

Rezultate experimentale pentru rezolvarea problemei echilibrării liniilor de asamblare utilizând algoritmi genetici

Prof.dr. Rodica MIHALCA, Catedra de Informatica Economica
Conf.dr. Carmen HARTULARI, Catedra de Cibernetica Economica, A.S.E. Bucuresti

În articol sunt prezentate o serie de rezultate ale autorilor, obținute în cadrul activității de cercetare efectuate la contractul cu A.N.S.T.I., având ca tema „Modelarea și simularea proceselor de producție folosind rețele neuronale și algoritmi genetici”. În cadrul cercetării efectuate au fost abordate mai multe tipuri de probleme de ordonantare și au fost elaborate o colecție de metode și modele cu utilizarea tehnicilor neuronale și genetice.

Cuvinte cheie: Echilibrare, linie de asamblare, ritmul liniei, cromozom, gena, timp de neutilizare, abatere de echilibrare.

Introducere

Abordarea prin metode evolutive a problemei de echilibrare a liniilor de asamblare constituie una din direcțiile de rezolvare a acestui tip de problema de ordonantare.

Problema echilibrării liniilor de asamblare constă în atribuirea setului de operații, care reprezintă procesul de montaj al unui produs, pe posturile de lucru ale liniei, respectând un set de restricții tehnologice și economice astfel încât să se minimizeze timpul de neutilizare a liniei și implicit numărul de posturi de lucru.

Pentru modelare se utilizează tehnici genetice. Obiectele identificate în proiectarea aplicației sunt:

Cromozomul, definit ca o soluție a algoritmului, reprezintă lista posturilor de lucru și a operațiilor asociate fiecărui post. Generarea unui cromozom presupune alocarea pentru fiecare stație a operațiilor care vor fi executate pe aceasta.

Gena, este componenta unui cromozon și definește o operație (activitate) alocată pe un post de lucru.

Populația reprezintă o mulțime de cromozomi de dimensiune fixă (cardinal dat apriori).

Pentru rezolvarea problemei se propune un algoritm de tip steady-state, care se desfășoară în următoarele etape:

- se generează un cromozom inițial pe baza ordonării pe niveluri a operațiilor și cu respectarea setului de restricții;

- se generează o populație inițială prin clonarea (copierea) și prin generarea aleatoare a cromozomului inițial;
- se evaluează fiecare cromozom al populației inițiale cu ajutorul funcției de evaluare FevalPop;
- cât timp numărul de generații nu depășește un prag dat sau funcția de evaluare nu atinge o anumită valoare, atunci :
 - se selectează cei mai buni cromozomi care vor fi părinți pentru noua populație;
 - se aplică mutații simple prin schimbarea poziției unei gene în cadrul aceluiași cromozom sau prin interschimbarea a două gene aflate pe poziții diferite cu respectarea setului de restricții;
- se adaugă la populația de la iterația precedentă populația temporară de indivizi nou creați
- se elimină cei mai slabi cromozomi, utilizând în acest scop funcția de evaluare, astfel încât să se ajungă din nou la dimensiunea inițială

Notatii și parametri de testare:

- ritmul liniei;
- durata activităților;
- dispersia duratelor activităților;
- structura grafului (liniara, cu subasamblări paralele);
- complexitatea grafului funcție de numărul de relații de precedentă existente;
- numărul maxim admis de posturi de lucru ;

- numărul maxim de generari;
- numărul procentual de cromozomi.

Rezultatele obtinute sunt:

- numărul de posturi de lucru;
- numărul total de generari;
- abaterea de echilibrare;
- indicile de uniformitate;
- numărul de mutatii efectuate.

Modul de efectuare al testelor

Algoritmul prezentat a stat la baza realizării unui produs program utilizând o metoda de proiectare orientata obiect (OMT - Object Modeling Technique), iar implementarea s-a facut în limbajul C++, cu interfata în Visual C.

