INFOTEH-JAHORINA Vol. 16, March 2017.

Rješavanje problema minimalnog pokrivanja lokacija primjenom različitih operatora selekcije genetskog algoritma

Jovana Janković
Filozofski fakultet Univerziteta u Istočnom Sarajevu
Pale, RS, BiH
jovanajankovic91@yahoo.com

Sadržaj— **Minimizacija broja pokrivenih lokacija nepoželjnim objektima predstavlja problem kombinatorne optimizacije od velikog praktičnog značaja. U ovom radu analizira se rješavanje problema minimalnog pokrivanja lokacija sa dozvoljenim višestrukim pokrivanjem primjenom genetskog algoritma. Predstavljeni su rezultati uporedne analize uticaja različitih tipova operatora selekcije na efikasnost pronalaženja dobrih rješenja za prostore koji obuhvataju 100, 400 i 500 lokacija. Na osnovu rezultata numeričkih testova utvrđeni su adekvatni tipovi operatora selekcije sa aspekta kvaliteta rješenja i vremena izvršavanja algoritma.**

Ključne riječi- genetski algoritam; operator selekcije; lokacijski problemi; problem minimalnog pokrivanja lokacija

I. UVOD

Posebnu klasu lokacijskih problema kombinatorne optimizacije predstavljaju problemi pokrivanja lokacija (Covering Location Problem - CLP). Njihov zadatak je pronalaženje optimalne pozicije objekata na datom skupu lokacija, tako da pokrivenost lokacija bude maksimalna ili minimalna [1].

Osnovni zadatak problema minimalnog pokrivanja lokacija (Minimal Covering Location Problem - MinCLP) je pronalaženje optimalnih pozicija za fiksan broj nepoželjnih objekata (deponije smeća, nuklearne elektrane, zatvori, itd), tako da oni pokrivaju što manji broj lokacija na nekom prostoru. U radu [2], opisano je nekoliko analitičkih modela za postavljanje neželjenih objekata. Problem minimalnog pokrivanja lokacija na ravni razmatran je u radu [3]. Berman i saradnici su, u [4], proučavali problem na mreži i predstavili algoritam koji ga rješava. U radu [5] opisano je pet modela problema minimalnog pokrivanja lokacija sa definisanjem parametra udaljenosti između objekata. Autori rada [6], predstavili su dva matematička modela MinCLP-a: model problema minimalnog pokrivanja lokacija sa jednostrukim pokrivanjem (Minimal Covering Location problem with Single Coverage MinCLP-SC) i problema minimalnog pokrivanja lokacija sa višestrukim pokrivanjem (Minimal Covering Location Problem with Multiple Coverage – MinCLP-MC). Glavni nedostatak MinCLP-SC je taj što uslov jednostrukog pokrivanja znatno smanjuje skup mogućih rješenja. Kod praktičnih problema često se zahtijeva da se svi objekti postave u prostoru nezavisno od toga da li su neke lokacije višestruko pokrivene [7].

Metaheurističke ili moderne heurističke metode, se uz određene modifikacije mogu primjenjivati za rješavanje širokog skupa problema kombinatorne optimizacije, [8-10]. Najpoznatije metaheuristike su genetski algoritmi (Genetic Algorithms - GA), simulirano kaljenje (Simulated Annealing - SA), tabu pretraga (Tabu Search - TS), metoda promjenljivih okolina (Variable Neighbourhood Search - VNS), itd. MinCLP-MC je veoma pogodan za primjenu genetskog algoritma, čija je osnovna prednost pronalaženje kvalitetnih rješenja za relativno kratko vrijeme. Međutim, kao i kod svih metaheurističkih metoda, ne postoji čvrsta garancija da su dobijena rješenja dopustiva, optimalna, ili čak bliska tačnom rješenju. Iz prethodnog razloga neophodno je dobro prilagođenje genetskog algoritma problemu MinCLP-MC, gdje je izbor operatora selekcije od posebne važnosti, [11-13].

