SISTEM DE CONDUCERE ORIENTAT PE STRUCTURI DE CALCUL EVOLUTIV

Constantin ABABII[1], Eugen NEGARA[2], Dimitrie BORDIAN[2], Dmitri CALUGARI[3], Andrei MIRON[2], Neonil ROŞCA[2], Iulian LUNGU[2]

[1]SCHUNK Electronic Solutions GmbH, Germany;
[2]Universitatea Tehnică a Moldovei, R. Moldova; [3]ICG Engineering, R. Moldova;

Abstract: *În lucrare sunt prezentate rezultatele proiectării unui sistem de conducere orientat pe structuri de calcul evolutiv. Soluţionarea problemei de conducere se bazează pe aplicarea algoritmilor de calcul evolutiv în combinaţie cu calculul membranar şi implementarea acestora pe structuri de calcul multi-procesor sau multi-controler. S-a formulat problema de optimizare multi-criterială, şi s-au elaborat: modelul matematic pentru soluţionarea problemei de optimizare multi-criterială, structura sistemului de calcul membranar-evolutiv, şi structura sistemului de calcul multi-procesor.*

Cuvinte cheie: *Calcul evolutiv, optimizare multi-dimensională, optimizare multi-criterială, structuri de calcul, sistem multi-procesor, calcul membranar.*

Introducere

Lumea ce ne înconjoară este o lume multi-dimensională [1]. Conducerea unui proces definit în această lume poate fi considerat ca un proces multi-dimensional în care este definită o problemă de optimizare multi-criterială [2,3]. În acest caz este necesar să fie optimizate simultan mai multe funcţii obiectiv, care se află, în unele cazuri, în conflict de interese. În acelaşi timp unele funcţii obiectiv necesită ca să fie minimizate, pe când altele, necesită ca să fie maximizate. De asemenea valoarea optimală pentru anumite funcţii obiectiv nu coincide cu valoarea optimală realizată de alte funcţii obiectiv.

Găsirea unui model matematic sau algoritm clasic care să soluţioneze contrazicerile descrise mai sus este foarte complicat şi practic irealizabil. O metodă de soluţionare poate fi găsită prin aplicarea algoritmilor de calcul evolutiv [4], în combinaţie cu calculul membranar [5], fiind implementaţi pe structuri de calcul reconfigurabil multi-procesor sau multi-controler [6].

1. Formularea problemei de optimizare multi-criterială

Este definit procesul $\boldsymbol{P}$ în spaţiul $\Re^{\boldsymbol{N}}$, unde $\boldsymbol{N}$ este dimensiunea spaţiului [3]. Starea procesului $\boldsymbol{P}$ este identificată prin vectorul $\boldsymbol{X} = \left[\boldsymbol{x_i}, \forall \boldsymbol{i} = \overline{1, \boldsymbol{N}} \right]$, unde $\boldsymbol{X} \in \Re^{\boldsymbol{N}}$. Optimizarea procesului $\boldsymbol{P}$ în spaţiul $\Re^{\boldsymbol{N}}$ este efectuată prin intermediul vectorului de control $\boldsymbol{Y} = \left[\boldsymbol{y_j}, \forall \boldsymbol{j} = \overline{1, \boldsymbol{N}} \right]$, unde $\boldsymbol{Y} \in \boldsymbol{R^N}$ care este spaţiul de resurse admisibile pentru controlul procesului $\boldsymbol{P}$.

Este pusă problema (1) care să asigure:

$$\boldsymbol{f}(\boldsymbol{X}) \xrightarrow{\boldsymbol{Y}} \boldsymbol{X}^{\boldsymbol{opt}}; \forall \boldsymbol{X} \in \Re^{\boldsymbol{N}} \ \& \boldsymbol{Y} \in \boldsymbol{R}^{\boldsymbol{N}}. \tag{1}$$

În modelul (1) sunt menţionate $\boldsymbol{f}(\boldsymbol{X})$ - evoluţia procesului $\boldsymbol{P}$ şi $\boldsymbol{X}^{\boldsymbol{opt}}$ - valoarea optimală a acestuia.

