

Experiente cu un algoritm evolutiv pentru rezolvarea problemelor generale de ordonantare

Prof.dr. Csaba FABIAN

Catedra de Informatica Economica, A.S.E. Bucuresti

Se raporteaza rezultatele experientelor efectuate cu un algoritm evolutiv rafinat (C-GA) pentru rezolvarea problemelor de ordonantare generala. Cercetarile au fost efectuate in cadrul unui contract de cercetare cu Ministerul Cercetarii si Tehnologie cu tema "Modelarea si simularea procesului de productie folosind retele neuronale si algoritmi genetici" [1]. În cadrul cercetarii au fost atacate mai multe tipuri de probleme de ordonantare si a fost elaborata o colectie de metode si modele bazate pe retele neuronale si genetice. Ulterior ele au fost implementate si au fost efectuate experiente cu probleme din literatura [7].

Cuvinte cheie: experiente, job shop scheduling, ordonantare generala, algoritmi evolutivi.

1. Introducere

Problema de Ordonantare Generala (POG, în engleza Jobshop Scheduling Problem, JSSP) consta din n lucrari (job-uri) care se prelucreaza pe m masini. Se cunosc timpi de prelucrare si ordinea tehnologica de prelucrare a lucrarilor pe masini. Se cere o ordonantare a lucrarilor pe m masini astfel încât timpul total de prelucrare sa fie cât mai mic .

S-GA este algoritmul care, utilizând mutații simple evolutive care a fost implementat si rezultatele caruia au fost experimentate în [3].

În continuare s-a rafinat si s-a perfectionat algoritmul S-GA, care a devenit C-GA (Genetic Algorithms cu Crossover mutations), algoritmul evolutiv bazat pe încrucisari de diferite tipuri [2].

În prezentul articol se raporteaza experientarile efectuate cu algoritmul C-GA

S-au comparat algoritmi genetici simpli cu varianta "stady state", s-au studiat calitatea diferitelor scalari, si diferite metode de selectie a parintilor din populatie. De asemenea s-au studiat efectul numarului de indivizi schimbati la o generatie, procentul de mutatii simple si variante de încrucisari implementate. Rezultatele principale sunt reprezentate grafic

prin diagrame si se interpreteaza fiecare rezultat deosebit.

2. Probleme de test utilizate

Problemele de test utilizate sunt acelasi ca si in cazul algoritmului S-GA obtinute de pe internet de la adresa *anonymous ftp* la ***mscmga.ms.ic.ac.uk*** sau numeric 155.198.66.4, sau pagina web: ***http://mscmga.ms.ic.ac.uk***. Parametrii generali ai probleme-lor se pot extrage din tabela 1. S-au rezolvat 37 de probleme selectate din 79 probleme prezentate în literatura de specialitate specificata.

Numarul de joburi variaza între 10 si 50, numarul masinilor între 5 si 15, complexitatea problemelor variaza între 1 si 5, iar media duratelor operatiilor este între 45 si 60. Daca se acopera toti patru parametri de care depind problemele de test cu cel puțin trei valori avem $3 \times 3 \times 3 \times 3 = 81$ tipuri de probleme iar daca din fiecare tip se considera 5 instante avem 405 probleme distincte de considerat. Se poate observa ca numarul mediu de joburi este 15,53, numarul mediu de masini este 8,24, deci avem în medie de 2 ori mai multe joburi decât masini, valoarea functiilor scop au o medie de 1006,74 iar complexitatea medie a problemelor este 2,27.

3. Notatii si parametrii de testare

Vom folosi urmatoarele notatii:

N - numarul joburilor; Val - valoarea cea mai buna cunoscuta a functiei scop; Val1 - valoarea functiei scop obtinuta cu algo-ritmul S-GA; Val2 - valoarea functiei scop obtinuta cu algoritmul C-GA; %MUT - % de mutatie simpla; RANK (R) Selectia **rank** a indivizilor din populatie; WHEELS (W) Selectia **roulette wheels**; TURN (T) Selectia **Tournament**; DETER(D) Selectia **Deterministic sampling**; STOCH (S) Selectia **Stochastic sampling**; UNIF (U) Selectia **Uniform sampling**; GENER Nu-mar de generari; NON Fara scalare; LIN Scalare cu functii liniare; SIGMA Scalare cu functia sigma; POWER Scalare **Power law**; UC Încriscari **Uniform Crossover**; OE Încriscari **Odd – Even**; OP Încriscari **One Point**; TP Încriscari **Two Points**; PM Încriscari **Partial Matching**; COMPL Complexitatea problemelor calculata dupa metoda din [4]; TIMP - Timpul de calcul în secunde necesara pentru rezolvarea problemei; SEL - Metoda de selectie a parintilor; %D1D2 - Diferenta procentuala între %Del1 si %Del2.

Algoritmul C-GA depinde de urmatoarele parametrii:

- Metode de scalare (5 variante)
- Metode de selectie (5 variante)
- Metode de încriscari (6 variante)
- Marimea populatiei (cel puțin 3 variante)
- Numarul generarilor (cel puțin 3 variante)
- Numarul indivizilor încriscati din populatie (Cel puțin 3 variante)

În total de varinate de executat in cazul de **steady state** sunt $5 \times 5 \times 6 \times 3 \times 3 \times 3 \times 3 = 12150$ de variante de algoritmi pentru 405 probleme înseamnă peste 12000000 de probleme rezolvate. Evident ca s-a facut o reducere substantiala a numarului problemelor de test

pentru ca testarea sa poata fi realizata în timp util.

Algoritmul C-GA are urmatoari parametrii fixati pentru rulara problemelor de test:

- Populatia de 1000 de indivizi
- Numarul generarilor este fixat la 10000
- Schema de scalare este liniara
- Varianta *stady state* este cu 1000 de indivizi înlocuiti
- Probabilitatea de încriscare este de 100% cu metoda *roulette wheel*.

S-au efectuat si teste cu parametri modificali, ceace se va indica în fiecare caz în parte.

4. Rezultate de teste cu C-GA

În Tabela 1 sunt trecute si principalele rezultate obtinute cu S-GA si cu C-GA pentru a efectua o comparatie între ele:

Problemele sunt grupate în mai multe categorii. Astfel:

- 1) %Del1 = 0 (Procente de deviere la S-GA=0, solutii optime obtinute)
- 2) %Del1 < 10%
- 3) %Del1 ~ 20%
- 4) %Del1 ~ 40%
- 5) %Del1 > 50%

Se studiaza dependentia %Del2 de %Del1 si se constata:

- Daca %Del1 = 0 sau %Del1 < 10% avem %Del2 = 0 (s-a obtinut solutia cea mai buna cunoscuta daca %Del1 < 10%)
 - Daca %Del1 ~ 20% avem %Del2 ~ 2% (Reducere de 10 ori a %Del2 fata de %Del1)
 - Daca %Del1 ~ 40% avem %Del2 ~ 5% (Reducere de 8 ori a %Del2 fata de %Del1)
 - Daca %Del1 > 50% avem %Del2 ~ 40% (Reducere de 1,5 ori a %Del2 fata de %Del1)
- Deci cu cât %Del1 este mai mare %Del2 se reduce mai puțin..

De aici se poate vedea ca varianta C-GA, rafinarea si perfectionarea variantei S-GA este mult mai eficienta. Pe exemplul de test **la18** s-a studiat efectul % de mutatii la valoarea functiei scop obtinuta. S-a constatat ca cea

mai buna valoare este pe la 2-3 % (vezi figura 1)

