

Matematčki fakultet

Metaheurističke metode

Seminarski rad

**Rešavanje problema nepoželjne p-mediane korišćenjem
metode optimizacije roja čestica**

Student:
Damjan Stanković
1081/2024

Nastavnik:
dr Zorica Stanimirović



1 Opis problema

Problem nepoželjne p-medijane (engl obnoxious p-median problem) je varijanta poznatijeg problema p-mediane. Zadatak problema p-mediane je izbor lokacija na kojima je potrebno postaviti uslužne tačke tako da se minimizuje trošak usluge svih korisnika na nekom prosotru.

Problem nepoželjne p-medijane zahteva da se na zadanom konačnom skupu potencijalnih lokacija postavi određen broj (p) postrojenja koja se pridružuju njima najbližim korisnicima tako da se zbir distanci od korisnika do postrojenja koja su im pridružena maksimizuje. U praksi se pojavljuje u planiranju lokacija deponija, hemijskih i nuklearnih postrojenja.

Jedan od prvih radova koji se bavi ovim problemom je [1]. Problem je prepoznat kao NP težak. Ranije korištene metaheurističke metode za ovaj problem su GRASP [7] i VNS [5].

U ovom radu je implementovana metaheuristička metoda zasnovana na optimizaciji roja čestica (engl. particle swarm optimization) i njene performanse su upoređene sa CPLEX rešavačem.

2 Matematičke formulacije

2.1 Originalni model

Prva formulacija je postavljena u radu [3]. Ova formulacija je optimizovana za rešavanje pomoću Branch-and-cut metode. Problem je formulisan kao problem binarnog linearnog programiranja.

Objašnjenja parametara:

- I - Skup korisnika
- J - Skup lokacija potencijalnih centara
- $c_{i,j}$ - Udaljenosti korisnika i lokacija potencijalnih centara

Objašnjenja promenljivih odluke:

- y_j - promenljiva odluke o uspostavljanju centra na lokaciji
- $x_{i,j}$ - promenljiva alokacije korisnika i centru j
- $S(i,j)$ - skup svih potencijalnih centara iz J koji su udaljeniji od korisnika i u odnosu na potencijalni centar j

Matematički zapis modela je:

$$\max \sum_{i \in I} \sum_{j \in J} c_{i,j} x_{i,j} \quad (1)$$

$$\text{s.t.} \quad \sum_{j \in J} y_j = p \quad (2)$$

$$x_{i,j} \leq y_j \quad \forall i \in I, j \in J \quad (3)$$

$$y_j + \sum_{k \in S(i,j)} x_{i,k} \leq 1 \quad \forall i \in I, j \in J \quad (4)$$

$$x_{i,j}, y_j \in \{0, 1\} \quad \forall i \in I, j \in J \quad (5)$$

Funkcija cilja (1) predstavlja sumu distanci korisnika od centara kojima su alocirani. Ograničenja osiguravaju da korisnik bude alocirano samo najbližem uspostavljenom centru. Cilj problema je maksimizacija funkcije cilja.

Ograničenje (2) osigurava da ćemo otvoriti p centara.

Ograničenje (3) osigurava da je scaki korisnik alocirano otvorenom centru.

Ograničenje (4) osigurava da je korisnik alocirano najbližem otvorenom centru. Ako je $y_j = 1$ to znači da je centar j otvoren, i da korisnik ne može biti alocirano daljem centru. Ako je $y_j = 0$ onda centar j nije otvoren, a korisnik može biti alocirano samo jednom od udaljenijih centara. Primetimo da ovo ograničenje takođe garantuje da jedan korisnik može biti alocirano najviše jednom centru.

Ograničenje (5) znači da su $x_{i,j}$ i y_j binarne promenljive.

Ovaj model koristi CPLEX jer je napravljen specifično da bi bio lako rešiv metodama koje koristi rešavač.

2.2 Kompaktni model

U radu [4] je predstavljena alternativna, kompaktnija formulacija istog problema:

Sve oznake koje su iste kao u prethodnoj formulaciji imaju isto značenje.

