

Maximum Map Labeling

Mihajlo Živković Svetožar Iković

Septembar 2023.

Sadržaj

1	Uvod	3
1.1	Problem	3
1.2	Rešenje	3
2	Optimalni algoritmi	4
2.1	Brut force	4
2.2	Brutfoce sa kešom	4
2.3	B algoritam	4
2.3.1	Faza I:Predobrada	4
2.3.2	Faza II: Eliminacija nemogućih kandidata	4
2.3.3	Faza III: Uključivanje heuristike	4
3	Metaheuristike	5
3.1	Genetski algoritam	5
3.1.1	Hromozom	5
3.1.2	Mutacije	5
3.1.3	Selekcija, elitizam i ukrštanje	5
3.1.4	Fitnes funkcija	5
3.2	Poboljšani genetski algoritam	5
3.2.1	Hromozom i mutacija	5
3.2.2	Fitnes funkcija	6
3.2.3	Fitnes	6
4	Rezultati	6
5	Zaključak	9

1 Uvod

Obeležavanje mapa je jedan od klasičnih ključnih problema koji se mora rešiti u procesu proizvodnje mape. Obično proizvođač mape ne želi da prikaže samo tačne geografske pozicije prikazanih objekata, već i da objasni osobine tih objekata. Ona mora organizovati ovu informaciju na mapi tako da:

- za svaki deo informacije bude intuitivno jasno koji objekat je opisan;
- informacije budu čitljive veličine;
- različiti tekstovi se ne preklapaju.

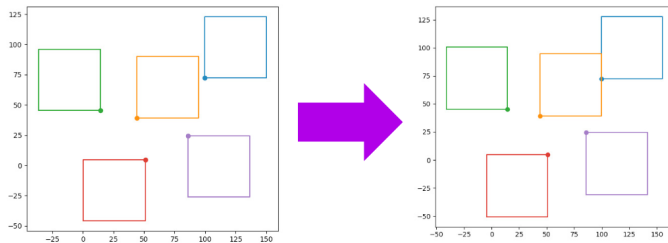
Sa razvojem internet tehnologija i onlajn mapa, stvara se potreba za načinima za učitavanje i obeležavanje mapa u realnom vremenu.

1.1 Problem

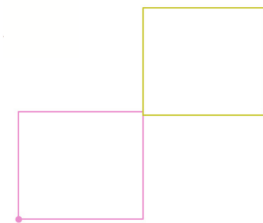
Maksimalno obeležavanje mape je problem postavljanja jednakih kvadrata, paralelnih sa x-osom u ravni, najveće moguće dimanzije, tako da se nikoja dva kvadrata ne presecaju. Svaki kvadrat pripada tačno jednoj tački koja se nalazi u jednom od njegovih temena.

1.2 Rešenje

Broj potencijalno optimalnih rešenja je ograničen, jer veličine za koje se kvadrati ne dodiruju, mogu sigurno da se prošire (slika 1). Za potencijalne veličine uzimamo razdaljine izmedju tačaka, i njihove polu-razdaljine(slika 2).



Slika 1: Optimalne veličine



Slika 2: polovina razdaljine

Obzirom da imamo ograničen broj potencijalno optimalnih veličina, možemo ih sortirati i binarnom pretragom naći optimalnu.

2 Optimalni algoritmi

Zasnivaju se na binarnoj pretrazi mogućih veličina kvadrata. Za svaku veličinu se proverava da li postoji kombinacija orijentacija kvadrata tačaka, tako da nikoja dva nemaju presek.

2.1 Brut force

Ovaj algoritam je najintuitivniji, ali zato i najsporiji. Zadatak mu je da prođe kroz sve moguće kombinacije kvadrata za datu veličinu i da vrati optimalan raspored ako je moguće.

2.2 Brutfoce sa kešom

Brut force sa kešom je sličan brut force algoritmu, sa dodatnim mehanizmom za čuvanje rešenja za već tražene rasporede tačaka.

