentru situația 5

Figura 4.4.35 prezintă o creștere a disponibilității inerente corespunzătoare lui  $S_{final}$  față de disponibilitatea inerentă pentru  $S_{initial}$  cu aproximativ 1,5%.

## **5. Concluzii. Contribuții personale. Diseminarea rezultatelor.**

### **Sinteza realizărilor din cadrul tezei**

În cele ce urmează se prezintă în sinteză dezvoltările din cadrul tezei precum și contribuțiile personale.

În capitolul întâi s-au sintetizat noțiuni teoretice referitoare la dependabilitate utilizate în cadrul celorlalte capitole.

Capitolul al doilea a prezentat în prima parte clasificări ale structurilor redundante după mai multe criterii și a detaliat anumite tipuri de redundanțe. Pe această bază, în a doua parte a fost dezvoltată, teoretic și practic, structura redundantă m-n-k hibridă, a fost implementată pe un procesor de semnal digital, respectiv a fost aplicată într-un studiu de caz referitor la o acționare electrică. Partea a treia a capitolului a analizat o altă structură redundantă, anume structura omogenă cu k entități reparabile din perspectiva stabilirii relațiilor dintre ratele de tranziție corespunzătoare sistemului privit ca întreg și ratele de tranziție individuale ale componentelor. Totodată s-a propus o metodă de determinare a ratei de reparare a structurii serie, necesară dezvoltărilor practice din capitolul 3. Ultima parte a capitolului a studiat situația în care structura redundantă omogenă cu k entități reparabile admite și tranziții simultane, stabilind relațiile dintre ratele de tranziție ale sistemului privit ca întreg și ratele de tranziție individuale pentru un sistem cu două entități.

Capitolul al treilea a prezentat cercetări referitoare la probleme de disponibilitate dintr-o centrală electrică geotermală. S-a arătat cum s-a realizat implementarea dezvoltărilor din capitolul 2 cu privire la studiul tranzițiilor și a ratelor de tranziție, inclusiv pentru situația în care se admit tranziții simultane, referitoare la structurile cu redundanță omogenă cu k componente reparabile, cu scopul analizei disponibilității. De asemenea, s-a prezentat o soluție de reglare a temperaturii bazată pe interpolare, soluție prin care se asigură atât o reglare performantă, cât și creșterea disponibilității în cadrul blocului vaporizatoare, parte a centralei electrice geotermale. În ultima parte a capitolului s-a dezvoltat și s-a implementat un modul mobil de detecție și diagnoză a defectelor pentru sisteme automate bazat de corelații identificate pentru anumite tipuri de defecte din cazul centralei electrice geotermale.

Capitolul 4, de asemenea cu caracter aplicativ, conține cercetări referitoare la disponibilitatea unor roboți mobili cu roți care se deplasează în spații semicunoscute. S-au studiat din punctul de vedere al avantajelor și dezavantajelor structurile de reglare utilizate în sistemele robotizate și defectele care pot să apară la roboți. Ca alternativă la soluțiile existente a fost conceput și dezvoltat un regulator interpolativ tolerant la defecte pentru roboții mobili cu roți, toleranța la defecte referindu-se la nivelul structurii de achiziție de date care furnizează distanța față de obstacolele mobile. În ultima parte a capitolului s-a propus și s-a implementat o metodă de corecție a direcției de deplasare a robotului prin preluare și procesare de imagine. Studiile s-au încheiat cu analize de fiabilitate și disponibilitate referitoare la structurile menționate anterior.

### Contribuții personale

- realizarea clasificării structurilor redundante după mai multe criterii
- conceperea și dezvoltarea structurii redundante m-n-k hibride
- implementarea structurii m-n-k hibride pe un procesor de semnal numeric și exemplificarea acesteia
- studiul și elaborarea în colaborare a relațiilor între ratele de tranziție a structurii redundante omogene cu k entități reparabile și ratele de tranziție individuale ale entităților componente, precum și determinarea ratei de reparare a unui ansamblu serie
- studiul și elaborarea în colaborare a relațiilor între ratele de tranziție a structurii redundante omogene cu k entități reparabile și ratele de tranziție individuale ale entităților componente în situația în care se admit ratele de tranziție simultane
- realizarea unei analize de disponibilitate în cadrul centralei electrice geotermale referitoare la aplicarea de structuri redundante omogene cu k entități reparabile
- realizarea unei analize de disponibilitate în cadrul centralei electrice geotermale referitoare la aplicarea de structuri redundante omogene cu k entități reparabile în cazul în care se admit tranziții simultane
- conceperea și dezvoltarea unei soluții de reglare a temperaturii bazată pe interpolare în cadrul blocului vaporizatoare al centralei electrice geotermale, soluție utilă și din punctul de vedere al disponibilității
- conceperea și implementarea unui modul mobil de detecție și diagnoză a defectelor pentru sisteme automate, aplicat pentru centrala electrică geotermală, care se bazează pe identificarea unor corelații caracteristice unor tipuri de defecte
- conceperea și dezvoltarea unui regulator tolerant la defecte ale sistemului de achiziție de date pentru un robot mobil cu roți și realizarea analizei de disponibilitate a structurii rezultate
- dezvoltarea și implementarea unei metode de corecție a direcției de deplasare a robotului prin preluarea și procesarea de imagine și analiza fiabilității și disponibilității sistemului rezultat

### Diseminarea rezultatelor prezentate în cadrul tezei

Majoritatea studiilor prezentate pe parcursul tezei au fost valorificate prin lucrări științifice. Lucrările științifice elaborate de autor în calitate de unic autor sau coautor sunt în ordine cronologică următoarele:

**Korodi A.**, Peană L., *DSP Implementation of Interpolative Type Controllers*, Proceedings of the International Symposium on System Theory, Automation, Mechatronic Systems, SINTES 11, Craiova, 23-24 Octombrie, 2003

Peană L., **Korodi A.**, *Some Ways to Implement Interpolative Type Controllers on DSP*, Analele Universității Tehnice, Craiova, nr.27, vol.1, pp. 33-39, 2003 (apărut 2004)

Dragomir T. L., Gabor G., **Korodi A.**, *Some Aspects Regarding Availability of Repairable Structures*, Scientific Bulletin of "Politehnica" University of

---

Timisoara, Trans. On A.C. & C.S., vol. 49 (63), NO. 2, pp. 159 - 166, 2004

Dragomir T. L., Gabor G., **Korodi A.**, *On an Active Redundant Structure With  $k$  Identically Repairable Components*, Proceedings of the 12th International Symposium on Modeling, Simulation and System Identification, SIMSIS 12, Galați, Septembrie 24-25, 2004