Novi parametri su:

- d_i je promenljiva koja meri udaljenost od korisnika i do njemu najbližeg centra.
- M je dovoljno veliki broj da osigura da će $c_{i,j} + M$ biti veće od bilo kog drugog $c_{i,j'}$. Najjednostavnije je postaviti $M = \max_{i \in I, j \in J} (c_{i,j})$

$$\max \sum_{i \in I} d_i \quad (6)$$

$$\text{s.t.} \quad \sum_{j \in J} y_j = p \quad (7)$$

$$d_i \leq c_{i,j} + (1 - y_j)M \quad \forall i \in I, j \in J \quad (8)$$

$$y_j \in \{0, 1\} \quad \forall j \in J \quad (9)$$

Funkcija cilja (6) je suma svih rastojanja korisnika od njima pridruženih centara, a cilj problema je njena maksimizacija. Ograničenja osiguravaju da d_i jeste upravo stanica do najbližeg uspostavljenog centra.

Ograničenje (7) osigurava da otvaramo tačno p centara.

Ograničenje (8) Osigurava da je d_i manje ili jednako od udaljenosti do svih uspostavljenih centara ($y_j = 1$). Ako je $y_j = 0$ M neutrališe ograničenje.

Ograničenje 9 osigurava da je y_j binarna promenljiva.

Ovu verziju modela predstavljamo jer je intuitivnija i bliža problemu kako ga implementiramo u kodu.

3 Metaheuristička metoda - optimizacija rojem čestica

3.1 Generalni uvod

Particle swarm optimization je metaheuristička metoda koja se u literaturi prvi put pojavljuje u radu Kenedija i Eberharta[6].

Metoda simulira ponašanje životinja u grupama, recimo jata ptica ili riba. Metoda se inicijalizuje tako što generišemo nasumičnu populaciju početnih rešenja koja se onda ponašaju kao simulirano jato.

Ponašanje svakog rešenja u jatu ima tri osnovna aspekta:

1. Mehanička inercija: svako rešenje pamti pomeraj od svoje pozicije u prethodnoj iteraciji, i koristi je da pronađe svoju sledeću poziciju.
2. Kognitivna inercija: svako rešenje pamti svoju najbolju raniju poziciju i teži da se u nju vrati.
3. Socijalna inercija: svako rešenje zna najbolje rešenje koje je jato prethodno našlo i teži da mu se približi.

Formula za nalaženje naredne pozicije svakog rešenja iz jata je:

$$v_{i+1} = \omega \cdot v_i + c_1 \cdot p_{best} + c_2 \cdot g_{best}. \quad (10)$$

$$x_{i+1} = \text{clip}(x_i + v_{i+1}). \quad (11)$$

Objašnjenja promenljivih i parametara:

- x_i - i -ta pozicija rešenja iz jata,
- ω - koeficijent mehaničke inercije,
- v_i - korak od x_{i-1} do x_i ,
- c_1 - koeficijent kognitivne inercije,
- p_{best} - najbolje rešenje koje je ova jedinka jata našla,
- c_2 - koeficijent socijalne inercije,
- g_{best} - najbolje rešenje koje je bilo koja jedinka jata pronašla,
- $\text{clip}(\ast)$ - operator koji rešenje vraća u dopustivi skup.

3.2 Implementacija metode za problem nepoželjne p-medijane

Rešenje problema je predstavljeno vektorom dužine p čiji su elementi indeksi izabranih centara.

Funkcija cilja se računa sa složenosti $O(np)$ tako što se samo nađu najmanje distance od svakog korisnika do nekog uspostavljenog centra.

Tokom inicijalnog testiranja primećeno je da metoda brzo konvergira ka suboptimalnim rešenjima, te je uzimanjem ideje slične kao u radu [8] odlučeno da se u svakom koraku socijalni i kognitivni parametri perturbuju sa matricama r_1, r_2 čiji su elementi uniformno raspoređeni na intervalu $[0, 1]$.

Implementacija se sastoji iz dve osnovne petlje: eksploraciona i eksploataciona.

Eksploraciona petlja se kreće kroz čitav prostor dopustivih rešenja i traži najbolje rešenje. Kako bi izbegli prebrzu konvergenciju rešenja perturbujemo socijalni i kognitivni aspekt promene brzine, tako da formula ima sledeći oblik:

$$v_{i+1} = \omega \cdot v_i + c_1 \cdot r_1 \cdot p_{best} + c_2 \cdot r_2 \cdot g_{best}. \quad (12)$$

gde su r_1 i r_2 nasumične matrice dimenzije $n \times \text{swarm_size}$ sa elementima nasumično unifomno izabranim iz segmenta $[0, 1]$. swarm_size je parametar veličine jata. Parametri (ω, c_1, c_2) su inicijalno postavljeni na $(0.8, 2, 2)$. Pri inicijalnom testiranju nije ustanovljeno značajno poboljšanje efikasnosti pri promeni parametara, tako da su ostavljeni.