Ovaj rad je organizovan na sljedeći način: u drugom poglavlju su opisani matematički modeli za MinCLP-SC i MinCLP-MC, sadržaj trećeg poglavlja obuhvata implementaciju GA-a za rješavanje MinCLP-MC i predložene operatore selekcije, u četvrtom poglavlju prikazani su rezultati rješavanja MinCLP-MC primjenom GA i data je analiza uticaja operatora selekcije na performanse GA-a.

II. MATEMATIČKI MODELI PROBLEMA MINIMALNOG POKRIVANJA LOKACIJA

A. Problem minimalnog pokrivanja lokacija sa jednostrukim pokrivanjem

Parametri matematičkog modela MinCLP-SC su:

I – skup lokacija,
J – skup objekata,

$$x_j = \begin{cases} 1, & \text{ako je objekat postavljen u lokaciju j} \\ 0, & \text{inače} \end{cases},$$

y_i – stepen pokrivenosti lokacije i,
t_{ij} – udaljenost između lokacija i i j,
S – radijus pokrivenosti,
P – broj objekata,
$N_i = \{j | t_{ij} \le S\}$ – skup svih objekata j koji pokrivaju lokaciju i.

Formulacija matematičkog modela je sadržana u narednoj relaciji:

$$\min \sum_{i \in I} y_i, \tag{1}$$

uz uslove:

$$\sum_{j \in N_i} x_j \le y_i, \quad \forall i \in I, \tag{2}$$

$$\sum_{j \in J} x_j = P, \tag{3}$$

$$x_j \in \{0,1\}, \quad \forall j \in J, \tag{4}$$

$$y_i \in \{0,1\}, \quad \forall i \in I. \tag{5}$$

B. Problem minimalnog pokrivanja lokacija sa višestrukim pokrivanjem

Da bi se omogućilo višestruko pokrivanje lokacija, potrebno je ulov (2) iz prethodnog modela zamijeniti sljedećim uslovom, [6]:

$$\min(1, \sum_{j \in J} x_j) \le y_i, \quad \forall i \in I. \tag{6}$$

Analizirani model dozvoljava da objekti budu proizvoljno udaljeni jedni od drugih, te postoji mogućnost da u optimalnom rješenju svi objekti budu smješteni neposredno jedan do drugog, što nije poželjno u slučaju zagađivača. Uvođenjem parametra D, koji predstavlja minimalnu udaljenost između objekata, i definisanjem uslova (7) i (8) dobija se uopštenje modela MinCLP-MC.

$$t_{j_1 j_2} \ge D, \quad \forall j_1, j_2 \in J, \tag{7}$$

$$j_1 < j_2 \quad \wedge \quad x_{j_1} \cdot x_{j_2} = 1. \tag{8}$$

III. Implementacija genetskog algoritma

Implementacija GA je izvršena u razvojnom okruženju *Visual Studio 2010,* korišćenjem programskog jezika *C#.* Testiranja su vršena na računaru sa procesorom *Intel Core i3*, 2.40 GHz i 4 GB radne memorije. Problemi su rješavani na instancama dimenzije 100, 400 i 500 lokacija, sa 10, 15 i 20 objekata, radijusa pokrivenosti 2, 3 i 4 r.j. Međusobna udaljenost između lokacija u matematičkom smislu predstavlja euklidsku udaljenost. Koordinate lokacija su proizvoljno generisane na mreži 30x30 r.j.

Ulazni parametri GA su: dimenzija problema n, udaljenosti između lokacija zadane preko matrice udaljenosti, radijus pokrivenosti S. Genetski kod jedinke sastoji se od n bita, pri čemu svaki bit odgovara jednoj lokaciji. Jedinica na poziciji i označava da je na i-toj lokaciji postavljen objekat. U slučaju da jedinka sadrži više ili manje jedinica od broja objekata P, višak (ili manjak) jednica se uklanja (ili dodaje) slučajnim izborom. Početna populacija je generisana proizvoljno. Broj jedinki u populaciji je 100. Kriterijum zaustavljanja je definisan slučajem da je dostignut maksimalan broj generacija (ograničen na 1000). Svaka instanca je rješavana 10 puta sa različitim vrijednostima promjenljivih (seed).