**Korodi A.**, *M-N-K Hybrid Type Redundancy Implementation on DSP*, Proceedings of the 46th International Scientific Conference RTU, Section of Electronics and Telecommunications, vol. 5, pp. 16-21, Riga, Letonia, 13-15 Octombrie 2005

**Korodi A.**, Voloșencu C., *M-N-K Hybrid Type Redundancy Applied on an Electrical Drive*, Proceedings of the EDS'05 IMAPS CS International Conference, Brno, Cehia, 15-16 Septembrie 2005

Gabor G., **Korodi A.**, Dragomir T. L., *Increasing Availability for Vaporizers Temperature Stabilization of a Geothermal Power Plant*, Scientific Bulletin of „Politehnica” University of Timișoara – Trans. On A.C. & C.S., vol. 1, pp. 25 - 30, Iunie 2006

**Korodi A.**, *Interpolative Fault Tolerant Controller for a Mobile Robot*, Proceedings of the 12th IEEE International Conference on Methods and Models in Automation and Robotics MMAR 2006, pp. 651-656, Miedzydroje, Polonia, 28-31 August 2006

**Korodi A.**, Dragomir T. L. , Gabor G., *Temperature Control Solution Using Interpolation for a Geothermal Power Plant*, Proceedings of IFAC WS ESC'06 Energy Saving Control in Plants and Buildings, pp. 235-240, Bansko, Bulgaria, 2-5 Octombrie, 2006

**Korodi A.**, Dragomir T. L., *Mobile Fault Detection and Diagnosis Module for Automatic Systems*, Proceedings of the 15th Mediterranean Conference on Control and Automation MED'07, Atena, Grecia, 27-29 Iunie, 2007

O parte din cercetări au fost finanțate prin granturile CNCSIS:

- *Dezvoltarea de structuri automate pentru a crește gradul de dependabilitate a sistemelor de reglare cu aplicații în sisteme industriale (Energetică, Chimie, Aviație, Robotică)*, cod 309, tip A,
- *Cercetare aplicativă pentru dezvoltarea sistemelor de detecție și diagnoză a defectelor, bazată pe logică fuzzy și rețele neurale, cu aplicații în acționarea electrică*, cod 205, tip A,

din echipele cărora a făcut parte și autorul.

# Anexa

## Modulul de prelucrare numerică cu DSP

Anexa se axează pe descrierea unor elemente fizice utilizate în cadrul tezei. Având în vedere că unele dintre implementările din teză au ca centru de greutate procesorul de semnal TMS320C301-50, considerațiile teoretice urmăresc descrierea aspectelor strict necesare pentru dezvoltările practice din cadrul paragrafului 2.2.2. Așadar, prima și cea mai importantă prezentare este cea a plăcii DSP Starter Kit, care înglobează procesorul de semnal, urmând apoi aspectele legate de multiplexor, numărător și demultiplexor.

### Placa DSP Starter Kit

DSP starter kit (DSK) este un kit realizat în jurul procesorului de semnal TMS320C3-50. El cuprinde o serie de componente hard și soft, care împreună formează un sistem puternic și ușor de utilizat pentru dezvoltarea programelor de aplicație.

În primul rând placa se alimentează la 7V și interacționează cu calculatorul prin portul paralel. Cu ajutorul software-ului aferent plăcii, se permite crearea unui cod pe platforma calculatorului și încărcarea lui pe placa DSK, pentru ca mai apoi să se poată rula. Programele aplicative sunt create în limbaj de asamblare.

Lucrarea [110] descrie în detaliu placa DSP Starter Kit. Componentele hardware principale ale plăcii sunt :

- Procesorul de semnal TMS320C31-50
- Interfața analogic-numerică TLC32040

DSK-ul permite accesul la toți pinii procesorului de semnal prin porturile dispuse pe placă. De asemenea se pot conecta diferite plăci sau instrumente opționale la DSK, care ajută sau sunt necesare în dezvoltarea aplicației.

#### *Procesorul de semnal TMS 320C31-50*

Procesorul de semnal TMS320C31 se încadrează între produsele Texas Instruments, anume în generația DSP-urilor C3x. Această generație este de înaltă performanță, iar TMS320C31-50 are o viteză de execuție de 40 ns pe ciclu mașină. Procesorul de semnal TMS 320C31 este cel mai rapid din cele 9 tipuri de procesoare din generația C3x.

Procesorul de semnal este alcătuit din următoarele componente, [107]:

- *Unitatea centrală* compusă din: înmulțitor, unitatea aritmetică și logică (ALU pe 32 de biți), registrul de deplasare (pe 32 biți), magistralele interne (CPU1/CPU2 și REG1/REG2), unitatea aritmetică a regiștrilor auxiliari (ARAU), regiștrii CPU
- *Memorie internă Cache* de 64 x 32 biți
- *Două blocuri de memorie RAM* de 1K x 32 biți fiecare
- *Două magistrale la memoria program*: de adrese (PADDR) și de date (PDATA) separate
- *Trei magistrale la memoria de date*: de adrese (DADDR 1,2) și de date (DDATA) separate

- *Două magistrale la DMA: de adrese (DMAADDR) și de date (DMADATA) separate*
- *Logică de control a întreruperilor*
- *Două temporizatoare pe 32 biți*
- *Un port serial configurabil pe 8, 16, 24, 32 biți*
- *Controler DMA*

În continuare se vor prezenta doar elemente care sunt în directă legătură cu implementarea practică din cadrul paragrafului 2.2.2.

TMS320C3x conține 28 de *registri ai unității centrale* în care nu a fost inclus registrul de control al programului, PC, [107]. Toți acești regiștri pot fi folosiți de înmulțitor și de ALU, și pot fi de asemenea utilizați ca regiștri generali pe 32 de biți. Cei 8 regiștri auxiliari suportă o serie de adresări indirecte și pot fi folosiți de asemenea ca regiștri întregi pe 32 de biți sau regiștri logici, de uz general. Regiștrii rămași folosesc unor funcții de sistem cum ar fi adresările, managementul stivei, starea procesorului, întreruperile și iterațiile.

Tabelul următor conține o scurtă enumerare a regiștrilor unității centrale.