Eksploraciona matrica se završava ako prođe 2000 iteracija, ili u 300 iteracija ne nađe bolje rešenje.

Eksploataciona matrica se ponaša slično, osim što čitavo jato inicijalizuje u poziciju najboljeg rešenja nađenog u eksploracionoj fazi, a brzine se inicijalizuju kao nasumično vektori i zanemaruje socijalnu inerciju. Formula promene jedinke jata ima sledeći oblik:

$$v_{i+1} = \omega \cdot v_i + c_1 \cdot r_1 \cdot p_{best}. \quad (13)$$

3.2.1 Pseudokod

```

Inicijalizuj jato
Pronađi najbolje rešenje
Za iteraciju od 1 do max_iter:
    Za i od 1 do veličine jata:
        Oцени trenutnu poziciju  $x_i$ 
        Promeni lično i globalno najbolje rešenje
        Promeni brzinu i položaj po 12
        Testiraj kriterijum zaustavljanja
Inicijalizuj jato za eksploataciju:
Za iteraciju od 1 do max_iter:
    Za i od 1 do veličine jata:
        Oцени trenutnu poziciju  $x_i$ 
        Promeni lično i globalno najbolje rešenje
        Promeni brzinu i položaj po 13
        Testiraj kriterijum zaustavljanja

```

Pun kod i rezultate koje dobijamo implementacijom možete naći na linku: https://drive.google.com/drive/folders/1AtIrP8566BVnlwHCVB1Hth7jUu4NVq9q?usp=drive_link

4 Instance

4.1 Generisanje instanci

Instance su generisane na osnovu instanci OpM.LIB.2016. Ove instance su preuzete sa sajta istraživačke grupe GRAFO [2]. Instance korištene u ovom radu su generisane na osnovu instanci pmed17, specifično pmed17.txt.table.p25.A, pmed17.txt.table.p25.B, pmed17.txt.table.p50A, pmed17.txt.table.p50B, pmed17.txt.table.p100A, pmed17.txt.table.p100B.

Instance u biblioteci su dimenzija 200×200 , te su izvan mogućnosti CPLEX rešavača da ih reši u 2 sata. U skladu sa tim na osnovu obe ove instance generisane su 72 nove instance.

Iz matrica uzeti su prvi minori matrice distanci iz biblioteke veličine 50×50 , 100×100 , 150×150 i 200×200 te za njih varirali parametar p : za instance 50×50 $p = 10, 15, 25$, za instance 100×100 $p = 15, 25, 50$, za instance 150×150 $p = 25, 50, 75$, za instance 200×200 $p = 25, 50, 100$.

Instance su podeljene u tri grupe:

- Male: 50×50 za $p = 15, 20, 25$ i 100×100 za $p = 15$,
- Srednje: 100×100 $p = 25, 50$ i 150×150 $p = 25, 50$,
- Velike: 150×150 $p = 75$ i 200×200 za $p = 25, 50, 100$.

Za prilagođavanje instanci čitanju podataka svaki *.txt* fajl sa podacima je sledećeg oblika:

- Ime fajla: *pmed17pnC_nxm.pk* gde *nC* određuju instancu na osnovu koje je generisana nova, $n \times m$ je dimenzija minora a *k* je broj centara koje treba uspostaviti.
- Forma fajla: Svi fajlovi su oblika:
 n
 m
 p
 M
777
gde n i m označavaju dimenziju matrice u nazivu fajla smo t , p broj centara koje treba uspostaviti a M je $n \times m$ matrica sa distancama između korisnika i potencijalnih centara. 777 je konstanta za proveru validnosti učitavanja.