S-au efectuat teste asupra unor probleme variate ca dimensiune (12, 25 si 78 de activitati) si ca dispersie a duratelor. Pentru aceeasi definitie de problema (continând durate si relatii de precedenta) s-a rulat algoritmul genetic pentru mai multe valori ale ritmului. În cadrul aceleiasi instante de problema s-au efectuat rulari experimentale cu varierea parametrilor algoritmului genetic, dupa cum urmeaza:

Parametri constanti:

Dimensiunea populatiei: 200 indivizi
 Numar de generatii (iteratii): 300

Parametri variabili:

- La variatia procentului de crestere a populatiei s-au considerat trei variante, respectiv 2%, 4% si 6%, mentinând constanta probabilitatea de mutatie 70%
- La variatia probabilitatii de mutatie s-au considerat trei variante, respectiv 50%, 60%, 70%, mentinând constant procentul de crestere a populatiei 4%

Rezultatele testelor si concluzii

Datele luate în considerare la test au avut în vedere mai multe criterii:

- dimensiunea problemei;
- ritmul liniei raportat la duratele activitatilor;
- dispersia duratelor activitatilor;
- structura grafului în doua acceptiuni: de arbore si de graf orientat si în acest caz, s-a avut în vedere pentru fiecare caz în parte complexitatea grafului, adica numărul de legaturi existente între activitatile din graf;
- procentul de mutatii utilizate în algoritmul genetic si procentul de crestere a populatiei, referitor la efectul acestuia la solutia obtinuta;
- calitatea solutiei obtinute a fost apreciata prin trei indicatori: abaterea de echilibrare, indicele de uniformitate si numărul de statii obtinute

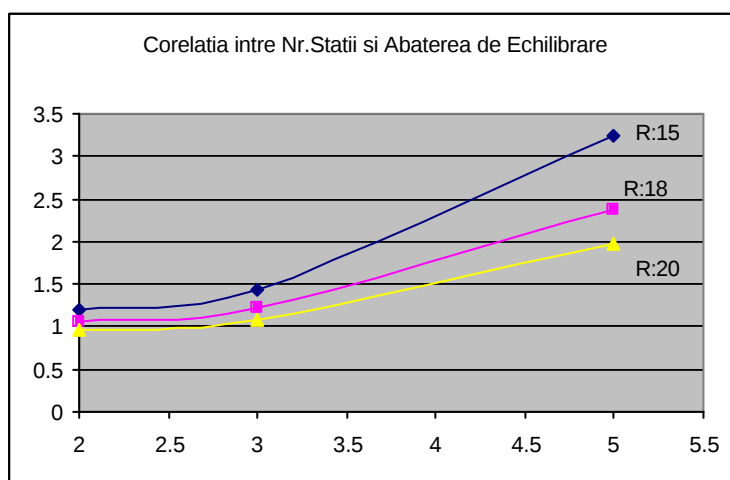


Figura 1

Din experimentarile facute se constata ca pentru probleme de dimensiuni mici (de exemplu. 12 job-uri) si pentru acelasi ritm, cu durate diferite ale operatiilor, pastrând aceeași structura a grafului, cu cât duratele sunt mai mici în raport cu ritmul liniei si dispersia duratelor are o valoare mai mica, obtinem o mai buna încarcare a liniei, adica o abatere de echilibrare cât mai mica si un indice de uniformitate foarte bun, adica o valoare mica a acestuia. (figura 1). Pentru aceleasi probleme, dar cu o dispersie mai mare a duratelor activitatilor, rezultatele sunt mai slabe, obtinând o abatere de echilibrare mai mare si un indice de uniformitate mai slab. Pe masura ce ritmul liniei creste, cei trei indicatori ai liniei (ce caracterizeaza calitatea solutie) au valori

din ce în ce mai mici. Acest lucru este explicabil, întrucât creste numarul de combinatii posibile ce pot fi luate în considerare. Un al doilea grup de teste se refera la procentul de cresteri ale populatiei pentru aceeasi problema, cu un coeficient de crestere între 2% si 6 %. Din experimentari se obtin valori mai proaste ale solutiei atunci când acest procent este mic (2%) sau mare (6%). Pentru valori medii (4%) se obtine cea mai buna solutie, indiferent de dispersia datelor si structura grafului. Se poate puncta ca transformarea grafului din arbore în graf orientat conduce la înrautătirea solutiei, lucru explicabil prin cresterea numarului de restrictii si, implicit, a scaderii numarului de combinatii fezabile. (figura. 2)