Registrul	Denumire
R0 - R7	Regiștri de precizie extinsă 0 - 7
AR0 - AR7	Regiștri auxiliari 0 - 7
DP	Registrul pointare la pag. de date
IR0	Registrul de indexare 1
IR1	Registrul de indexare 2
BK	Registrul dimensiune bloc de date
SP	Registrul pointare la stiva
ST	Registrul de stare
IE	Registrul de activare a întreruperilor CPU/DMA
IF	Registrul fanioanelor întrerupere
IOF	Registrul fanioanelor de I/O
RS	Adresa de start bloc de repetat
RE	Adresa de sfârșit bloc de repetat
RC	Registrul contor al iterației

În cadrul TMS320C31 se disting următoarele *periferice*, [107]:

- 2 Temporizatoare
- 1 Port serial
- 1 Controller DMA

Aceste module sunt controlate prin regiștri de memorie de pe magistrala periferică dedicată. Figura A1 ilustrează modulele periferice.

Procesorul TMS320C31 pune la dispoziția utilizatorului două *temporizatoare* de uz general pe 32 de biți. Fiecare temporizator are două moduri de funcționare, utilizând un semnal de tact intern sau extern. Temporizatoarele pot furniza un semnal procesorului C31 sau în exterior la intervale de timp prestabilite sau pentru a număra evenimentele externe. Cu un semnal de tact de sursă intern, temporizatoarele pot semnala unui convertor A/D efectuarea unei conversii, sau pot întrerupe controllerul DMA pentru a începe un transfer de date. Fiecare din cele două temporizatoare are un pin care poate fi utilizat ca pin de intrare, ieșire de tact sau ca pin de intrare/ieșire de uz general.

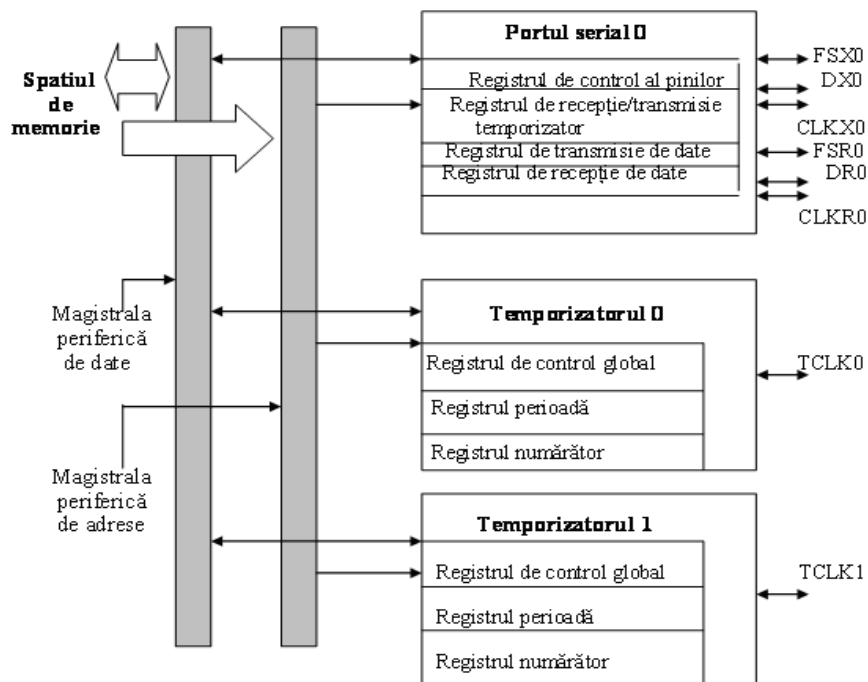


Fig. A1 Modulele periferice ale procesorului de semnal digital

Fiecare temporizator constă dintr-un numărator pe 32-biți, un comparator, un circuit selector al tactului de intrare și un generator de impulsuri. Temporizatoarele folosesc trei regiștri de control:

- Registrul de control global
- Registrul perioadă
- Registrul numărător

Fiecare registru este încărcat software cu o valoare care determină funcționarea temporizatorului în cauză.

#### Interfața analogic-numerică TLC32040 (AIC)

Interfața TLC32040 de pe placa DSK conține, [109]:

- Un singur canal intrare/ieșire, interfața analogică pe 14 biți ADC (analog – digitală) și DAC (digital – analogică). Viteza de conversie ADC și DAC este de 20.000 de eșantioane pe secundă și cu o precizie pe 14 biți
- Un filtru de reconstituire a semnalului la ieșire
- Un filtru de intrare
- Canal de intrare analogic auxiliar selectabil

Interfața TLC32040, [108] este conectată la portul serial al procesorului de semnal TMS320C31. Cei doi pini adiționali ai procesorului de semnal care controlează semnalul de „reset” și de „clock” al interfeței sunt:

- pinul TCLK0 al temporizatorului 0 care furnizează semnalul de tact al interfeței AIC
- pinul XF0 al procesorului de semnal care resetează interfața.

După ce semnalul de tact este furnizat de către procesor și interfața AIC este resetată programatorul poate inițializa interfața la o anumită frecvență de eșantionare/conversie.



## Multiplexorul MAX 307

Figura A2 ilustrează pinii multiplexorului MAX 307:

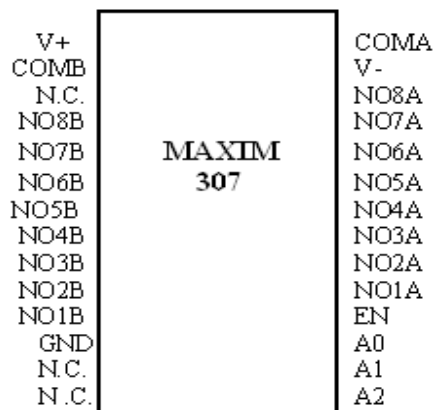


Fig. A2 Pinii multiplexorului MAX 307

Alimentarea se face prin conectarea unei tensiuni de +15V la pinul V+, respectiv -15V la pinul V-, iar pinul GND se conectează la masă.

Resetarea și pornirea multiplexorului se face prin acționarea asupra pinului EN. Dacă EN este setat la valoarea 1, MAX 307 este pornit.

Intrările multiplexorului sunt în număr de 8 și anume: NO1A, NO2A, ..., NO8A sau NO1B, NO2B, ..., NO8B, iar ieșirea este COMA sau COMB. MAX 307 are 3 pini de selecție: A2, A1, A0. Prin setarea/resetarea acestora se face selecția uneia din cele 8 intrări așa cum urmează :

A2	A1	A0	EN	La alegere intrarea
X	X	X	0	Nici una
0	0	0	1	1
0	0	1	1	2
0	1	0	1	3
0	1	1	1	4
1	0	0	1	5
1	0	1	1	6
1	1	0	1	7
1	1	1	1	8

## Numărătorul CD4040BE

Numărătorul CD4040BE [110] furnizează prin ieșirile  $Q_i$  cifre binare, ca urmare a acțiunii unui semnal la pinul IO. Pe frontul căzător al pulsului de la pinul IO se realizează o incrementare. Circuitul poate număra binar până la maxim  $2^{12}$ .