4.2 Primer instance

Način formatiranja instance u txt fajlu je naveden niže. Parametri n i m su odvojeni radi primenjivosti koda na instance gde broj potencijalnih centara i korisnika nije jednak. Ovaj rad se ne bavi takvim instancama.

```
4
4
2
19.00 22.00 50.00 12.00
26.00 36.00 51.00 19.00
30.00 36.00 39.00 41.00
40.00 49.00 50.00 39.00
777
```

Optimalno rešenje u ovom slučaju je uspostaviti centre (2,3) i vrednost funkcije cilja je tada 133.

5 Rezultati

Opisanim postupkom su generisane 72 instance. Iz oblika u kom su bile u biblioteci u oblik naveden iznad su prevedene python skriptom.

Metoda je implementirana u jeziku Python koristeći numpy paket za proračune. Rezultati su dobijenu na mašini sa procesorom AMD Ryzen 5 7535HS i 16GB RAM memorije. Rezultati CPLEX rešavača su generisani na istoj mašini koristeći IBM ILOG CPLEX Optimization Studio 22.1.1 rešavača, sa ograničenjem trajanja rada na 2 sata. Vreme u tabelama za CPLEX je procena potrebnog

vremena za postizanje istog rezultata na jednom jezgru koju je izračunao sam CPLEX.

Rezultati osnovne metode sa jatom od 1000 jedinki su u tabelama 1, 2, 3. U tabeli su podaci o rezultatima koje dobija CPLEX za instancu, vreme izvršavanja, broj iteracija i broj čvorova. U tabeli su sačuvane maksimalne vrednosti funkcije cilja koje je našao PSO, vreme do prvog pronalaska tog rešenja u sekundama (t), ukupno vreme metode u sekundama (t_{tot}), broj iteracija, prosečno odstupanje od optimalnog nađenog rešenja u procentima ($agap$) i standardna devijacija u procentima (σ).

Tabela 4 poredi učinak PSO u slučaju kada uzmemo veće jato (2000 jedinki) ili kada se kriterijumi zaustavljanja bazirani na iteracijama bez poboljšanja udvostruče (eksploatacija sa 300 na 600, a eksploatacija sa 200 na 400). U tabeli su boldovani bolji rezultati u pogledu vremena izvršavanja i vrednosti funkcije cilja.

6 Analiza rezultata

6.1 Poređenje PSO i CPLEX-a

U tabeli 1 vidimo da za male instance CPLEX i PSO rade uporedivo brzo i precizno, mada u najvećim instancama u ovoj tabeli PSO ima rešenja manja za oko

U tabeli 2 vidimo da PSO pronalazi rešenja čija je vrednost funkcije cilja koja su preko 96% iste vrednosti rešenja koja nalazi CPLEX za vreme koje je od 30-400 puta kraće.

U tabeli 3 vidimo da PSO pronalazi rešenja čija je vrednost funkcije cilja koja su preko 96% iste vrednosti rešenja koja nalazi CPLEX za vreme koje je par hiljada puta kraće.

6.2 Poređenje povećanja broja iteracija i populacije jata

U tabeli 4 se nalaze rezultati dobijeni primenom PSO sa duplo većim jatom (2000 jedinki) i sa duplo većim dozvoljenim brojem iteracija bez poboljšanja (600 za eksploataciju i 400 za eksploataciju).

Primećujemo da povećanje jata generalno daje optimalnija rešenja, mada za jednostavnije verzije problema se izvršava sporije.

Dolazimo do zaključka da ako nam je potrebno poboljšanje dobijenog rešenja u većini slučajeva želimo povećati jato a ne broj iteracija.