Operator selekcije se u biološki inspirisanoj literaturi naziva još i operator odabiranja. To je postupak odlučivanja koje će jedinke preživjeti i prenijeti svoj genetski materijal na naredne generacije, a koje će nestati [13]. Ovaj operator primjenjuje se u skladu sa vrijednostima funkcije prilagođenosti. Karakteristike predloženog GA su vrednovane za četiri različita tipa operatora selekcije: selekcija skraćivanja (truncation selection - TRS), turnirska selekcija (tournament selection - TOS), rulet-točak selekcija (roulette-wheel selection - RWS) i stohastička univerzalna uzorkovanja (stochastic universal sampling- SUS).

Operator ukrštanja se najčešće definiše proizvoljnim brojem prekidnih tačaka jedinki roditelja. U zavisnosti od izbora tih tačaka moguća su sljedeća ukrštanja: jednopoziciono, dvopoziciono, uniformno, itd. Kod implementiranog GA upotrebljen je operator dvopozicionog ukrštanja sa nivom ukrštanja 0.8. Na slučajan način se biraju dvije tačke prekida od kojih se vrši raspodjela genetskog materijala sa roditelja na potomke.

Mutacija omogućava vraćanje korisnog genetskog materijala koji može biti potencijalno izgubljen pri selekciji i ukrštanju. Dobar je mehanizam za izbjegavanje lokalnih ekstremuma prilikom pretrage prostora potencijalnih rješenja. Kao dio predloženog GA, primjenjena je prosta mutacija nivoa p_m= 0.001.

A. Operatori selekcije genetskog algoritma

Selekcija skraćivanja (TRS). U slučaju selekcije skraćivanja potencijalna rješenja su sortirana prema svom fitnesu. Potom se bira procenat p $(10\% \le p \le 50\%)$ najsposobnijih jedinki koje će se ukrštati.

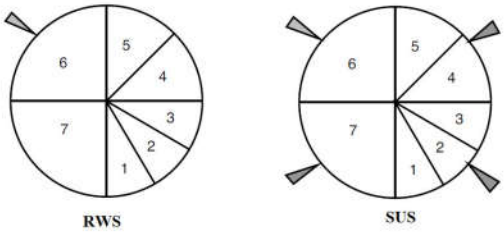
Turnirska selekcija (TOS). TOS je selekcija zasnovana na pojmu ranga. Na slučajan način se izabere m jedinki. Parametar m predstavlja dimenziju turnira, tj. broj takmičara u grupi. Ovaj parametar se najčešće unaprijed zadaje. Turnir se, potom, primjeni na m članova grupe kako bi se izabrao najbolji član, pobjednik. Ako je potrebno turnirskom selekcijom odabrati k jedinki, onda se ovaj postupak ponavlja k puta.

Rulet-točak selekcija (RWS). Osnovna karakteristika ove selekcije je da se svakoj jedinki i trenutne populacije dodijeli vjerovatnoća $p(i)$ koja je proporcionalna fitnesu jedinke $f(i)$:

$$p(i) = \frac{f(i)}{\sum_{j=1}^{N} f(j)} \tag{9}$$

gdje je N veličina populacije. Za objašnjavanje ove selekcije koristi se pojam ruleta. Jedinke su postavljene na točku, pri čemu veće dijelove točka zauzimaju jedinke sa boljim fitnesom. Izbor k jedinki ostvaruje se okretanjem točka k puta. Nakon jednog obrtaja rulet-točka, izabere se tačno jedna jedinka pomoću pokazivača na točku. Pošto jedinke sa boljim fitnesom zauzimaju više prostora na točku, veća je vjerovatnoća da jedinke bolje prilagođenosti budu izabrane za proces ukrštanja. Nedostatak ove selekcije je prerana konvergencija genetskog algoritma jer u većini slučajeve bolje jedinke pobijede konkurenciju i bude izabrane kao roditelji.