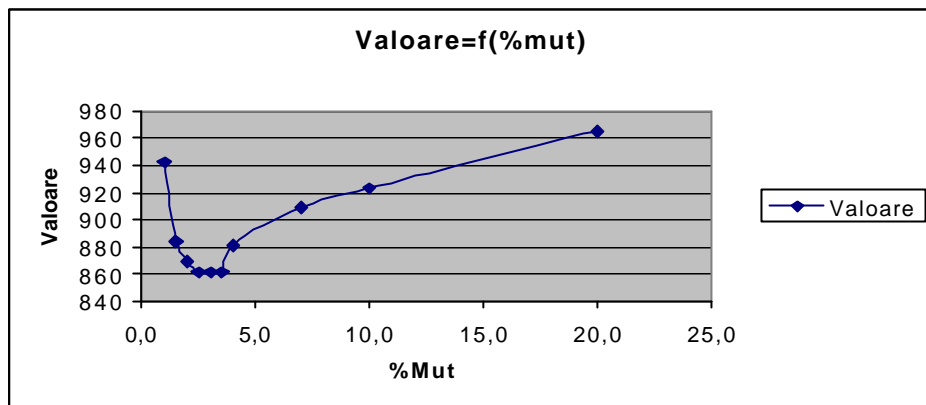


Fig. 1. Efectul %MUT asupra Valoare (Val2)

Numarul joburilor (n) influenteaza diferenta %Del2 si complexitatea problemelor influenteaza diferenta %Del2 în mod semnificativ. În continuare s-au studiat efectul proportiei pastrate din populatia de la o generare la alta asupra timpului de calcul, pentru atingerea aceleiasi valori a functiei scop. S-a constatat ca, o data cu înlocuirea în proportie mai mare a indivizilor din populatie creste timpul necesar pentru rezolvarea problemei, legatura fiind aproa-pe liniara.

S-a mai studiat efectul combinarii a doua metode de selectie a indivizilor din populatie, pentru mutatie si/sau încrucisare. S-a constatat ca combinarea metodelor roulette wheels (W) si selectia pur aleatore (UC) este %Del2 cea mai mica la aplicarea metodelor pure adica la 0% si 100%. Acelasi lucru se poate observa la combinare în diferite procente (0%, 25%,50%, 75% si 100%) ale metodelor OE cu TP, UC cu OE, UC cu TP si OP cu UC (figura 2).

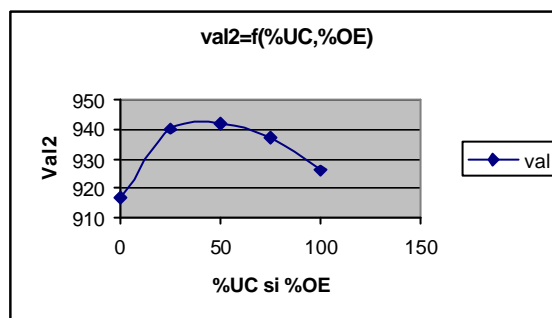


Fig. 2. Efectul % UC si OE asupra Val2
 În continuare s-au reprezentat influenta scalarii si a selectiei asupra valorii obtinute. Se constata în ambele situatii o influenta redusa a scalarii si a metodei de selectie.(figura 3)

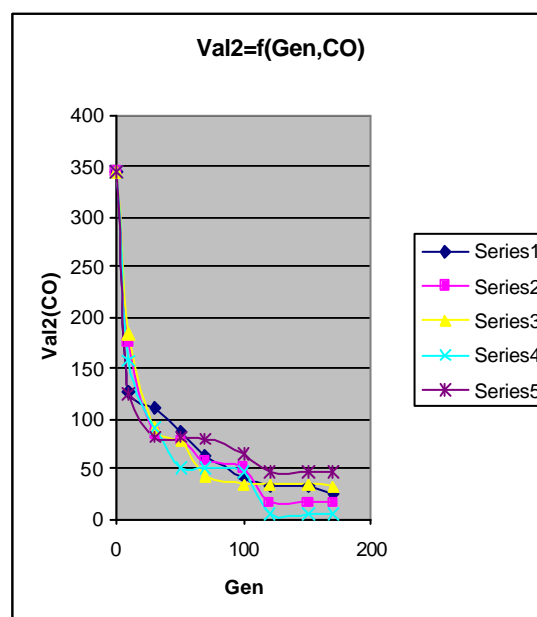


Fig. 3. Influenta metodelor de incrucisare

Totusi este posibila o clasificare a metode-lor de scalare. Astfel cele mai bune rezultate se obtin în cazul liniar (LIN) si cu metoda sigma (SIGMA), cea mai proasta metoda de scalare este NON adica când nu se foloseste nici o metoda de scalare.

Metoda "roulette wheel" (W) de selectie a parintilor este cea mai buna, urmata de metoda "tournamnet"; cea mai proasta metoda de selectie s-a dovedit a fi "deterministic sampling".