Tabela 1: Instance malih dimenzija, jato od 1000 jedinki

Ime instance	CPLEX					PSO				
	Rezultat	Vreme(s)	#iteracija	#čvorova	Rezultat	$t(s)$	$t_{tot}(s)$	#iteracija	$agap(\%)$	$\sigma(\%)$
pmed17p25A_50x50_p15	1631	9.02	1010	0	1631	0.44	6.26	38.05	0.03	0.08
pmed17p25A_50X50_p20	1441	4	1184	0	1441	0.56	7.46	37.25	0.04	0.10
pmed17p25A_50x50_p25	1283	3.20	893	0	1283	0.96	9.94	63.75	0.03	0.06
pmed17p25A_100x100_p15	3662	4261.08	1605453	8376	3626	2.30	10.99	165.65	0.04	0.13
pmed17p25B_50x50_p15	1659	4.41	1179	0	1659	0.53	6.83	43.55	0.03	0.12
pmed17p25B_50X50_p20	1453	9.70	2156	0	1453	0.68	7.09	56.50	0.04	0.13
pmed17p25B_50x50_p25	1253	5.52	1230	0	1253	0.67	7.18	55.35	0.03	0.09
pmed17p25B_100x100_p15	3676	1282.16	699726	7125	3643	2.0321	10.38	152.30	0.02	0.12
pmed17p50A_50x50_p15	1777	2.70	698	0	1777	0.60	6.85	49.65	0.03	0.09
pmed17p50A_50X50_p20	1567	2.69	1180	0	1567	1.00	8.75	71.40	0.04	0.11
pmed17p50A_50x50_p25	1386	2.69	1015	0	1386	0.77	7.82	58.95	0.03	0.13
pmed17p50A_100x100_p15	3955	2847.84	1175935	9981	3941	1.98	10.15	148.95	0.05	0.11
pmed17p50B_50x50_p15	1601	29.56	82311	2512	1601	1.08	8.49	77.35	0.01	0.05
pmed17p50B_50X50_p20	1445	23.78	56134	2476	1445	0.78	8.04	57.50	0.02	0.08
pmed17p50B_50x50_p25	1318	12.89	50117	1293	1318	0.88	8.32	64.20	0.02	0.04
pmed17p50B_100x100_p15	3678	6482.25	1770944	15211	3632	2.13	10.51	161.15	0.04	0.12
pmed17p100A_50x50_p15	1585	9.09	2016	0	1585	0.64	7.75	46.15	0.03	0.09
pmed17p100A_50X50_p20	1398	11.91	24990	1059	1398	0.63	7.97	43.45	0.02	0.06
pmed17p100A_50x50_p25	1284	3.91	958	0	1281	1.32	13.62	57.45	0.03	0.08
pmed17p100A_100x100_p15	3784	991.44	484583	4950	3749	2.50	11.29	189.10	0.04	0.15
pmed17p100B_50x50_p15	1665	5.16	941	0	1665	0.44	6.31	37.65	0.02	0.05
pmed17p100B_50X50_p20	1484	6.72	1188	0	1484	0.53	6.60	45.25	0.03	0.07
pmed17p100B_50x50_p25	1328	2.52	926	0	1322	0.80	7.39	67.00	0.02	0.05
pmed17p100B_100x100_p15	3652	5857.67	2287986	13207	3642	2.64	11.49	198.95	0.03	0.09

Tabela 2: Instance srednjih dimenzija, jato od 1000 jedinki

Ime instance	CPLEX				PSO					
	Rezultat	Vreme (s)	# iteracija	# čvorova	Rezultat	t (s)	t _{tot} (s)	# iteracija	agap (%)	σ (%)
pmed17p25A_100x100_p25	3148	3072.47	1139170	13570	3050	1.76	10.30	123.25	0.0387	0.11
pmed17p25A_100X100_p50	2288	630.30	456009	7491	2159	2.46	12.48	154.10	0.05	0.11
pmed17p25A_150x150_p25	4903*	67187	9467896	44553	4688	8.01	27.13	310.00	0.04	0.10
pmed17p25A_150X150_p50	3865*	74495.5	15350602	80743	3544	3.29	19.86	117.30	0.03	0.0861
pmed17p25B_100x100_p25	3115	2013.09	861647	6845	2964	1.83	11.01	118.90	0.04	0.13
pmed17p25B_100X100_p50	2180	754.06	595016	6482	2066	1.12	9.97	68.90	0.04	0.08
pmed17p25B_150x150_p25	5192	50372.3	11477280	35094	5065	4.05	17.18	200.90	0.09	0.19
pmed17p25B_150X150_p50	4005	18903.8	2921110	18730	3568	1.53	13.87	65.40	0.05	0.14
pmed17p50A_100x100_p25	3339	774.11	397765	4156	3251	1.73	10.40	113.65	0.07	0.15
pmed17p50A_100X100_p50	2341	851.78	488296	7435	2189	0.74	9.19	44.95	0.04	0.11
pmed17p50A_150x150_p25	5302*	75188.8	12109730	35068	5001	5.30	19.51	261.25	0.04	0.13
pmed17p50A_150X150_p50	4048	22189.4	3614628	22683	3670	2.91	16.63	123.95	0.05	0.11
pmed17p50B_100x100_p25	3147	2417.98	878607	10278	3053	1.55	9.84	108.85	0.04	0.09
pmed17p50B_100X100_p50	2372	1157.39	714555	13888	2263	0.83	9.42	49.95	0.03	0.08
pmed17p50B_150x150_p25	5029*	71567.3	11915076	41277	4772	5.97	21.04	299.65	0.04	0.09
pmed17p50B_150X150_p50	3924*	67398.8	11336775	58762	3635	2.54	16.22	108.55	0.05	0.11
pmed17p100A_100X100_p25	3095	4473.8	2016534	12048	3036	2.35	11.49	165.70	0.04	0.11
pmed17p100A_100x100_p50	2288	710.67	487586	7016	2220	1.22	10.18	74.45	0.05	0.10
pmed17p100A_150x150_p25	5150*	71820	11946923	41850	4832	4.66	18.30	235.70	0.05	0.10
pmed17p100A_150X150_p50	3882*	77946.8	18071094	101831	3537	2.52	15.81	106.20	0.03	0.08
pmed17p100B_100X100_p25	3145	7227.11	2618505	41129	3041	1.73	10.20	120.45	0.04	0.11
pmed17p100B_100x100_p50	2383	660.52	357236	4534	2315	1.01	9.84	61.35	0.06	0.11
pmed17p100B_150x150_p25	5049*	62726.5	9546496	26082	4865	6.06	20.91	304.60	0.05	0.14
pmed17p100B_150X150_p50	3973	32042.2	5360046	25093	3707	3.30	17.13	145.05	0.05	0.10

