UNIVERZA V LJUBLJANI FAKULTETA ZA MATEMATIKO IN FIZIKO

Finančna matematika

Marcel Blagotinšek, Peter Milivojević

Maximum number of edges in a connected graph with n vertices and diameter d

Skupinski projekt Poročilo

Advisers: doc. dr. Janoš Vidali, prof. dr. Riste Škrekovski

1. Navodilo naloge

A connected graph with diameter d on n vertices with the minimal number of edges will be a tree and henceforth, it will have n-1 edges. It will be harder to answer which graphs on a fixed number of vertices n and fixed diameter d have the maximal number of edges. We want to analyse the structure of such graphs. So, for a fixed number of vertices n and a fixed diameter d, when these two values are small, apply an exhaustive search. Next, for larger n and d, apply some metaheuristic. Try to obtain some specific properties of these graphs. Verify for how large n and d your exhaustive search and your metaheuristic implementations are efficient.

2. Opis problema

Želiva poiskati povezane grafe na n vozliščih s premerom d, ki bodo imeli maksimalno število povezav. Najin cilj je, na podlagi testiranja oz. generiranja, pridobiti kar se da dober vpogled v strukturo teh grafov in posledično ugotoviti, če za njih veljajo kakšne posebne lastnosti. Za majhne vrednosti n in d, se bova problema lotila z generiranjem grafov, za večje pa bova uporabila metodo simulated annealing. Ugotavljala bova tudi efektivnost najinih metod v odvisnosti od vrednosti n in d.

3. Potek Dela

Ideja prve faze projekta t.i. exhaustive search-a je, da z generiranjem vseh možnih povezanih grafov na n vozliščih s premerom d, poiščeva tiste, ki imajo maksimalno število povezav. To bova počela za majhne vrednosti n in d. Kako majhne, bo odvisno od časovne zahtevnosti samega algoritma, kajti je pričakovati, da bo že pri ne malo od 5 večjih vrednostih n algoritem počasen. Na podlagi generiranja grafov za različne n in d bova poskušala ugotoviti kakšne lastnosti, tako strukturne kot vizualne, lahko pripiševa tem grafom. Naraščanje/padanje števila povezav v odvisnosti od števila vozlišč oz. premetra bova prikazala tudi s pomočjo grafa, ki se bo morda obnašal podobno kot kakšna znana funkcija, kar bo vsekakor pomagalo pri oceni števila povezav za večje vrednosti n in d. Kot omenjeno bova poskusila najti kakšno formulo za maksimalno število povezav pri številu vozlišč n in premeru d. Tako pridobljene formule, četudi bo morda držala, ne bova dokazovala in jo bova posledično uporabila kot oceno v primeru generiranja grafov. Na koncu bova poleg ugotovitev glede lastnosti grafov v poročilu zapisala tudi pri kako velikih vrednostih n in d je najin algoritem prenehal učinkovito delovati. V drugi fazi projekta se bova problema lotila z metahevristično metodo simulated annealing. Začela bova z nekim začetnim povezanim grafom G, ki bo ustrezal pogojem n in d, nato pa bova dodala povezavo iz množice povezav komplementa grafa G. V kolikor bo premer grafa G+eostal isti, imamo nov graf, ki ima isti premer vendar povezavo več. Ce bo premer novega grafa manjši od d, pa bova poiskala vozlišči u in v na maksimalni razdalji in odstranjevala povezave iz poti med u in v toliko časa, dokler ne bo premer spet d. Seveda se lahko zgodi, da bo premer večji od d, takrat pa bova spet poiskala vozlišči u in v na maksimalni razdalji in dodajala neke povezave na poti med u in v toliko časa, dokler ne bo premer spet d. Povezave bova morala dodajati med ustreznimi vozlišči. Torej, če bo nov premer d-1, bova dodala povezavo med vozliščema na oddaljenosti 2. Pri tem se zavedava, da z neko verjetnostjo v nekem koraku vzameva graf z manj povezavami, ki pa je morda boljše izhodišče za naprej. Tudi tukaj bova začela na manjših vrednostih, in s tem preveriva, če najin algoritem deluje, nato pa n in d povečujeva. Tudi v drugi fazi projekta bova pozorna na efektivnost oz. časovno zahtevnost, ter bova ugotovitve glede tega zapisala v poročilu. Algoritme in programe bova v obeh fazah pisala v CoCalc Jupyter notebook-u.