Vom considera că $\boldsymbol{f}(\boldsymbol{X}) = \boldsymbol{f}^{\min}(\boldsymbol{X}) \cup \boldsymbol{f}^{\max}(\boldsymbol{X})$, unde $\boldsymbol{f}^{\min}(\boldsymbol{X})$ - mulţimea de funcţii care minimizează valoarea de stare $\boldsymbol{X}^{\min}$ a procesului $\boldsymbol{P}$ şi, respectiv $\boldsymbol{f}^{\max}(\boldsymbol{X})$ - mulţimea de funcţii care maximizează aceste valori $\boldsymbol{X}^{\max}$, $\boldsymbol{X}^{\boldsymbol{opt}} = \boldsymbol{X}^{\min} \cup \boldsymbol{X}^{\max}$.

2. Modelul matematic pentru soluţionarea problemei de optimizare multi-criterială

Soluţionarea problemei (1) se reduce la rezolvarea unui sistem de ecuaţii (2), unde: $\Delta \boldsymbol{y_i}, \forall \boldsymbol{i} = \overline{1, \boldsymbol{N}}$ - variaţia semnalului de acţiune asupra procesului care duce la îndeplinirea condiţiei de optimizare; $\Delta \boldsymbol{x_i}, \forall \boldsymbol{i} = \overline{1, \boldsymbol{N}}$ - variaţia stării procesului care duce la îndeplinirea condiţiei de optimizare; $\boldsymbol{X}$ - starea procesului controlat; $\boldsymbol{X}^{\min}$ - starea procesului optimală-minimală; $\boldsymbol{X}^{\max}$ - starea procesului optimală-maximală; $\boldsymbol{f_i}^{\min}, \forall \boldsymbol{i} = \overline{1, \boldsymbol{k}}$ - funcţie pentru verificarea condiţiei de minimizare a stării procesului; $\boldsymbol{f_i}^{\max}, \forall \boldsymbol{i} = \overline{\boldsymbol{k}+1, \boldsymbol{N}}$ - funcţie pentru verificarea condiţiei de maximizare a stării procesului.

$$
\begin{cases}
\Delta y_1 \left| f_1^{\min}(X, \Delta x_1) \to X^{\min}, \right. \\
\Delta y_2 \left| f_2^{\min}(X, \Delta x_2) \to X^{\min}, \right. \\
\ldots \\
\Delta y_k \left| f_k^{\min}(X, \Delta x_k) \to X^{\min}, \right. \\
\Delta y_{k+1} \left| f_{k+1}^{\max}(X, \Delta x_{k+1}) \to X^{\max}, \right. \\
\Delta y_{k+2} \left| f_{k+2}^{\max}(X, \Delta x_{k+2}) \to X^{\max}, \right. \\
\ldots \\
\Delta y_N \left| f_N^{\max}(X, \Delta x_N) \to X^{\max}. \right.
\end{cases} \qquad (2)
$$

3. Structura sistemului de calcul membranar-evolutiv

Structura sistemului de conducere orientat pe calcul evolutiv este prezentată în Figura 1.

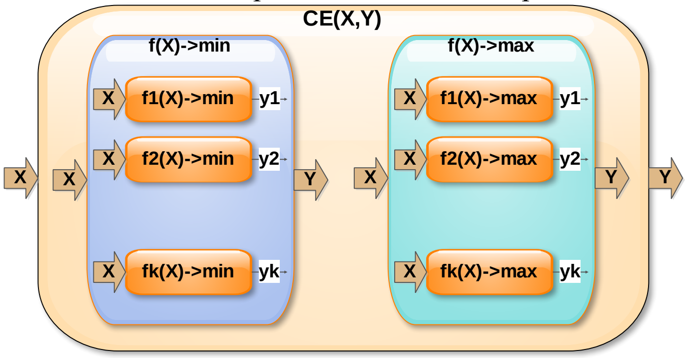

Fig. 1. Structura sistemului de conducere orientat pe aplicaţii evolutive.