2.3 B algoritam

Ovaj algoritam je razvijan po ugledu na Franka Vagnera i Aleksadra Volfa[1] u njihovom pokušaju da označe sve podzemne vode grada Minhena.

Validacija se sastoji od tri faze:

Faza I: Predobrada.

Faza II: Eliminisanje kandidata koji ne mogu biti deo rešenja.

Faza III: Za one tačke koje još uvek imaju dva ili više preostalih kandidata, izaberite tačno dva i proverite da li se ovaj preostali problem može rešiti pomoću 2-SAT.

2.3.1 Faza I:Predobrada

U ovoj fazi se kreira lista konflikata i izbacuju se kandidati koji sadrže drugu tačku.

2.3.2 Faza II: Eliminacija nemogućih kandidata

Prolazimo kroz sve tačke. Razmatramo sledeće četiri slučaja:

- Ako su svi kandidati trenutne tačke eliminisani, zaustavljamo se i vraćamo "nema rešenja za datu veličinu" programu koji obavlja binarnu pretragu na listi konflikata.
- Ako trenutna tačka ima kandidate koji se ne preklapaju sa drugim kandidatima, biramo proizvoljni od njih i eliminišemo sve druge kandidate date tačke tačke. Pre brisanja kandidata, vršimo sledeće ažuriranja: brišemo njegovu listu informacija o preklapanju i simetrične unose sa svim kandidatima koji se preklapaju s njim.
- Ako trenutna tačka ima samo jednog preostalog kandidata, izvršavamo ista ažuriranja sa svim kandidatima koji se preklapaju sa njim, a zatim ih brišemo.
- Ako trenutna tačka ima kandidata koji se preklapa sa poslednja dva kandidata druge tačke, ažuriramo i eliminišemo tog kandidata.

2.3.3 Faza III: Uključivanje heuristike

Ovde prolazimo kroz sva mesta sa aktivnim kandidatima dva puta. U prvom prolazu, posmatramo samo one sa četiri preostala kandidata, eliminišemo onog sa

najviše konflikata i donosimo sve odluke kao u Fazi II. Tokom drugog prolaza, radimo isto za mesta koja još uvek imaju tri aktivna kandidata. Zatim preostali problem (sastavljen samo od mesta sa tačno dva aktivna kandidata) predajemo 2-SAT.

3 Metaheuristike

3.1 Genetski algoritam

Jednostavan genetski algoritam, prva varijanta rešenja za koju je najveći problem pravila kompleksnost fitnes funkcije. Ovaj algoritam se slabo pokazao čak i na malom broju tačaka.

3.1.1 Hromozom

Hromozom se sastoji trenutne veličine kvadrata, kao i niza proto kvadrata koji u sebi čuvaju orijentaciju i početnu tačku.

3.1.2 Mutacije

Mutacija se dešava sa verovatnoćom 0.01%. U slučaju da treba da se desi mutacija proto kvadrata, promeni mu se orijentacija, a u slučaju da treba da se mutira veličina svih kvadrata, ona se ili povećava (sa verovatnoćom 75%) ili smanjuje za jednu poziciju u nizu mogućih veličina.

3.1.3 Selekcija, elitizam i ukrštanje

Selekcija je turnirska, sa podrazumevanom vrednošću 5, a takodje je implementiran elitizam sa podrazumevanom vrednošću 20%. Implementirano je jednopoziciono i uniformno ukrštanje.

3.1.4 Fitnes funkcija

Prolazi se kroz kombinacije parova niza proto kvadrata, i broji se koliko ima parova pre prvog preseka nekog para. Nakon toga se ovaj postupak ponovi od kraja niza parova, i dobijena dva broja se saberu i pomnože sa trenutnom veličinom.

3.2 Poboljšani genetski algoritam

Ovaj algoritam je nadogradnja na prethodni, većina podrazumevanih vrednosti i pomoćnih metoda su mu ostale iste. Razlikuje se doduše u konstrukciji hromozoma kao i fitnes funkcije.

3.2.1 Hromozom i mutacija

Ovde je hromozom pojednostavljen i sastoji se samo od niza orijentacija. Prilikom mutacije određena mutacija se samo promeni nekom drugom.