La studiul metodelor de încrucisare au fost evidentiata metodele "Two Points (TP)" si metoda "Odd-Even (OE)", cea mai proasta fiind "Partial matching (PM)".

Pe figura 3 se poate observa o convergenta rapida (exponentiala inversa) a variantei C-GA cea ce este o calitate pozitiva a lor.

5. Comparatii S-GA cu C-GA

Între %Del1 si %Del2 se pastreaza o diferenta constanta în functie de numarul joburilor, cea ce se poate vedea din graficul din figura 4.

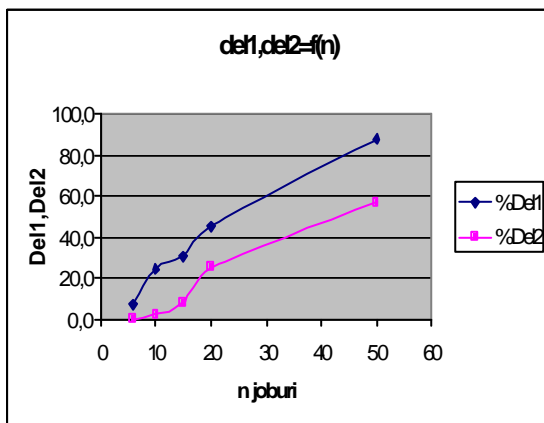


Fig. 4. Influenta numarului de joburi asupra Del1 si Del2

Reprezentarea informatiilor obtinute utilizând valorile medii este mai ilustrativa decât valorile individuale, care au o dispersie si perturba

reprezentarile legi-tatilor observate din experimentari.

5. Concluzii

Concluziile cele mai importante obtinute în urma celor doua experimentari sunt:

Experimentari cu încrucisari:

-Solutii mai bune s-au obtinut cu încrucisari decât cu mutatii

-Îmbunatatirile folosind încrucisari au fost mai bune la probleme la care mutatiile simple au dat rezultate mai bune

-Cu cât %Del1 este mai mare, cu atât %Del2 este mai mare

-procentul de mutatii este minim la 2-3%

-%Del2 creste cu numarul joburilor

-%Del2 creste cu complexitatea problemelor din litratura

-Timpul de calcul creste cu numarul indivizilor pastrati în populatie

-Reducerea numarului indivizilor din populatie nu afecteaza esential calitatea solutiei

-Folosirea valorilor medii la toate legaturile studiate, este mai ilustrativa decât valorile individuale prezentate.

Recomandari în privinta caracteristicilor problemelor:

-Problemele sa fie de dimensiuni maxime de 15-20 joburi cu maxim 20 masini

-Complexitatea problemelor sa fie între 1 si 2.

Recomandari în privinta parametrilor algoritmului C-GA:

-Sa se foloseasca varianta stady state al lui C-GA

-Sa se foloseasca scalare liniara

-Sa se foloseasca selectia parintilor dupa metoda roulette wheel

-Sa se foloseasca încrucisari de tip two points

-Populatia sa fie fixata la 500 de indivizi

-Numarul generarilor sa fie aproximativ $(n*m)*(n*m)/10$

Bibliografie

[1] Fabian, Cs., Mihalca R., Nica V., *Modelarea si simularea procesului de productie folosind retele neuronele si algoritmi genetici*, contract cu MCT, Bucuresti, 1996-1999

[2] Fabian, Cs., *Evolutionary Job Shop Scheduling*, Computer Scinces, Ed. INFO-REC, 1997, Bucharest

[3] Fabian Cs., Simion F., *Experinces with Evolutive Mutations for Job Shop Problems*, Information Technology, Ed. INFOREC, 1999, Bucharest

[4] Fabian Cs. *O masura a complexitatii pentru problema de ordonantare generala*, SCCECE nr.4, 1999

[5] Mattfeld.,D.C., *Evolutionary Search and Job Shop Problems. Investigations on Genetic Algorithms for Production Scheduling*, Physica Verlag, 1996, Heidelberg

[6] Michalewicz, Z., *Genetic Algorithms +Data Structure*, Springer Verlag, 1992, Berlin