Structura de calcul evolutiv prezintă un model de calcul membranar format din: X - porturi de intrare a vectorului de stare; Y - porturi de ieşire a vectorului de acţiune; $f_i(X) \to \min$ - membrane elementare care soluţionează problema spre o valoare minimală; $f(X) \to \min$ - membrană complexă care integrează toate membranele de minimizare; $f_i(X) \to \max$ - membrare elementare care soluţionează problema spre o valoare maximală; $f_i(X) \to \max$ - membrană complexă care integrează toate membranele de maximizare; $CE(X,Y)$ - membrană complexă care integrează toate membranele şi realizează modelul de calcul evolutiv în baza unui sistem de cromozomi.

Fiecare membrană elementară $f_i(X) \to \min/\max$ prezintă un procesor care rezolvă independent modelul matematic de minimizare sau maximizare a valorilor de stare X prin generarea vectorului de acţiune Y .

Modelul de calcul evolutiv $CE(X,Y)$ se bazează pe modele de algoritmi genetici cu codificare binară şi este prezentat în Ficura 2.

X^{GA}	x_1	x_2	x_3	…	x_k	x_{k+1}	x_{k+2}	…	x_N
X(T[0])	0	0	0		0	1	1		1
Y(T[0])	0	0	0		0	1	1		1
Y^{GA}	y_1	y_2	y_3	…	y_k	y_{k+1}	y_{k+2}	…	y_N

Fig. 2. Modelul de calcul evolutiv.

În Figura 2 sunt menţionate următoarele: X^{GA} - codificarea binară a genelor de stare identificate prin „0” logic dacă parametrul de stare respectiv scade la fiecare pas de comanda şi, „1” logic dacă parametrul de stare respectiv creşte la fiecare pas de comandă; $X(T[0])$ - starea iniţială a genelor; Y^{GA} - codificarea binară a genelor de acţiune, „0” – scăderea semnalului de comandă şi „1” – creşterea semnalului de comandă.

În tabelul de mai jos sunt prezentate combinațiile posibile pentru evaluarea sistemului.

(*T*)		(*T+1*)		**Explicații la modelul dinamic de evoluție a sistemului**
X^{GA}	**Y^{GA}**	**X^{GA}**	**Y^{GA}**	
				Pentru condiția $f_i(X) \rightarrow \min$
0	0	0	0	Condiție îndeplinită. Descreșterea semnalului de control duce la descreșterea parametrului de stare.
0	1	0	1	Condiție îndeplinită. Creșterea semnalului de control duce la descreșterea parametrului de stare.
1	0	0	1	Condiție ne-îndeplinită. Descreșterea semnalului de control duce la creșterea parametrului de stare. Încrucișarea genelor: Condiție planificată $\boldsymbol{Y}^{GA}(\boldsymbol{T}+1)=\overline{\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)\boldsymbol{XOR}\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)}$ Condiție așteptată $\boldsymbol{X}^{GA}(\boldsymbol{T}+1)=\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)\boldsymbol{XOR}\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)$.
1	1	0	0	Condiție ne-îndeplinită. Creșterea semnalului de control duce la creșterea parametrului de stare. Încrucișarea genelor: Condiție planificată $\boldsymbol{Y}^{GA}(\boldsymbol{T}+1)=\overline{\left(\boldsymbol{Y}^{GA}(\boldsymbol{T})\right)}$, Condiție așteptată $\boldsymbol{X}^{GA}(\boldsymbol{T}+1)=\overline{\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)}$.
				Pentru condiția $f_i(X) \rightarrow \max$
0	0	1	1	Condiție ne-îndeplinită. Descreșterea semnalului de control duce la descreșterea parametrului de stare. Încrucișarea genelor: Condiție planificată $\boldsymbol{Y}^{GA}(\boldsymbol{T}+1)=\overline{\left(\boldsymbol{Y}^{GA}(\boldsymbol{T})\right)}$, Condiției așteptată $\boldsymbol{X}^{GA}(\boldsymbol{T}+1)=\overline{\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)}$.
0	1	1	0	Condiție ne-îndeplinită. Creșterea semnalului de control duce la descreșterea parametrului de stare. Încrucișarea genelor: Condiție planificată $\boldsymbol{Y}^{GA}(\boldsymbol{T}+1)=\overline{\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)\boldsymbol{XOR}\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)}$, Condiție așteptată $\boldsymbol{X}^{GA}(\boldsymbol{T}+1)=\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)\boldsymbol{XOR}\left(\boldsymbol{X}^{GA}(\boldsymbol{T})\right)$.
1	0	1	0	Condiție îndeplinită. Descreșterea semnalului de control duce la creșterea parametrului de stare.
1	1	1	1	Condiție îndeplinită. Creșterea semnalului de control duce la creșterea parametrului de stare.