3.2.2 Fitnes funkcija

Glavna razlika ova dva algoritma je u fitnes funkciji. Naime, ovde se pronalazi najveća moguća veličina kvadrata koja može da se postavi pri trenutnom nizu orijentacija. Pronalazak je implementiran uz pomoć binarne pretrage po mogućim veličinama kvadrata.

Takodje, prilikom predprocesiranja se za svaku tačku pronadu sve one tačke koje su joj bliže od dvostruke najveće moguće veličine kvadrata. Tako da, umesto da se proverava da li svake dve tačke imaju presek pri datoj orijentaciji i veličini kvadrata, proveravaju se samo one koje su blizu jedna drugoj.

3.2.3 Fitnes

4 Rezultati

Tabela 1: Rezultati

number of points	seed	B size	B elapsed	Genetic size	Genetic elapsed	Improved Genetic size	Improved Genetic elapsed
5	0	35.3020	0.0002	22.7321	0.1100	35.3020	0.0846
5	1	42.1144	0.0002	39.7943	0.1108	42.1144	0.0870
5	2	61.3179	0.0011	42.9225	0.1371	61.3179	0.2088
5	3	36.9589	0.0009	25.8712	0.1028	36.9589	0.2080
5	4	69.8648	0.0008	48.9053	0.1003	69.8648	0.2202
5	5	59.3896	0.0008	41.5728	0.1451	59.3896	0.1914
5	6	49.8109	0.0001	48.4583	0.1138	49.8109	0.0928
5	7	47.7883	0.0008	33.9297	0.1069	47.7883	0.2132
5	8	55.6683	0.0009	39.5245	0.1361	55.6683	0.1998
5	9	53.5835	0.0009	38.0443	0.1041	45.6572	0.1721
10	0	20.1374	0.0027	14.6017	0.2650	19.4759	0.6594
10	1	26.8340	0.0004	17.0165	0.2730	26.8340	0.1957
10	2	22.4876	0.0037	13.6350	0.2128	19.4009	0.4560
10	3	30.9898	0.0038	15.4949	0.2719	30.9898	0.6949
10	4	29.5408	0.0035	29.5408	0.2752	29.5408	0.6761
10	5	32.0455	0.0035	16.7846	0.2813	32.0455	0.5254
10	6	24.4790	0.0034	16.0045	0.2291	21.6525	0.6013
10	7	25.3977	0.0030	16.2122	0.2888	25.3977	0.5525

Continued on next page

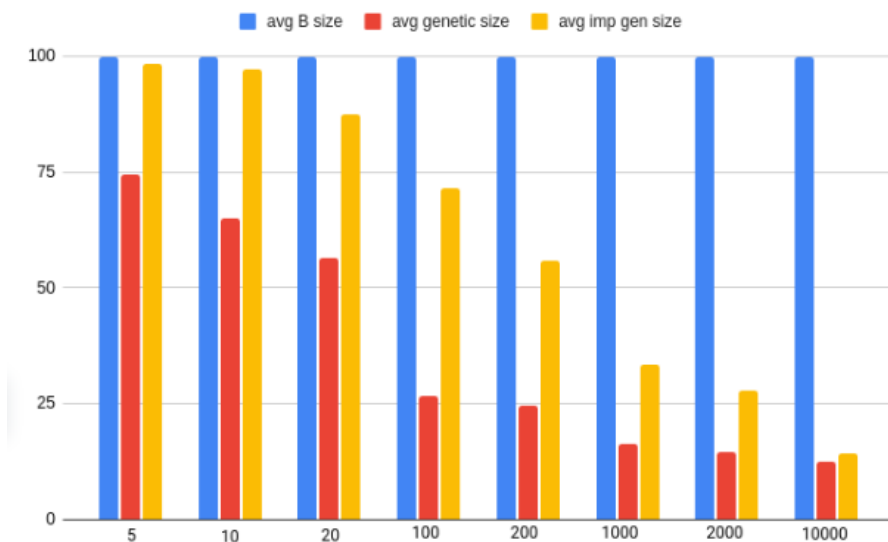
Tabela 1: Rezultati (Continued)