Stohastička univerzalna uzorkovanja (SUS). Ova selekcija je varijanta prethodne. Nastala je u cilju smanjenja rizika od preranе konvergencije. Na obodu točka postavljeno je k ravnomjerno raspoređenih pokazivača. Stoga je dovoljno točak okrenuti samo jednom da bi se dobilo potrebnih k jedinki.



Slika 1. Rulet-točak selekcija i stohastička univerzalna uzorkovanja sa 7 jedinki [12]

IV. Rješavanje MinCLP-MC genetskim algoritmom

Za predstavljeni model problema minimalnog pokrivanja lokacija sa višestrukim pokrivanjem riješene su odabrane instance genetskim algoritmom. Kvalitet rješenja je mjeren srednjom relativnom greškom (oznake *agap*) koja pokazuje odstupanje dobijenog rješenja od optimalnog:

$$agap = \frac{1}{N}\sum_{i=1}^{N} gap_i, \tag{10}$$

gdje je N broj izvršavanja metaheuristike, dok gap_i predstavlja individualnu relativnu grešku ili odstupanje i-tog rješenja sol_i od optimalnog rješenja opt_{sol}, prema sljedećem izrazu:

$$gap_i = 100 \cdot \frac{|\, sol_i - opt_{sol} \,|}{opt_{sol}}. \tag{11}$$

U slučaju da optimalno rješenje opt_{sol} za datu instancu nije poznato, individualna relativna greška se računa u odnosu na najbolje poznato rješenje $best_{sol}$:

$$gap_i = 100 \cdot \frac{|\, sol_i - best_{sol} \,|}{best_{sol}}. \tag{12}$$

Kvalitet rješenja se može statistički iskazati i standardnom devijacijom σ na osnovu sljedećeg izraza:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(gap_i - agap)^2}. \tag{13}$$

Ispitan je kvalitet rješenja implementiranog GA u zavisnosti od razmatranih operatora selekcije. Poređeni su rezultati izvršavanja GA-a sa dvopozicionim ukrštanjem i različitim operatorima selekcije (TRS, TOS, RWS i SUS).

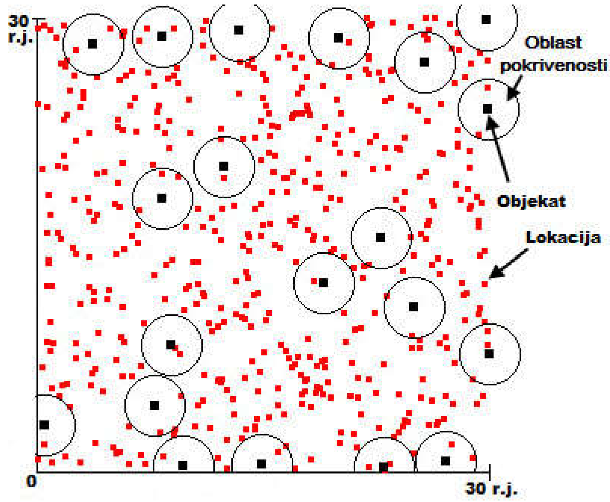
Kao na primjer, na Sl. 2 dat je grafički prikaz rješenja problema MinCLP-MC za instance dimenzije 500, sa parametrima P=20, S=2, D=4, dobijenog metaheuristikom koja implementira turnirsku selekciju.

U Tab.I dati su rezultati testiranja instanci (dimenzija 100, 400 i 500) za različite tipove operatora selekcije i ukupno vrijeme izvršavanja, tj. vrijeme izvršavanja 1000 iteracija. Kolona opt_{sol} (optimalna rješenja) predstavlja vrijednost funkcije cilja rješenja dobijenog pomoću IBM ILOG CPLEX rješavača (optimalna rješenja su preuzeta iz [7]), a kolona $best_{sol}$ predstavlja vrijednost funkcije cilja najboljeg rješenja koje je genetski algoritam dostigao tokom 10 pokretanja.