4. Koda

4.1. 1. FAZA - KODA:

```
from sage.graphs.graph_generators import graphs
       def najdi_graf_z_premerom(n, d):
           # Najvecje stevilo povezav in graf z najvec povezavami
           max_povezave = 0
6
           graf_z_max_povezav = None
           # Zanka po vseh povezanih grafih z n vozlisci, ki jih
8
               generiramo z uporabo nauty_geng().
           for G in graphs.nauty_geng(str(n) + "_{\sqcup}-c"):
9
               # Premer grafa.
11
               premer = G.premer()
12
13
               # Ce srecamo graf katerega premer je enak nasemu
14
                    premeru d
               if premer == d:
                    # zabelezimo stevilo povezav
16
                    stevilo_povezav = G.size()
18
                    # Ce je stevilo povezav vecje od trenutnega
19
                        maksimuma, posodobi maksimum.
                    if stevilo_povezav > max_povezave:
2.0
                        max_povezave = stevilo_povezav
21
                        graf_z_max_povezav = G.copy()
22
23
           return graf_z_max_povezav, max_povezave
24
25
       # Primer za neko stevilo vozlisc n in premer d.
26
       n = 8
27
       d = 3
28
       # Poiscemo povezan graf z dolocenim stevilom vozlisc in
30
           premerom, ki bo imel maksimalno stevilo povezav.
       graf_z_max_povezav, max_povezave = najdi_graf_z_premerom(n
31
           , d)
```

```
# Ce je graf najden ga prikazemo
34
       if graf_z_max_povezav:
35
            36
                 max_povezave}_povezavami:")
            print(graf_z_max_povezav)
37
38
            graf_z_max_povezav.show()
39
       else:
40
            print(f"Graf_{\sqcup}z_{\sqcup}\{n\}_{\sqcup}vozlisci_{\sqcup}in_{\sqcup}premerom_{\sqcup}\{d\}_{\sqcup}ni_{\sqcup}bil_{\sqcup}
41
                najden<sub>□</sub>.")
42
43
        import pandas as pd
44
       import matplotlib.pyplot as plt
45
46
       rezultati = []
47
48
       # Zanka za preiskovanje razlicnih kombinacij n in d, grafe
49
             z maksimalnim stevilom povezav shranjujemo v slovar
       for n in range(1, 10):
            for d in range(1, n):
                 graf_z_max_povezav, max_povezave =
                     najdi_graf_z_premerom(n, d)
                 rezultat_slovar = {
53
                     'n': n,
54
                     'd': d.
                     'max_povezave': max_povezave
56
                 }
57
                 rezultati.append(rezultat_slovar)
58
59
       # Prikazemo rezultate s tabelo
       df = pd.DataFrame(rezultati)
61
       print(df)
63
       # Prikazemo tudi graf, ki predstavlja maksimalno stevilo
65
            povezav v odvisnosti od d za razlicne n
       plt.figure(figsize=(10, 6))
66
       for n in range(1, 9):
            podskupina = df[df['n'] == n]
68
            plt.plot(podskupina['d'], podskupina['max_povezave'],
69
                 label=f'n={n}')
       plt.xlabel('premer<sub>□</sub>(d)')
70
       plt.ylabel('maksimalnousteviloupovezav')
71
       plt.legend()
72
       plt.title('max_povezave(d)_{\sqcup}za_{\sqcup}razlicne_{\sqcup}n')
73
       plt.show()
74
```