La orice moment în urma acțiunii asupra pinului RESET se poate reinițializa numărătoarea. Reinițializarea se face introducând valoarea „1” la RESET, în restul timpului RESET trebuind să aibă valoarea „0”.

Figura A3 ilustrează circuitul numărător CD4040BE:

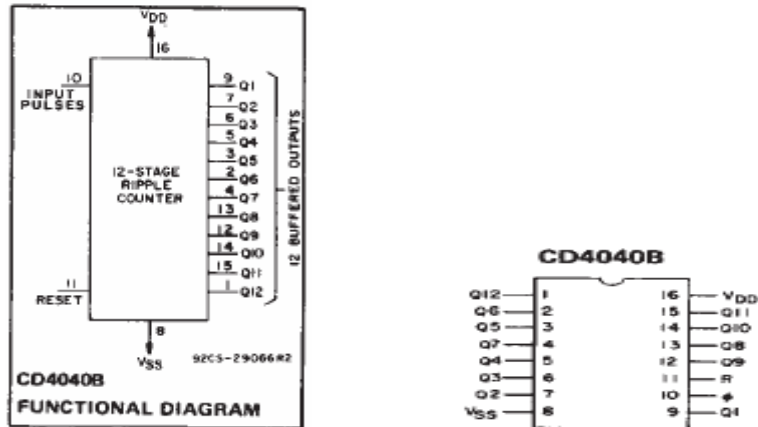


Fig. A3 Numărătorul CD4040BE

### Demultiplexorul CD4051BE

Demultiplexorul analogic CD4051BE [111] este un demultiplexor cu 8 canale, având trei intrări binare de selecție A, B, C și o intrare implicită care activează circuitul (INH). Pentru ca să se poată realiza demultiplexarea, la pinul INH trebuie să fie setată valoarea „0”. Intrarea care va fi demultiplexată este preluată la pinul COM iar ieșirea circuitului se poate furniza pe canalele CHANNEL 0-7.

Circuitul demultiplexorului CD4051BE este redat de figura A4:

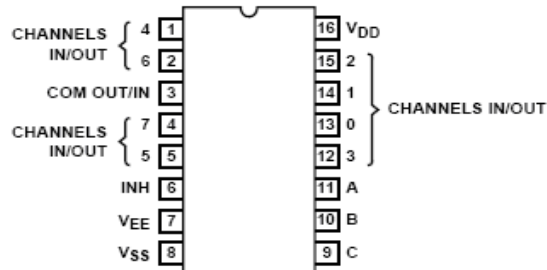


Fig. A4 Demultiplexorul CD4051BE

Selecția uneia dintre cele 8 ieșiri se realizează după cum se prezintă în continuare:

C	B	A	INH	Ieșirea
X	X	X	0	Nici una
0	0	0	1	1
0	0	1	1	2
0	1	0	1	3
0	1	1	1	4
1	0	0	1	5
1	0	1	1	6
1	1	0	1	7
1	1	1	1	8

## Bibliografie

- [1] Akella, M., *Vision-Based Adaptive Tracking Control of Uncertain Robot Manipulators*, IEEE Trans. on Robotics, Vol. 21, No. 4, pp. 748-753, August, 2005
- [2] Antal, C., *Improvement of the geothermal power plant parameters from the University of Oradea*, Ph.D. thesis, University of Oradea, 1999.
- [3] Antics M., Sanner, B., *Status of Geothermal Energy Use and Resources in Europe*, Proceedings European Geothermal Congress, Unterhaching, Germany, Iunie, 2007
- [4] Austin, A., Fletcher, L., Zelinsky, A., *Mobile Robotics in the Long Term*, Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS 2001), Maui, Hawaii, October/November, 2001
- [5] Austin, D., Kouzoubov, K., *Robust, Long Term Navigation of a Mobile Robot*, Proc. IARP/IEE-RAS Joint Workshop on Technical Challenges for Dependable Robots in Human Environments, October, 2002.
- [6] Avizienis, A., Laprie, J. C., Randell, B., Landwehr, C., *Basic Concepts and Taxonomy of Dependable and Secure Computing*, IEEE Trans. on Dependable and Secure Computing, vol. 1, no. 1, pp. 11-33, January-March, 2004
- [7] Avizienis, A.: *Architectures of fault tolerant computing systems*, FTCS pp 3-16, 1975
- [8] Avizienis, A.: *N-version approach to fault tolerance software*, IEEE Trans. on Software Eng., Vol. SE-11, No.12, pp. 1491-1501, Dec.1985
- [9] Avizienis, A., Laprie, J.K.: *Dependable computing: from concepts to design diversity*, Proc. IEEE Vol.74, No.5, May 1986
- [10] Baczynski, M., Baczynski, J., *Artificial Visual Landmarks for Mobile Robot Localization*, Proceedings of the 12<sup>th</sup> IEEE International Conference on Methods and Models in Automation and Robotics, 28-31 August, 2006, Miedzydroje, Polonia
- [11] Băjenescu, T., *Fiabilitatea, Disponibilitatea și Mentenabilitatea Sistemelor Electronice complexe*, Editura de Vest Timișoara, 1997
- [12] Bank, D., *An Error Detection Model for Ultrasonic Sensor Evaluation on Autonomous Mobile Systems*, In Proc. of the 11th IEEE Int. Workshop on Robot and Human interactive Communication, ROMAN2002, Berlin, Germany, pp. 288-293, September 25-27, 2002