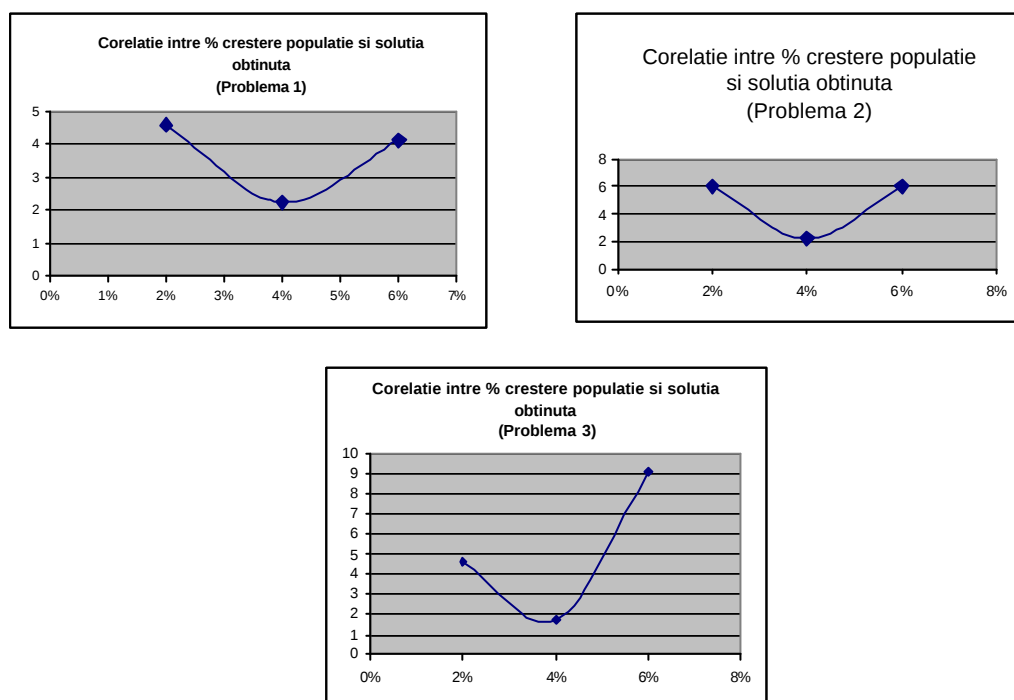


Figura 2.2

Se poate concluziona, daca avem în vedere si timpul de calcul pentru obtinerea solutiei, ca durata sa creste vertiginos pe ma-

sura ce creste dimensiunea problemei si numarul de restrictii de precedenta din graf pe de alta parte (figura. 3).

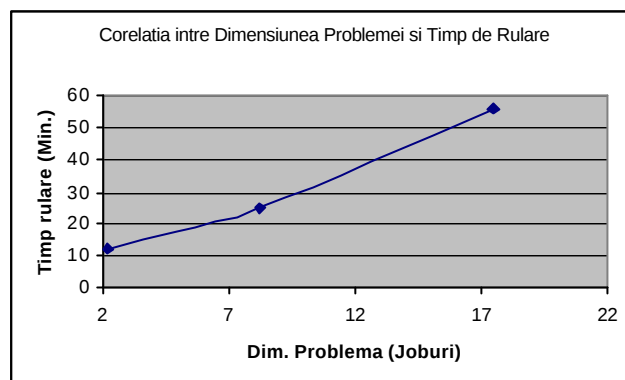


Figura 3

Algoritmul își arată eficiența mai ales în cazul problemelor de dimensiuni mici și medii.

Bibliografie

[1] O. Brudaru, C. Rusu – Proiectarea unor linii de fabricație flexibile, Ed. Tehnica, București, 1990

[2] * * * - Modelarea și simularea procesului de producție folosind rețele neuronale și algoritmi genetici, 5. Implementarea și testarea componentelor colecției, Raport de cercetare la contractul nr. 198/1996 MCT.

[3]*** - Evolutionary Computation, MIT Journal (1993-1999)

[4]*** - Genetic and Evolutionary Computation Conference, July 1999, Orlando, Florida, USA

[5] Mihalca R Object oriented modeling for the assembly line balancing problem, Informations Technology, Ed INFOREC, 1999, Bucharest