Tabela 3: Instance velikih dimenzija, jato od 1000 jedinki

Ime instance	CPLEX					PSO				
	Rezultat	Vreme (s)	# iteracija	# čvorova	Rezultat	t (s)	t_{tot} (s)	# iteracija	$agap$ (%)	σ (%)
pmed17p25A_150x150_p75	3205	14123.4	3552180	25181	2921	3.17	19.89	105.15	0.03	0.10
pmed17p25A_200x200_p25	6690*	55818.7	3751981	9370	6406	9.36	27.85	398.45	0.03	0.11
pmed17p25A_200X200_p50	5426*	70086.4	5720237	18125	4929	5.54	23.36	203.15	0.03	0.10
pmed17p25A_200x200_p100	4003*	68741.5	3992328	16397	3529	6.20	33.16	139.45	0.04	0.11
pmed17p25B_150x150_p75	3200	8051.27	1703242	11464	2832	2.92	18.10	110.75	0.03	0.08
pmed17p25B_200x200_p25	7212*	63314.6	4723793	6122	6487	6.92	23.67	300.15	0.07	0.11
pmed17p25B_200X200_p50	5670*	74558.8	6083869	27024	4978	7.46	25.72	284.75	0.05	0.13
pmed17p25B_200x200_p100	4109*	78291.3	6256727	35087	3470	3.48	23.74	93.50	0.04	0.08
pmed17p50A_150x150_p75	3257	26780.1	6727304	45959	2897	2.17	16.48	82.55	0.04	0.11
pmed17p50A_200x200_p25	6964*	65556.4	4254247	9107	6779	12.04	31.10	529.40	0.08	0.16
pmed17p50A_200X200_p50	5328*	65424.3	5750124	15457	4950	8.83	29.14	328.75	0.04	0.09
pmed17p50A_200x200_p100	3887*	65227.4	4621528	22506	3337	2.96	21.30	87.65	0.04	0.08
pmed17p50B_150x150_p75	3202*	62772.4	13649000	107531	2909	2.60	17.48	98.05	0.02	0.07
pmed17p50B_200x200_p25	6647*	50079.3	3063154	13912	6301	10.69	31.86	403.95	0.06	0.13
pmed17p50B_200X200_p50	5207*	75731.3	6105763	23105	4916	13.53	46.19	308.25	0.04	0.10
pmed17p50B_200x200_p100	3852*	77390.4	6481882	38716	3350	3.59	28.00	79.50	0.02	0.06
pmed17p100A_150x150_p75	3207	36931.2	15103028	373050	2937	2.11	16.53	79.95	0.04	0.11
pmed17p100A_200x200_p25	6888*	65929.1	5598175	9994	6630	17.24	43.47	566.10	0.10	0.18
pmed17p100A_200X200_p50	5253*	75467.3	6167025	19127	4939	11.78	39.39	354.60	0.05	0.11
pmed17p100A_200x200_p100	3949*	77544.4	4714840	25548	3370	2.68	21.34	76.20	0.04	0.08
pmed17p100B_150x150_p75	3314	13018.2	2320152	19785	2988	2.63	17.30	102.15	0.04	0.09
pmed17p100B_200x200_p25	6816*	65485.9	4031203	12572	6686	10.46	29.55	390.40	0.08	0.12
pmed17p100B_200X200_p50	5419*	75750.8	5959340	17046	4925	7.89	27.99	285.80	0.05	0.09
pmed17p100B_200x200_p100	4013*	77273.6	7736962	41735	3435	2.61	21.53	74.35	0.04	0.07