U Tab. II date su vrijednosti parametara *agap* i σ, tj. ocjene kvaliteta rješenja instanci dobijenih genetskim algoritmom. Na osnovu rezultata prikazanih u Tab. I, može se zaključiti da je za instance dimenzije 100 i 400, najviše optimalnih rješenja dostigao genetski algoritam koji implementira turnirsku selekciju. Najlošija rješenja dobijena su primjenom rulet-točak selekcije i stohastičkog univerzalnog uzorkovanja. Najveće vrijednosti greške *agap* u slučaju TRS i TSO selekcije (za instance dimenzije 100 i 400) su 8.846% i 8.333%, redom, dok najveće vrijednosti devijacije σ iznose 0.055 i 0.047. Vrijednosti ovih parametra za RWS i SUS selekciju su dosta veće. Najmanja vrijednost greške *agap* (u okviru instanci dimenzija 100 i 400) za RWS selekciju je 8%, a najveća 92.222%, dok je za SUS selekciju najmanja vrijednost ovog parametra 6.667%, a najveća 96.296%. Jasno je da metaheuristika sa RWS i SUS selekcijom nije efikasno rješila zadate instance.

Na Sl.3 prikazano je poređenje vremena izvršavanja GA-a različitim operatorima selekcije. Upotrebom operatora selekcije RWS i SUS postiže se brža konvergencija za probleme dimenzija 400 i 500. Rezultati prikazani u Tab. I ukazuju na činjenicu da za veće radijuse pokrivenosti vrijeme konvergencije opada. Na Sl.4 dato je poređenje prosječnog odstupanja rješenja GA-a sa različitim operatorima selekcije. Može se zaključiti da se najbolji kvalitet rješenja postiže primjenom operatora TOS.

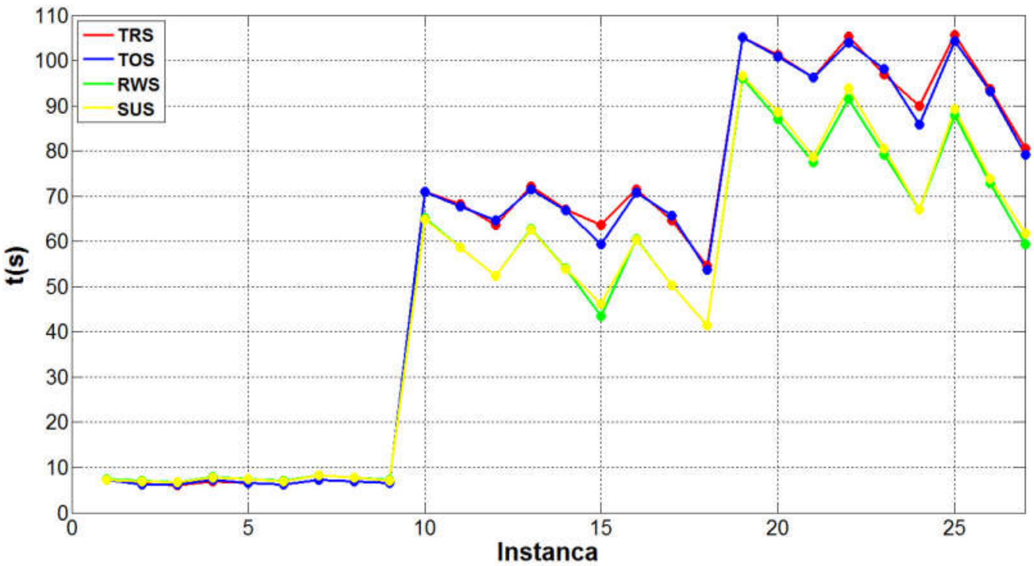
Kako za instance dimenzije 500, CPLEX nije uspio pronaći rješenja, prilikom određivanja parametra *agap* i σ, za određivanje individualne greške dobijenog rješenja korišćena je formula (12). Iz Tab. I može se primijeti da je za instance dimenzije 500, maksimalno vrijeme izvršavanja algoritma sa TRS i TOS selekcijom 105.7 i 105.1 sekundi, redom. Maksimalne vrijednosti parametra *agap* za TRS i TOS selekciju (za instance dimenzije 500) su 5.111% i 5.263%, a maksimalne vrijednosti devijacije σ su 0.038 i 0.080, redom.