- [13] Beauchamp, Y., Stobbe, T., *A review of experimental studies on human-robot system situations and their design implications*, Int. J. Human Factors Manuf., vol. 5, pp. 283–302, 1995
- [14] Bentley, J. P.: *Reliability & Quality Engineering*, Addison Wesley Longman, 1999
- [15] Binfet, J., Wilamowski, B., *Microprocessor Implementation of Fuzzy Systems and Neural Networks*, Proceedings of the INNS-IEEE International Joint Conference on Neural Networks, Washington, 2001
- [16] Bonivento, C., Gentili, L., Paoli, A., *Internal model based fault tolerant control of a robot manipulator*, Proc. 43rd IEEE CDC, Paradise Island, Bahamas, December, 2004
- [17] Borenstein, J., Feng, L., *UMBmark: A Benchmark Test for Measuring Odometry Errors in Mobile Robots*, Proceedings of the SPIE Conference on Mobile Robots, Philadelphia, October 22-26, 1995
- [18] Böttcher, W., *Vergleich von Dreipunktreglern mit einem linearen kontinuierlichen PJ-Regler*, Regelungstechnik, 10, 114-119, 210-213, 1962
- [19] Caccavale, F., Villani, L., *Fault Diagnosis and Fault Tolerance for Mechatronic Systems: Recent Advances*, Springer-Verlag Berlin Heidelberg, 2003
- [20] Cardoso, A., Dourado, A., *Supervision and Robust Fault-Tolerant Control of a Nonlinear Reconfigurable Process*, Proceedings of The 12th IEEE International Conference on Methods and Models in Automation and Robotics MMAR, pp. 769–744, Miedzyzdroje, Poland, 28-31 August, 2006.
- [21] Carlson, J., Murphy, R. R., *Reliability Analysis of Mobile Robots*, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2003), pp. 274-281, 2003
- [22] Carlson, J., Murphy, R., Nelson, A., *Follow-up Analysis of Mobile Robot Failures*, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2004), New Orleans, USA, 2004
- [23] Carlson, J., Murphy, R. R., *How UGVs Physically Fail in the Field*, IEEE Trans. on Robotics, Vol. 21, No. 3, pp. 423-437, June, 2005
- [24] Casper, J., et al., *Issues in intelligent robots for search and rescue*, SPIE Ground Vehicle Technology II, Vol. 4, pp. 41–46, 2000
- [25] Certic, J., Milic, L., *Signal Processor Implementation of a Low-Pass/High-Pass IIR Digital Filter with Variable Cutoff Frequency*, Proceedings of the International Conference on Computer as a Tool, vol. 2, pp. 1618-1621, 2005
- [26] Chong, K.S., Kleeman, L., *Accurate Odometry and Error Modelling for a Mobile Robot*, MECSE-1996-6, 1996

- 
- [27] Clauser, C., *Geothermal Energy, Renewable Energies*, 480 – 595, Springer Verlag, Heidelberg-Berlin, 2006
- [28] Cojocaru, D., *Achiziția, Prelucrarea și Recunoașterea imaginilor*, Editura Universitaria Craiova, 2002
- [29] Cortez R.A., Papageorgiou X., Tanner H.G., Klimenko A.V., Borozdin K.N., Priedhorsky W.C., *Experimental Implementation of Robotic Sequential Nuclear Search*, Proceedings of the 15<sup>th</sup> Mediterranean Conference on Control and Automation, Atena, Grecia, 27-29 Iulie, 2007
- [30] Dickson, M. H., Fanelli, M., *Geothermal Energy: Utilization and Technology*, Earthscan, Mai, 2005
- [31] DiPippo, R., *Geothermal Power Plants: Principles, Applications and Case Studies*, Elsevier, Iulie, 2005
- [32] Dixon, W. E., Walker I. D., Dawson D. M., *Fault Detection for Wheeled Mobile Robots with Parametric Uncertainty*, Proceedings of the 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 1245–1250, Como, Italy, July, 2001
- [33] Dragomir, T. L., Dale, S., Balas, M. M., *Some aspects regarding interpolative control*, Proceedings of IWICS, București, pp. 6-19, 29 Noiembrie, 2001
- [34] Dragomir, T. L., Gabor, G., **Korodi, A.**, *Some Aspects Regarding Availability of Repairable Structures*, Scientific Bulletin of "Politehnica" University of Timisoara, Trans. On A.C. & C.S., vol. 49 (63), NO. 2, pp. 159 - 166, 2004
- [35] Dragomir, T. L., Gabor, G., **Korodi, A.**, *On an Active Redundant Structure With k Identically Repairable Components*, Proceedings of the 12th International Symposium on Modeling, Simulation and System Identification, SIMSIS 12, Galați, Septembrie 24-25, 2004
- [36] Dragomir, T.L., *Regulatoare automate*, vol I, IPT, 1986
- [37] Drechsel, D., *Regelbasierte Interpolation und Fuzzy Control*, Vieweg, 1996
- [38] Dumitrache, I., *Ingineria reglării automate*, Edit. Politehnica Press, București, 2005
- [39] Elzer, P., F., Behnke, R., Beuthel, C., *New Techniques for Maintenance and Training in Process Supervision and Control*, IFAC Telematics Applications in Automation and Robotics, Germany, July 2001
- [40] Fairfield, N., Kantor, G., Wettergreen, D., *Real-Time SLAM with Octree Evidence Grids for Exploration in Underwater Tunnels*, Journal of Field Robotics, 2007
- [41] Felea, I., *Ingineria fiabilității în electroenergetică*, Editura Didactică și Pedagogică, R.A. - București, 1996
- [42] Feramez, M., *Reliability Engineering for Communication Systems*, Advanced Communication Systems, Telecommunication Systems Engineering, Lecture 8, September 2003