4.2. 2. FAZA - KODA:

42

```
import networkx as nx
2
       import matplotlib.pyplot as plt
       import random
       from itertools import combinations
4
       import math
       def spodnja_meja(n, d):
       if d >= n:
9
            return 'Izbrani u premer u je u prevelik'
10
       elif d < 1:
11
12
            return 'Izbrani premer je premajhen'
       elif d == 1:
13
           G = nx.complete_graph(n)
14
            return G
15
       else:
16
           G = nx.complete_graph(n - d + 1)
17
            new_node = n - d + 1
18
            for i in range(n - d):
19
                existing_node = i
20
                G.add_edge(new_node, existing_node)
21
            if d > 2:
22
                for i in range (n - d + 2, n):
23
                    new nodes = i
24
                    G.add_edge(new_nodes, new_nodes - 1)
25
            return G
26
27
2.8
       # Presteje stevilo povezav v grafu.
29
       def ciljna_funkcija(graf):
30
            return len(graf.edges)
31
32
33
       # Najde najkrajso mozno pot v grafu med zacetnim in
34
            koncnim vozliscem.
       def najdi_pot(graf, zacetek, konec):
35
            try:
36
                pot = nx.shortest_path(graf, source=zacetek,
37
                    target = konec)
                return pot
            except nx.NetworkXNoPath:
                return None
41
```

```
# Kot argument sprejme graf ter pot iz katere zelimo
43
           odstranit povezavo, nato iz nje nakljucno odstrani
           povezavo.
       def odstrani_nakljucno_povezavo_iz_poti_v_grafu(graf, pot)
           nakljucni_indeks_povezave = random.randint(1, len(pot)
                - 1)
           povezava_za_odstranitev = (pot[
               nakljucni_indeks_povezave - 1], pot[
               nakljucni_indeks_povezave])
           graf.remove_edge(*povezava_za_odstranitev)
47
           return graf
48
49
       # METAHEVRISTICNI ALGORITEM
       def simulirano_hlajenje_2_povezavi_razmaka_spodnja_meja(n,
            max_iteracij, zacetna_temperatura, stopnja_hlajenja,
           premer):
           trenutna_resitev = spodnja_meja(n, premer)
           najboljsa_resitev = trenutna_resitev.copy()
54
           temperatura = zacetna_temperatura
56
           for iteracija in range(max_iteracij):
               # Preverimo kaksen je premer, bodisi je vecji ali
58
                   enak premeru trenutne resitve, bodisi pa je
                   manjsi od 1. V zadnjem primeru je torej
                   nepovezan graf. Dodamo povezavo.
               if premer <= nx.diameter(trenutna_resitev) or nx.</pre>
59
                   diameter (trenutna resitev) < 1:
                   # Izbere 2 nakljucni vozlisci, ki nista
60
                       povezani.
                   vozlisce1 = random.choice(list(
61
                       trenutna_resitev.nodes))
                   vozlisca_2_razmaka = [vozlisce for vozlisce in
                        trenutna_resitev.nodes - set([vozlisce1])
                        if nx.shortest_path_length(
                       trenutna_resitev, source=vozlisce1, target
                       =vozlisce) == 2]
                   vozlisce2 = random.choice(vozlisca_2_razmaka)
63
                   # Dodamo povezavo med izbranima vozliscema.
64
                   nova_resitev = trenutna_resitev.copy()
                   nova_resitev.add_edge(vozlisce1, vozlisce2)
                   # Preverimo ali je nov graf tak, da ima vec
67
                       povezav. V primeru da je to res,
                       posodobimo najboljso resitev, sicer pa z
                       verjetnostjo izberemo ali bomo posodobili
                       trenutno resitev ali ne.
```

```
# Opomba: Lahko pride do izbire "slabsega"
68
                        grafa, upamo, da nas bo ta "slabsi" vseeno
                         pripeljal do boljse resitve v
                        nadaljevanju.
                   delta = ciljna_funkcija(nova_resitev) -
69
                        ciljna_funkcija(trenutna_resitev)