number of points	seed	B size	B elapsed	Genetic size	Genetic elapsed	Improved Genetic size	Improved Genetic elapsed
10	8	21.9025	0.0039	13.1415	0.3081	21.9025	0.4996
10	9	26.8072	0.0031	17.3233	0.2836	26.2364	0.5581
20	0	14.8317	0.0109	8.2715	0.6064	10.4326	1.2658
20	1	17.7698	0.0124	10.0212	0.5272	17.0165	1.0740
20	2	13.1543	0.0148	7.5067	0.6153	12.5291	0.8135
20	3	13.4754	0.0133	12.3385	0.6919	12.6793	1.1250
20	4	12.9604	0.0147	7.5211	0.6460	8.9035	0.8270
20	5	20.0668	0.0110	5.2276	0.5985	16.7846	1.2776
20	6	12.2823	0.0115	7.4243	0.5285	10.9630	1.9486
20	7	14.2098	0.0116	6.1811	0.6022	11.1956	1.3397
20	8	12.9753	0.0135	8.4649	0.6453	12.9753	1.2777
20	9	15.3393	0.0141	10.2480	0.5252	15.3393	1.1358
100	0	5.2101	0.3394	1.0167	3.8348	2.8310	5.4570
100	1	4.8196	0.3330	0.8188	4.0033	3.7424	8.6991
100	2	5.3345	0.3498	1.4050	4.1928	3.1100	5.3395
100	3	2.5173	0.4301	1.0134	4.1222	2.5173	3.9557
100	4	5.3397	0.3273	1.4967	4.0970	2.9960	5.8460
100	5	3.8286	0.0436	1.3737	4.1538	3.2918	2.6242
100	6	3.9952	0.4073	1.2149	4.0909	3.3872	6.6124
100	7	4.8685	0.3644	1.2188	4.0041	3.2676	6.1052
100	8	4.5989	0.3733	1.2414	4.0048	3.5007	6.1725
100	9	3.6960	0.4050	0.9395	3.8809	3.0161	4.2277
200	0	3.0682	1.6054	0.7022	10.4813	1.8340	12.8923
200	1	3.6387	1.3792	0.6034	9.9041	1.2897	11.4101
200	2	2.2561	1.7651	0.4451	12.6591	1.5089	14.4704
200	3	2.5173	1.5510	0.3319	10.8633	1.5361	6.6585
200	4	2.4096	1.9593	0.7761	10.2350	1.7333	10.8867
200	5	2.7550	1.8628	1.0367	10.3626	1.7471	8.6153

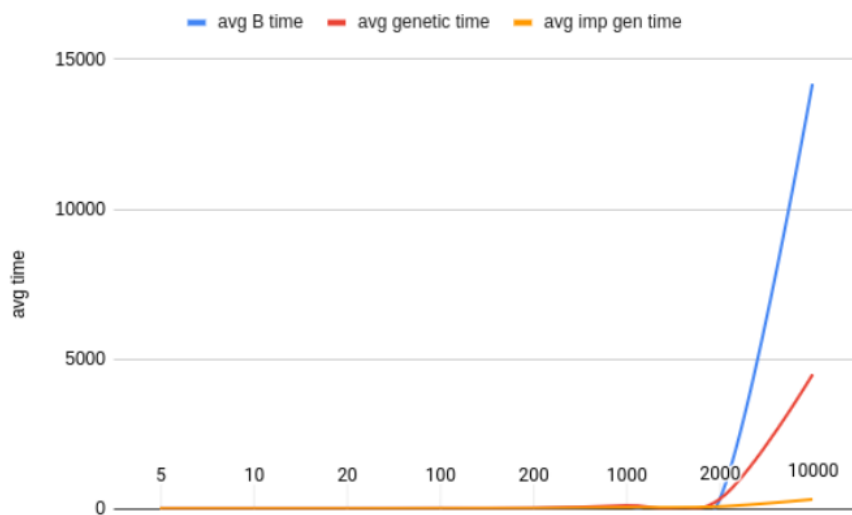
Continued on next page

Tabela 1: Rezultati (Continued)