- [43] Ferrarini, L., Carneiro, J.S.A., Radaelli, S., Ciapessoni, E., Dependability Analysis of Power System Protections using Stochastic Hybrid Simulation with Modelica, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2007), Roma, Italia, pp. 1584-1589, 2007
- [44] Frontoni E., Mancini A., Zingaretti P., *Vision based approach for active selection of robot's localization action*, Proceedings of the 15<sup>th</sup> Mediterranean Conference on Control and Automation, 27-29 Iulie, 2007, Atena, Grecia
- [45] Gabor, G., **Korodi, A.**, Dragomir, T. L., *Increasing Availability for Vaporizers Temperature Stabilization of a Geothermal Power Plant*, Scientific Bulletin of „Politehnica” University of Timișoara – Trans. On A.C. & C.S, vol. 1, pp. 25 - 30, Iunie 2006
- [46] Gabor, G. A., *Contributions to the research on control system availability with applications for the control system of a geothermal power plant*, Ph.D. thesis, “Politehnica” University of Timișoara, 2005.
- [47] Gobble, W. M.: *Evaluating Control Systems Reliability*, Instrument Society of America, 1996
- [48] Goodfellow, J. W., *Applying reliability centered maintenance (RCM) to overhead electric utility distribution systems*, Proceedings of the IEEE Power Engineering Society Summer Meeting, vol. 1, pp. 566-569, Seattle, USA, Iulie, 2000
- [49] Gorunescu, F., Prodan, A., *Modelare Stochastică și Simulare*, Editura Albastră, Cluj Napoca, 2001
- [50] Greifeneder, J., Frey, G., *Dependability Analysis of Networked Automation Systems by Probabilistic Delay Time Analysis*, Proceedings of the 12<sup>th</sup> IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2006), St. Etienne, Franța, vol. 1, pp. 269-274, Mai, 2006
- [51] Gupta, H., Roy, S., *Geothermal Energy – An Alternative Resource for the 21st Century*, Elsevier, 2007
- [52] Hajoon, L., Dongkyung, N., Cheol, H., *A sliding mode controller using neural networks for robot manipulator*, ESANN'2004 Proceedings of the European Symposium on Artificial Neural Networks, pp. 193-198, Brugges, Belgia, 28-30 Aprilie, 2004
- [53] Hallouzi, R., Verdult, V., Babuska R., Verhaegen, M., *Fault Detection and Identification of Actuator Faults Using Linear Parameter Varying Models*, Proceedings of the 16<sup>th</sup> IFAC World Congress, Prague, Czech Republic, 4-8 July, 2005.
- [54] Hocine, A., Maquin, D., Ragot, J., *Finite Memory Observer for Switching Systems: Application to Diagnosis*, Proceedings of the 16<sup>th</sup> IFAC World Congress, Praga, Cehia, 4-8 Iulie 2005
- [55] Iancu, E., Vinătoru, M., *Metode Analitice pentru Detecția și Localizarea Defectelor Studii de Caz*, Editura Universitaria Craiova, 2003

- 
- [56] Ivănescu, M., Cojocaru, D., Diaconu, I., *Introducere în Mecatronică*, Editura Universitaria Craiova, 2002
- [57] Iyer, R. K., *Hardware Redundancy*, Design of High Availability Systems and Networks, ECE 442, CS 436, Lecture 2, June 28, 2004
- [58] Johnson, B. W., *Design and Analysis of Fault-Tolerant Digital Systems*, Addison-Wesley Publishing Company, 1993
- [59] Kacprzyk, J., *Interpolation Reasoning in Optimal Fuzzy Control*, Fuzzy-IEEE'93, San Fransisco, pp. 1259-1263, 1993
- [60] Kim, M. S., Shin, J. H., Lee, J. J., *Design of a robust adaptive controller for a mobile robot*, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 3, pp. 1816-1821, Takamatsu, Japan, 2000
- [61] Kóczy, L. T., Hirota, K., *Interpolation in Structured Fuzzy Rule Bases*, Fuzzy-IEEE'93, San Fransisco, pp. 402-405, 1993
- [62] **Korodi, A.**, Peană, L., *DSP Implementation of Interpolative Type Controllers*, Proceedings of the International Scientific Symposium SINTES11, Craiova, 2003
- [63] **Korodi, A.**, *M-N-K Hybrid Type Redundancy Implementation on DSP*, Proceedings of the 46<sup>th</sup> International Scientific Conference RTU, Section of Electronics and Telecommunications, vol. 5, pp.16-21, Riga, Letonia, October 2005
- [64] **Korodi, A.**, Voloşencu, C., *M-N-K Hybrid Type Redundancy Applied on an Electrical Drive*, Proceedings of the EDS'05 IMAPS CS International Conference, Brno, Cehia, 15-16 Septembrie, 2005
- [65] **Korodi, A.**, Dragomir, T. L., Gabor, G., *Temperature Control Solution Using Interpolation for a Geothermal Power Plant*, Proceedings of IFAC WS ESC'06 Energy Saving Control in Plants and Buildings, pp. 235-240, Bansko, Bulgaria, 2-5 Octombrie, 2006
- [66] **Korodi, A.**, *Interpolative Fault Tolerant Controller for a Mobile Robot*, Proceedings of the 12<sup>th</sup> IEEE International Conference on Methods and Models in Automation and Robotics MMAR, pp. 651-656, Miedzyzdroje, Poland, 28-31 August, 2006
- [67] **Korodi, A.**, Dragomir, T. L., *Mobile Fault Detection and Diagnosis Module for Automatic Systems*, Proceedings of the 15th Mediterranean Conference on Control and Automation MED'07, Atena, Grecia, 27-29 Iunie, 2007
- [68] Koshaev, D. A., *Efficient Fault Detection, Estimation and Recovery Using an Augmented Kalman Filter*, Proceedings of the 13th IEEE IFAC International Conference on Methods and Models in Automation and Robotics, pp. 813-818, Szczecin, Poland, 27-30 August 2007
- [69] Laprie, J. C., *Dependability: Basic Concepts and Terminology*, Springer-Verlag, 1992
- [70] *Larousse Dicționar Enciclopedic*, 1995

- [71] Latzel, W., *Regelungen mit Prozessrechner*, BI, Zürich, 1977
- [72] Lee, C. S., Wang, J., Yuh, J., *Self-Adaptive neuro-fuzzy sitemaps with fast parameter learning for autonomous underwater vehicle control*, Int. J. Adv. Robot., Vol. 15, No. 5, pp. 589–608, 2001
- [73] Leitch, R. D.: *Reliability Analysis for Engineers*. Oxford University Press, 1995
- [74] Leuschen, M. R., Walker, I. D., Cavallaro, J. R., *Robotic Fault Detection Using Nonlinear Analytical Redundancy*, IEEE International Conference on Robotics and Automation, Washington D.C., Mai, 2002
- [75] Long, M., Murphy, R., Parker, L., *Distributed Multi-Agent Diagnosis and Recovery from Sensor Failures*, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS), Vol. 3, pp. 2506-2513, Octombrie 2003.
- [76] Lund, J., *100 Years of Renewable Electricity*, Renewable Energy World, Iulie–August 2004
- [77] Lund, J., *100 years of renewable electricity—geothermal power production*, Renewable Energy World 8(4): 252-259, 2005
- [78] Lunze, J., Steffen, T., *Rekonfiguration Linearer Systeme bei Aktor und Sensorfehlern*, 2003
- [79] Maghiar, T., Antal, C., Gabor, G., Gavrilescu, O., Popa, M., Crăciun, D., Bococi, D., *Centrale electrice geotermale cu fluid secundar*, Editura Universității din Oradea, 2002
- [80] Marusak, P. M., *Actuator Fault Toleration in Control Systems with Analytical Predictive Controllers and Output Constraints*, Proceedings of the 13th IEEE IFAC International Conference on Methods and Models in Automation and Robotics, pp. 825-832, Szczecin, Poland, 27-30 August 2007
- [81] McIntyre, M. L., Dixon, W.E., Dawson, D. M., Walker, I. D., *Fault Detection and Identification for Robot Manipulators*, Proceedings of the 2004 IEEE Conference on Robotics and Automation (ICRA), pp. 4981-4986, New Orleans, Louisiana, USA, April 2004
- [82] Melchior, N. A., Smart, W. D., *Autonomic Systems for Mobile Robots*, First International Conference on Autonomic Computing (ICAC'04), pp. 280-281, New York, USA, 2004
- [83] Meyna, A., *Zuverlässigkeitsbewertung zukunftsorientierter technologien*, 1994
- [84] Mohamed, M. G., Gosselin, C. M., *Design and Analysis of the Kinematically Redundant Parallel Manipulators with Configurable Platforms*, IEEE Trans. on Robotics, Vol. 21, No. 3, pp. 277-287, June, 2005
- [85] Morris, N., *Geothermal Power*, Black Rabbit Books, 2006