Tabela 4: Instance 100×100 , poređenje povećanja broja instanci i populacije jata

Ime instance	Rezultat CPLEX	Veće jato		Više iteracija	
		Rezultat	Vreme	Rezultat	Vreme
pmed17p25A_100x100_p15	3662	3654	29.6575	3626	19.3380
pmed17p25A_100x100_p25	3148	3069	36.2545	3050	18.2493
pmed17p25A_100X100_p50	2288	2177	37.7170	2159	23.0262
pmed17p25B_100x100_p15	3676	3624	238.8514	3643	18.3049
pmed17p25B_100x100_p25	3115	3009	21.9369	2964	17.7345
pmed17p25B_100X100_p50	2180	2071	22.2117	2066	18.9832
pmed17p50A_100x100_p15	3955	3911	24.5304	3941	19.5660
pmed17p50A_100x100_p25	3339	3229	22.6135	3251	18.0507
pmed17p50A_100X100_p50	2341	2241	24.3712	2189	18.6411
pmed17p50B_100x100_p15	3678	3656	22.5019	3632	18.7961
pmed17p50B_100x100_p25	3147	3036	23.8810	3053	18.2951
pmed17p50B_100X100_p50	2372	2325	21.9503	2263	21.1027
pmed17p100A_100x100_p15	3784	3768	22.5210	3749	25.4480
pmed17p100A_100X100_p25	3095	3031	26.2059	3036	31.0145
pmed17p100A_100x100_p50	2288	2232	24.3193	2220	20.1815
pmed17p100B_100x100_p15	3652	3652	24.0915	3642	32.8806
pmed17p100B_100X100_p25	3145	3098	24.2639	3041	31.6282
pmed17p100B_100x100_p50	2383	2282	22.3180	2315	32.1637

Literatura

- [1] S.B. Welch and S. Salhi. *The obnoxious p facility network location problem with facility interaction*. EJOR, 102:302–319, 1997.
- [2] Colmenar, J. M., Greistorfer, P., Martí, R., Duarte A. *Obnoxious p -median Problem*, Optscicom Project, 2016
- [3] Labbé et al. *Obnoxious p -Median problems: valid inequalities and a Branch-and-Cut approach*, ORP3, Pariz, 2001.
- [4] Lin Chang-Chun, Chiang Yen-I, *Alternative formulations for the obnoxious p -median problem*, Discrete Applied Mathematics, 289, 366-373, 2021.
- [5] Mladenović Nenad, Alkandari Abdulaziz, Pei Jun, Todosijević Raca, Pardalos Panos M. *Less is more approach: basic variable neighborhood search for the obnoxious p -median problem*, Intl. Trans. in Op. Res. 27 (2020) 480–493
- [6] Kennedy, J.; Eberhart, R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks. Vol. IV. pp. 1942–1948.
- [7] J. Manuel Colmenar, Peter Greistorfer, Rafael Martí, Abraham Duarte, *Advanced Greedy Randomized Adaptive Search Procedure for the Obnoxious p -Median problem*, EJOR, Volume 252, Issue 2, 2016, Pages 432-442, ISSN 0377-2217.
- [8] W. T. Li, X. W. Shi, *An improved Particle Swarm Algorithm for pattern synthesis of phased systems*, Progress In Electromagnetics Research, PIER 82, 319–332, 2008.