number of points	seed	B size	B elapsed	Genetic size	Genetic elapsed	Improved Genetic size	Improved Genetic elapsed
200	6	2.5865	0.1750	0.8431	10.6476	1.4087	5.5453
200	7	3.8616	1.6255	0.9269	10.3298	1.9366	10.0482
200	8	3.4888	1.6036	0.7183	10.4093	1.3351	11.2543
200	9	2.6795	1.8910	0.8107	10.6949	2.0250	9.5496
1000	0	1.1287	62.4307	0.2546	60.5390	0.3597	39.9337
1000	1	1.0390	60.1160	0.1624	123.8743	0.3516	39.5442
1000	2	1.1521	65.5126	0.1034	89.7775	0.3722	34.6695
1000	3	1.2391	4.4660	0.1416	78.1544	0.3069	34.6919
1000	4	0.4746	78.0351	0.1222	104.4416	0.3510	24.7008
1000	5	1.1983	52.1492	0.1587	92.3268	0.3777	45.1977
1000	6	0.9802	56.8678	0.2834	88.9586	0.3884	31.8742
1000	7	0.9750	69.3481	0.1155	110.8884	0.3253	42.0070
1000	8	1.1280	63.5581	0.1970	70.7388	0.3167	41.9968
1000	9	1.1687	58.0488	0.1677	100.7930	0.3446	37.9643
2000	0	0.5902	334.1976	0.0421	335.1482	0.1752	66.2694
2000	1	0.6949	325.7983	0.0872	241.0173	0.1659	67.1866
2000	2	0.9119	308.9079	0.2267	153.4743	0.1872	81.1795
2000	3	0.6945	380.5133	0.1180	316.2370	0.1708	74.4366
2000	4	0.3922	36.8053	0.1118	224.0539	0.2139	31.0206
2000	5	0.4914	30.3630	0.1513	292.9607	0.1968	38.3861
2000	6	0.6693	337.7248	0.0452	349.7639	0.1860	71.6278
2000	7	0.7234	294.3216	0.0613	369.5315	0.2141	82.7554
2000	8	0.5690	366.0344	0.0631	326.8942	0.1356	57.5886
2000	9	0.8197	297.6410	0.0519	332.8558	0.1697	74.9166
10000	0	0.2859	17301.5453	0.0564	1694.4324	0.0379	317.9344
10000	1	0.2854	1065.0139	0.0134	5370.4632	0.0284	264.2074
10000	2	0.1736	21978.9608	0.0407	2954.1380	0.0360	383.1619
10000	3	0.2518	16313.0537	0.0150	7801.0702	0.0385	256.0877



Slika 3: Zadovoljivost rezultata



Slika 4: Prosečno vreme

5 Zaključak

Popravljeni genetski algoritam se ponaša prihvatljivo pri manjem broju tačaka, ali sa porasom mu opada i preciznost. Sa druge strane vidimo da vreme izvršavanja B algoritma znatno brže raste, i da je prosečno vreme izvršavanja za 10000 tačaka par redova veličine iznad popravljenog genetskog algoritma.

Rezultati ovog rada, što se tiče genetskog algoritma, su dobijeni sa populacijom veličine 100 kroz 50 generacija, tako da je moguće u budućnosti proveriti da li bi algoritam bio precizniji za veći broj generacija i/ili populacije. Obzirom na vremensku neefikasnost B algoritma, pri većem broju tačaka, ima mesta za povećanje kapaciteta popravljenog genetskog algoritma, uz to da ostane idalje brži.

Literatura

- [1] Wagner, F., and Wolff, A. (1995), *An efficient and effective approximation algorithm for the map labeling problem* Proc. 3rd Ann. European Symp. on Algorithms, Lecture Notes in Comput. Sci. 979, Springer-Verlag, 420-433.