- 
- [86] Muller, A., *Internal Preload Control of Redundantly Actuated Parallel Manipulators: Its Application to Backlash avoiding Control*, IEEE Trans. on Robotics, Vol. 21, No. 4, pp. 668-677, August, 2005
- [87] Ngo, T. D., Raposo, H., Schioler, H., *Being Sociable: Multirobots with Self-sustained Energy*, Proceedings of the 15<sup>th</sup> Mediterranean Conference on Control and Automation, Atena, Grecia, 27-29 Iulie, 2007
- [88] Nourbakhsh, I. R., *The mobot museum robot*, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2002, Workshop Robots in Exhibitions, pp. 14-19, Lausanne, Elveția, 2002
- [89] Nourbakhsh, I. R., Kunz, C., Willeke, T., *The Mobot Museum Robot Installations: A Five Year Experiment*, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2003, Las-Vegas, USA, 2003
- [90] O'Connor, P. D., *Practical Reliability Engineering*, John Wiley and Sons Ltd. England, 1991
- [91] Oniga, S., *Sistem senzorial pentru recunoașterea gesturilor unei mâini utilizând rețele neuronale implementate în FPGA*, Teză de doctorat, 2005
- [92] Pakstas, A., Schagaev, I., Zalewski, J., *Redundancy Classification for Fault-Tolerant Computer Design*, IEEE 2001 International Conference on Systems, Man and Cybernetics, Tucson, Arizona, , pp. 3193-3198, October 7-10, 2001
- [93] Parkhomenko, P. P., *About redundancy classification*, Preprint of ICSAN, pp.1-7, 1988
- [94] Peana, L., **Korodi, A.**, *Some Ways to Implement Interpolative Type Controllers on DSP*, Analele Universitatii din Craiova, Nr. 27, Vol. 1, pp. 33-39, 2003 (apărut 2004)
- [95] Phillips, D., *Image Processing in C*, R&D Publications, 2000
- [96] Preßler, G., *Regelungstechnik*, BI, Mannheim-Wien-Zürich, 1967
- [97] Rabiner, Gold, B., *Theory and Application of Digital Signal Processing*, Prentice Hall International, 1975
- [98] Sala, A., Albertos, P., *Fault-Tolerant Time-Invariant Feedback Control*, Proceedings of the 16<sup>th</sup> IFAC World Congress, Prague, Czech Republic, 4-8 July, 2005.
- [99] Saluja, K.: *Simple Concepts in Fault Tolerance*, ECE 753 Fault-Tolerant Computing, May 2004
- [100] Schwall, M. L., Baker, B., Gerdes, J. C., Forchert, T., *A Probabilistic Vehicle Diagnostic System Using Multiple Models*, Proceedings of the Fifteenth Conference on Innovative Applications of Artificial Intelligence, Acapulco, Mexico, 12-14 August, 2003
- [101] Seem, J. E., House, J. M., *Integrated Control and Fault Detection of Air-Handling Units*, Proceedings of IFAC WS ESC'06 Energy Saving

- Control in Plants and Buildings, Bansko, Bulgaria, October 2-5, 2006.
- [102] Song, F., Smith, S., *Design of sliding mode fuzzy controllers for an autonomous underwater vehicle without system model*, Proc. IEEE Conf. OCEANS, Providence, USA, pp. 835–840, Sep. 2000
- [103] Stachera, K., Schumacher, W., Herbst, O., *Automatic Identification of Odometry Parameters of a Two-Wheel Mobile Robot Using Ultrasonic Sensors*, Proceedings of the 12<sup>th</sup> IEEE International Conference on Methods and Models in Automation and Robotics, 28-31 August, 2006, Miedzyzdroje, Polonia
- [104] Stancliff, S. B., Dolan, J., Trebi-Ollennu, A., *Mission Reliability Estimation for Multirobot Team Design*, Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2206 – 2211, Octombrie 2006.
- [105] Staroswiecki, M., *Fault Tolerant Control: The Pseudo-Inverse Method Revisited*, IEEE Trans. on Robotics, Vol. 21, No. 4, pp. 668-677, August, 2005
- [106] Starr, A., Wynne, R., Kennedy, I., *Failure analysis of mature robots in automated production*, Proc. Inst. Mech. Eng.—Pt. B (J. Eng. Manuf.), vol. 213, pp. 813–824, 1999
- [107] TI, *TMS320C3x User's Guide*, 1994
- [108] TI, *TLC32040 Data Sheet*, 1995
- [109] TI, *TMS320C3x DSP Starter Kit – User's Guide*, 1996
- [110] TI, *CD4040BE Data Sheet*, 2003
- [111] TI, *CD4051 Data Sheet*, 2003
- [112] Thongchai, S., Kawamura, K., *Application of Fuzzy Control to a Sonar-Based Obstacle Avoidance Mobile Robot*, Proc. of the IEEE International Conference on Control Applications, Anchorage, Alaska, USA, Sept. 25-27, 2000
- [113] Tomatis, N. et al., *Design and system integration for the expo.02 robot*, Proc. IEEE/RSJ IROS 2002 Workshop Robots in Exhibitions, pp. 67–72, 2002
- [114] Verma, V., Gordon, G., Simmons, R., Thrun, S., *Real-time fault diagnosis (robot fault diagnosis)*, Robotics & Automation Magazine, vol. 11, no. 2, pp. 56 – 66, Iunie, 2004
- [115] VEM – *Handbuch zuverlassigkeit von automatisierungs und elektroenergieanlagen*, Jurgen Nikolaizik, VEB Verlag, Berlin, 1981
- [116] Vînătoru, M., *Conducerea Automată a Proceselor Industriale*, vol. 1, Editura Universitaria Craiova, 2001
- [117] Vînătoru, M., *Conducerea Automată a Proceselor Industriale*, vol. 2, Editura Universitaria Craiova, 2005