Slika 2. Rješavanje instance MinCLP-MC dimenzije 500

TABELA I. REZULTATI TESTIRANJA GA-A SA RAZLIČITIM OPERATORIMA SELEKCIJE

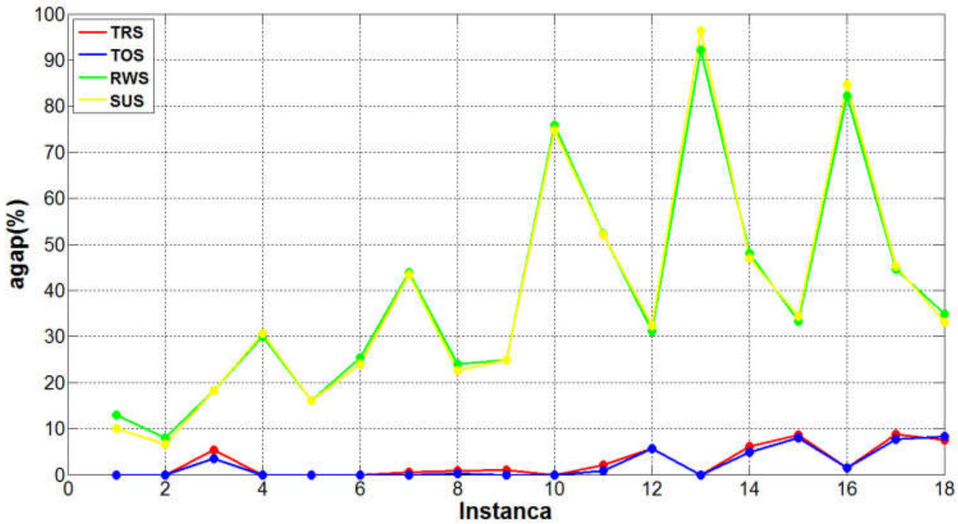
Instanca	*n*	*P*	*S*	*D*	opt_{sol}	$best_{sol}$				*Ukupno vrijeme izvršavanja(s)*			
						TRS	TOS	RWS	SUS	TRS	TOS	RWS	SUS
1	100	10	2	4	10	10	10	11	10	7.2	7.3	7.4	7.2
2	100	10	3	4	15	15	15	15	15	6.4	6.3	7.1	7.0
3	100	10	4	4	17	17	17	19	19	6.1	6.2	6.8	6.7
4	100	15	2	4	15	15	15	19	19	6.9	7.3	7.9	7.8
5	100	15	3	4	25	25	25	27	28	6.6	6.6	7.4	7.4
6	100	15	4	4	28	28	28	33	31	6.2	6.3	7.1	6.9
7	100	20	2	4	21	21	21	30	29	7.3	7.2	8.3	8.2
8	100	20	3	4	35	35	35	42	41	6.9	6.9	7.8	7.7
9	100	20	4	4	42	42	42	49	50	6.5	6.6	7.2	7.1
10	400	10	2	4	17	17	17	28	27	71.0	70.9	65.2	64.9
11	400	10	3	4	43	43	43	63	63	68.3	67.8	58.6	58.7
12	400	10	4	4	74	75	75	89	94	63.6	64.6	52.4	52.4
13	400	15	2	4	27	27	27	48	51	72.2	71.5	62.8	62.6
14	400	15	3	4	73	74	73	105	106	67.0	66.8	54.1	53.9
15	400	15	4	4	112	119	117	146	143	63.7	59.4	43.5	46.2
16	400	20	2	4	42	42	42	70	76	71.4	70.8	60.5	60.4
17	400	20	3	4	104	106	104	145	148	64.7	65.6	50.3	50.3
18	400	20	4	4	150	158	158	190	190	54.6	53.7	41.4	41.4
19	500	10	2	4	-	21	21	35	36	105.1	105.1	96.0	96.6
20	500	10	3	4	-	59	59	79	79	101.2	100.8	87.0	88.6
21	500	10	4	4	-	90	90	118	119	96.2	96.3	77.4	78.7
22	500	15	2	4	-	36	36	60	62	105.3	103.9	91.5	93.8
23	500	15	3	4	-	97	96	130	128	96.9	98.2	79.1	80.5
24	500	15	4	4	-	145	144	185	179	90.0	85.9	67.1	67.1
25	500	20	2	4	-	55	55	93	92	105.7	104.2	87.8	89.2
26	500	20	3	4	-	137	137	184	180	93.7	93.1	72.9	73.9
27	500	20	4	4	-	193	190	249	246	80.6	79.2	59.4	61.8