4. Structura sistemului de calcul multi-procesor

Structura sistemului de calcul multi-procesor este prezentată în Figura 3 și include $\boldsymbol{N}$ procesoare $\Pr_i, \forall \boldsymbol{i}=\overrightarrow{1,\boldsymbol{N}}$ conectate într-o rețea de comunicare.

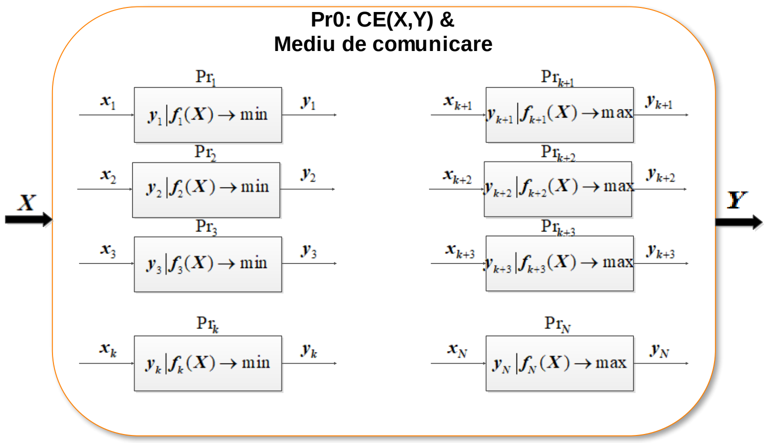

Fig. 3. Structura sistemului de calcul multi-procesor.

Modul de funcţionare.

Fiecare procesor îşi rezolvă sarcina sa, în mod concurent cu alte procesoare, citind datele de la intrare şi livrând rezultatul la ieşire. Schimbul de date este efectuat prin intermediul ***mediului de comunicare*** care livrează vectorul de stare $\boldsymbol{X}$ la toate procesoarele. Calculul condiţiei de convergenţă la valoarea condiţionată de modelul (2) şi calculul genetic este efectuat în baza procesorului $\Pr 0$, care şi determină modul de rutare în mediul de comunicare.

Menţiuni

Cercetările efectuate în această lucrare fac parte din tematica tezelor de doctorat planificate în cadrul Departamentului Informatica şi Ingineria Sistemelor, FCIM, UTM. Testarea experimentală şi funcţională s-a efectuat în cadrul ***ICG Engineering SRL, R. Moldova şi SCHUNK Electronic Solutions GmbH, Germania***.

Bibliografie

1. Dudgeon, D. and Mersereau R. *Multidimensional Digital Signal Processing*, Prentice-Hall, First Edition, 400 p. 1983, ISBN: 978-0136049593.
2. Yann Collette, Patrick Siarry. Multiobjective optimization. Principles and Case Studies. Springer-Verlag, 2003, 273 p., ISBN: 3-540-40182-2.
3. Аттетков А.В., Галкин С.В., Зарубин В.С. Методы оптимизации, Изд-во МГТУ им. Н.Э. Баумана, 2003, -440 с. ISBN: 5-7038-1770-6.
4. Rădulescu Iulia Cristina. Rezolvarea unor problem de optimizare multi-obiectiv bazată pe algoritmi evolutivi. Revista Română de Informatică şi Automatică, vol. 25, nr. 2, 2015, pp. 39-48.
5. Păun Gheorghe. Introduction to Membrane Computing. *Institute of Mathematics of the Romanian Academy*. 43p.
6. Taho Dorta, Jaime Jimenez, Jose Luis Martin, Unai Bidarte, and Armando Astarloa. Reconfigurable Multiprocessor Systems: A Review. International Journal of Reconfigurable Computing, Vol 2010, Art. ID 570279, 10p., DOI:10.1155/2010/570279.