[7] *** Imperial College, Management School : <http://mscmga.ms.ic.ac.uk>

Tabela 1

Nr. Crt	Nume	Job	Mas.	Valoare	Compl.	Val1	Val2	%Del1	%Del2	%D1D2
1	la05	10	5	593	1,80	593	593	0,0	0,0	0,0
2	la10	15	5	958	1,55	958	958	0,0	0,0	0,0
3	la14	20	5	1292	1,71	1292	1292	0,0	0,0	0,0
2)	%Del1 <10%							0,0	0,0	0,0
Nr. Crt	Nume	Job	Mas.	Valoare	Compl.	Val1	Val2	%Del1	%Del2	%D1D2
1	ft6	6	6	55	1,61	59	55	7,3	0,0	7,3
2	la01	10	5	666	0,82	710	666	6,6	0,0	6,6
3	la06	15	5	926	1,55	940	926	1,5	0,0	1,5
4	la09	15	5	951	1,54	995	952	4,6	0,1	4,5
5	la11	20	5	1222	1,48	1333	1222	9,1	0,0	9,1
6	la12	20	5	1039	1,68	1096	1039	5,5	0,0	5,5
7	la13	20	5	1150	1,46	1211	1150	5,3	0,0	5,3
3)	%Del1 ~20%							5,7	0,0	5,7
Nr. Crt	Nume	Job	Mas.	Valoare	Compl.	Val1	Val2	%Del1	%Del2	%D1D2
1	la02	10	5	655	1,56	780	675	19,1	3,1	16,0
2	la04	10	5	590	1,72	705	593	19,5	0,5	19,0
3	la07	10	5	890	1,47	1062	890	19,3	0,0	19,3
4	la15	20	5	1207	1,24	1443	1240	19,6	2,7	16,8
5	la18	10	10	848	2,96	1050	861	23,8	1,5	22,3
6	abz-5	10	10	1234	2,96	1476	1260	19,6	2,1	17,5
7	abz-6	10	10	943	3,76	1143	945	21,2	0,2	21,0
8	la16	10	10	945	3,06	1153	982	22,0	3,9	18,1
9	la17	10	10	784	2,98	949	793	21,0	1,1	19,9
10	la20	10	10	902	2,74	1091	942	21,0	4,4	16,5
4)	%Del1 ~40%							20,6	2,0	18,6
Nr. Crt	Nume	Job	Mas.	Valoare	Compl.	Val1	Val2	%Del1	%Del2	%D1D2
1	ft10	10	10	930	1,66	1256	951	35,1	2,3	32,8
2	ft20	20	5	1165	0,76	1665	1256	42,9	7,8	35,1
3	la36	15	15	1268	4,68	1700	1451	34,1	14,4	19,7
4	la37	15	15	1397	4,91	1946	1643	39,2	17,6	21,6
5	la38	15	15	1196	4,88	1676	1441	40,1	20,5	19,6
6	la39	15	15	1233	4,83	1743	1437	41,4	16,5	24,8

7	la40	15	15	1222	4,74	1743	1440	42,6	17,1	23,6
8	orb01	10	10	1059	1,34	1513	1148	42,9	8,4	34,5
9	orb05	10	10	887	3,34	1268	922	43,0	3,9	39,0
10	orb06	10	10	1010	1,34	1321	1056	30,8	4,6	26,2
11	la28	20	10	1216	3,34	1708	1420	40,5	16,8	23,7
5)	%Del1>50%							39,9	9,4	30,5
Nr. Crt	Nume	Job	Mas.	Valoare	Compl.	Val1	Val2	%Del1	%Del2	%D1D2
1	abz-7	20	15	656	2,96	999	858	52,3	30,8	21,5
2	abz-8	20	15	669	5,14	1046	922	56,4	37,8	18,5
3	abz-9	20	15	679	4,88	1027	918	51,3	35,2	16,1
4	swv01	20	10	1418	1,51	2391	1947	68,6	37,3	31,3
5	swv07	20	15	1620	2,54	2789	2361	72,2	45,7	26,4
6	swv15	50	10	2940	1,48	5527	4602	88,0	56,5	31,5
								64,8	40,6	24,2