Slika 3. Prikaz poređenja vremena izvršavanja GA-a sa različitim operatorima selekcije

TABELA II. PROSJEČNA ODSTUPANJA RJEŠENJA DOBIJENIH GA-OM SA RAZLIČITIM OPERATORIMA SELEKCIJE

Instanca	*agap(%)*				σ			
	TRS	TOS	RWS	SUS	TRS	TOS	RWS	SUS
1	0	0	13	10	0	0	0.046	0.045
2	0	0	8	6.667	0	0	0.050	0.042
3	5.294	3.529	18.235	18.235	0.055	0.047	0.032	0.049
4	0	0	30	30,667	0	0	0.033	0.033
5	0	0	16	16	0	0	0.036	0.025
6	0	0	25.357	23,929	0	0	0.034	0.068
7	0.476	0	43.81	43,333	0.014	0	0.019	0.033
8	0.857	0.286	24	22,571	0.013	0.008	0.019	0.032
9	0.952	0	25	25	0.019	0	0.037	0.049
10	0	0	75.882	74.706	0	0	0.055	0.095
11	2.093	0.93	52.326	52.093	0.024	0.015	0.035	0.035
12	5.676	5.676	31.081	32.432	0.029	0.029	0.045	0.030
13	0	0	92.222	96.296	0	0	0.067	0.037
14	6.164	4.932	48.082	46.986	0.031	0.027	0.021	0.015
15	8.661	8.036	33.304	34.375	0.019	0.019	0.020	0.031
16	1.429	1.429	82.143	84.762	0.022	0.019	0.055	0.032
17	8.846	7.692	44.712	45.192	0.036	0.043	0.024	0.014
18	7.467	8.333	34.933	33.133	0.017	0.018	0.035	0.040
19	0	0.952	5.429	2.778	0	0.019	0.039	0.030
20	2.881	1.525	3.797	4.177	0.024	0.016	0.020	0.020
21	5.111	4.111	3.475	2.941	0.027	0.023	0.028	0.024
22	2.222	1.389	6.667	3.71	0.017	0.019	0.032	0.020
23	1.546	2.5	1.692	3.906	0.018	0.014	0.013	0.0160
24	3.103	4.583	2.811	5.587	0.038	0.080	0.016	0.032
25	3.818	3.455	2.688	3.913	0.025	0.021	0.015	0.020
26	1.314	1.314	1.576	3.556	0.011	0.011	0.011	0.015
27	4.041	5.263	2.048	2.683	0.023	0.019	0.014	0.019



Slika 4. Grafički prikaz poređenja prosječnog odstupanja rješenja GA-a sa različitim operatorima selekcije