- 
- [118] Vladu, E., Dragomir, T. L., *Controller tuning using genetic algorithms*, Proc. of SACI – Symposium on Applied Computational Intelligence, Szeged – Arad, pp. 29-38, 2004.
- [119] *WiRobot X80 User Manual*, [www.drrobot.com](http://www.drrobot.com), Ianuarie, 2006
- [120] Wong, C. C., Huang, B. C., Lai, H. R., *Genetic-Based Sliding Mode Fuzzy Controller Design*, Tamkang Journal of Science and Engineering, Vol. 4, No. 3, pp. 165-172, 2001
- [121] [www.weibull.com](http://www.weibull.com)
- [122] Yang, S. K., *A Condition-Based Failure-Prediction and Processing-Scheme for Preventive Maintenance*, IEEE Trans. on, Reliability, vol.52, pp. 373-383, September 2003
- [123] Yoerger, D., Slotine, J., *Adaptive sliding control of an experimental underwater vehicle*, Proc. IEEE Int. Conf. Robotics and Automation, Sacramento, USA, pp. 2746–2751, Apr. 1991
- [124] Yuta, S., Asama, H., Thrun, S., Prassler, E., Tsubouchi, T., *Field and Service Robotics: Recent Advances in Research and Applications*, Springer Tracts in Advanced Robotics, Berlin, 2006.
- [125] Zadeh, L. A., *Interpolative reasoning as a common basis for inference in Fuzzy Logic, Neural Network theory and Calculus of fuzzy If/Then Rules*, Opening Talk, 2<sup>nd</sup> International conference on Fuzzy Logic and Neural Networks, Iizuka, 1992
- [126] Zeitz, K. H., *Regelungen mit Zwei- und Dreipunktreglern*, Oldenbourg, 1986
- [127] Zhao, S., Yuh, J., *Experimental Study on Advanced Underwater Robot Control*, IEEE Trans. on Robotics, Vol. 21, No. 4, pp. 695-703, August, 2005
- [128] Zingaretti, P., Frontoni, E., *Comparison and fusion of vision and range measurements for robot pose estimation*, Proceedings of the 15<sup>th</sup> Mediterranean Conference on Control and Automation, 27-29 Iulie, 2007, Atena, Grecia
- [129] Zunguo, H., Xicheng, L., *A Diversified Dynamic Redundancy Method Exploiting the Intrusion Tolerance*, ISW 2000, Boston, USA, October 2000

## Index

- amplificare de compatibilizare 56
- caracteristici pe performanță 22, 23
- centrală electrică geotermală 74, 75
- defect brusc 144-146
  - evolutiv 144-146
  - permanent 144-146
- dependabilitate 13, 15
- disponibilitate 15-19
  - inerentă 17
  - rezultată 17
  - operațională 17
- erori de odometrie 175, 176
- factor de corecție 145
- fiabilitate 15, 16
- limite de specificație 16, 20, 145, 151
- loc de sprijin 88, 89, 124-126, 137, 153, 155
- mecanism de corecție 134, 141, 145, 152-156
- mentenanța breakdown 20
  - preventivă 21
  - on-condition 21
- model Markov 18, 19, 62, 68, 72, 77, 78, 167, 185
- modul mobil de detecție și diagnoză 107-116
- non-disponibilitate 16
- non-fiabilitate 16
- procesare de imagini 170, 176
- punct de sprijin 88, 89, 124-126, 137, 153
- rată de defectare 19, 62, 68, 77, 87, 167, 187
  - reparare 19, 62, 68, 77, 87, 167, 187
  - tranziție simultană 71-73, 84
- redundanță 13, 18, 24, 25, 27-34, 61, 62, 64, 71, 76-85, 102, 103
  - analitică 44, 107
  - asistență automatizată 44
  - bazată pe referință 134, 141-152
  - de timp 39
  - hibridă 42
  - mascată 34
  - m-n-k hibridă 48-61
  - omogenă 34, 61
  - predictivă 37
  - standby cold 36
  - standby propriu-zisă 37
  - standby hot 36
- regulator interpolativ 87, 89, 124-126
  - interpolativ propriu-zis 134-140
  - interpolativ tolerant la defecte 134-167
- roboți mobili 13, 25, 119, 134, 170, 171
  - în medii cu hazard 128
  - în spații semicunoscute 131
  - manipulatoare 133
- sisteme automate 13, 25
- sisteme reparabile 16
  - parțial reparabile 16
  - total reparabile 16
- structură serie 18, 69
  - paralel 19, 34
- timp mediu până la defectare 16
  - între defecțiuni 16, 130-133, 167, 185-188
  - de reparare 17, 133, 185, 188
  - de cădere 17, 64, 67, 130-133, 167, 185-188
- toleranță la defecte 24, 27, 107, 134
- valoare de sprijin 88, 89, 124-126, 137, 153

## Titluri recent publicate în colecția „TEZE DE DOCTORAT” seria 1: Automatică

---

1. **Sanda Dale** – *Contribuții la studiul sistemelor de reglare cu reglatoare de tip interpolativ*, ISBN (10) 973-625-376-7, (13) 978-973-625-376-8, (2006).
2. **Alina Bogan-Marta** – *Tehnici de recunoaștere automată în analiza și determinarea similarității secvențelor biologice*, ISBN 978-973-625-415-4, (2007).
3. **Mihaela Răduca** – *Contribuții la identificarea, modelarea și simularea sistemelor de izolație ale hidrogenatoarelor*, ISBN 978-973-625-467-3, (2007).
4. **Dan Ungureanu** – *Contribuții la modelarea și implementarea sistemelor cu evenimente discrete cu aplicații directe asupra sistemelor de transport cu zone de acumulare*, ISBN 978-973-625-482-6, (2007).
5. **Dorin Berian** – *Structuri și strategii asociate sistemelor informatice pentru asistența medicală primară*, ISBN 978-973-625-503-8, (2007).
6. **Florin Drăgan** – *Analiza comportamentului de tip haotic al sistemelor cu aplicații în studiul unor convertoare electronice*, ISBN 978-973-625-504-5, (2007).



EDITURA POLITEHNICA