Zaključak

U ovom radu razmatrana je primjena različitih operatora selekcije genetskog algoritma u rješavanju problema minimalnog pokrivanja lokacija sa višestrukim pokrivanjem. Implementirano je četiri operatora selekcije: selekcija skraćivanja, turnirska selekcija, rulet-točak selekcija i stohastička univerzalna uzorkovanja. Data je analiza performansi četiri varijante GA-a u pogledu prosječnog odstupanja najboljeg rješenja od optimalnog i vremena izvršavanja GA-a. Numerički rezultati potvrdili su da je izbor operatora selekcije jedan od najvažnijih aspekata genetskog algoritma. Najlošija rješenja i najveća prosječna odstupanja rješenja od optimalnih dobijena su primjenom rulet-točak selekcije i stohastičkog univerzalnog uzorkovanja. Najmanja prosječna odstupanja rješenja od optimalnih postignuta su primjenom turnirske selekcije. Za razliku od CPLEX-a koji nije pronašao rješenje za instance problema sa 500 lokacija, genetski algoritam je pronašao rješenje za relativno kratko vrijeme. Kvalitetna rješenja postignuta su primjenom operatora turnirske selekcije, ali i selekcije skraćivanja.

Zahvalnica

Zahvaljujem se dr Darku Drakuliću na korisnim sugestijama i savjetima prilikom izrade ovog rada, koji je nastao u okviru istraživanja vezanog za završni rad na II ciklusu studija Filozofskog fakulteta, Univerziteta u Istočnom Sarajevu.

Literatura

[1] D. Drakulić, M. Marić, A. Takači, “Solving Maximal Covering Location Problem (MCLP) by Using the Particle Swarm Optimization (PSO) Method”, НАУЧНИ ТРУДОВЕ НА РУСЕНСКИЯ УНИВЕРСИТЕТ - 2012, том 51, серия 6.1

[2] E. Erkut and S. Neuman, “Analytical models for locating undesirable facilities”, European Journal of Operational Research, Volume 40, Issue 3, page 275–291, 1989.

[3] Z. Drezner , G. Wesolowsky, “Finding the circle or rectangle containing the minimum weight of points”, Location Science; 2:83–90. 1994.

[4] O. Berman, Z. Drezner and G. Wesolowsky, “Minimum covering criterion for obnoxious facility location on a network”, Networks;28:1–5., 1996.

[5] O. Berman and R. Huang, “The minimum weighted covering location problem with distance constraints”, Computers and Operations Research 35, page 356 – 372, 2008.

[6] D. Drakulić., A. Takači, M. Marić , “ The Minimal Covering Location Problem with single and multiple location coverage”, 6th International Conference on Information Society and Technology ICIST 2016.

[7] D. Drakulić, “Fazi skupovi u modelovanju lokacijskih problema kombinatorne optimizacije”, Doktorska disertacija, Univerzitet u Istočnom Sarajevu, 2016.

[8] M. Dorigo and T. Stützle , “The ant colony optimization metaheuristic: Algorithms, applications, and advances”, Handbook of Metaheuristics, Springer, pp. 250–285, 2003.

[9] M. Gendreau and J.-Y. Potvin, “Metaheuristics in combinatorial optimization”, Annals of Operations Research, Vol. 140 No. 1, pp. 189–213., 2005.

[10] M. Marić, “Rješavanje nekih NP-teških hijerarhijsko-lokacijskih probelma primjenom genetskog algoritma”, Doktorska disertacija, Matematički fakultet, Beograd, 2008.

[11] I. H. Osman , J. P.Kelly, "Metaheuristics: Theory and Applications", Kluwer Academic Publisher, Norwell 1996.

[12] E.G. Talbi, “Metaheuristics – From Design to Implementation”, John Wiley and Sons, 2009.

[13] V. Filipović, "Operatori selekcije i migracije i WEB servisi kod paralelnih evolutivnih algoritama", Doktorska disertacija, Matematički fakultet, Beograd 2006.

Abstract

Minimization of the number of locations covered by undesirable objects represents combinatorial optimization problem of great practical importance. This paper analyzes solving of the Minimal Covering Location Problem with multiple coverage allowed using genetic algorithm. The results of the comparative analysis of the impact of different types of selection operators on the efficiency of finding good solutions for spaces consisting of 100, 400 and 500 locations, are presented. Based on numerically obtained results, associated with solutions' quality and algorithm's execution time, adequate selection operators are identified.

SOLVING MINIMAL COVERING LOCATION PROBLEM BY USING DIFFERENT SELECTION OPERATORS OF GENETIC ALGORITHM

Jovana Janković