                   if (delta > 0 and premer <= nx.diameter(</pre>
70
                        nova_resitev)) or random.random() < math.</pre>
                        exp(-delta / temperatura):
                        trenutna_resitev = nova_resitev.copy()
71
                        if ciljna_funkcija(nova_resitev) >
72
                            ciljna_funkcija(najboljsa_resitev) and
                             premer == nx.diameter(nova_resitev):
                            najboljsa_resitev = nova_resitev.copy
73
               else:
74
                   # Poiscemo kombinacije vozlisc z najvecjo
75
                        ekscentricnostjo.
                   ekscentricnosti = nx.eccentricity(
76
                        trenutna_resitev)
                   max_ekscentricnost = max(ekscentricnosti.
77
                       values())
                   vozlisca_z_max_ekscentricnostjo = [vozlisce
78
                        for vozlisce, ekscentricnost in
                        ekscentricnosti.items() if ekscentricnost
                       == max_ekscentricnost]
                   kombinacije_parov = list(combinations(
79
                        vozlisca_z_max_ekscentricnostjo, 2))
                   rezultat = []
                   # Izberemo najbolj oddaljeni vozlisci.
81
                   for i, j in kombinacije_parov:
82
                        potencialen_rezultat = najdi_pot(
83
                            trenutna_resitev, i, j)
                        if potencialen_rezultat and len(
84
                            potencialen_rezultat) > len(rezultat):
                            rezultat = potencialen_rezultat
85
                   nova_resitev = trenutna_resitev.copy()
                   nova_resitev =
                        odstrani_nakljucno_povezavo_iz_poti_v_grafu
                        (trenutna_resitev.copy(), rezultat)
                   # Preverimo ali je nov graf tak, da ima vec
88
                        povezav. V primeru da je to res,
                        posodobimo najboljso resitev, sicer pa z
                        verjetnostjo izberemo ali bomo posodobili
                        trenutno resitev ali ne.
                   # Opomba: Lahko pride do izbire "slabsega"
89
                        grafa, upamo, da nas bo ta "slabsi" vseeno
                         pripeljal do boljse resitve v
                        nadaljevanju.
```

```
delta = ciljna_funkcija(nova_resitev) -
90
                         ciljna_funkcija(trenutna_resitev)
                    # IF pogoj je vedno izpolnjen, pustimo ga
91
                         zgolj za voljo testiranja.
                    if ((delta > 0 and premer <= nx.diameter(</pre>
92
                         nova_resitev)) or random.random() < math.</pre>
                         exp(-delta / temperatura)) and nx.
                         is_connected(nova_resitev):
                         trenutna_resitev = nova_resitev.copy()
93
                         if ciljna_funkcija(nova_resitev) >
94
                             ciljna_funkcija(najboljsa_resitev) and
                              premer == nx.diameter(nova_resitev):
                             najboljsa_resitev = nova_resitev.copy
95
                                 ()
                # Znizamo(ohladimo) temperaturo po stopnji
96
                    hlajenja.
97
                temperatura *= stopnja_hlajenja
98
            return najboljsa_resitev
99
       # Prikaz delovanja algoritma na primeru.
101
       st_vozlisc = 30
102
       max_iteracij = 1000
103
       zacetna_temperatura = 1.0
104
       stopnja_hlajenja = 0.95
105
       zeljen_premer = 7
106
       najboljsi_graf_s_m =
108
            simulirano_hlajenje_2_povezavi_razmaka_spodnja_meja(
            st_vozlisc, max_iteracij, zacetna_temperatura,
            stopnja_hlajenja, zeljen_premer)
       print (f"Steviloupovezavuvunajboljsemugeneriranemugrafu:u{
109
            ciljna_funkcija(najboljsi_graf_s_m)}")
110
       # Graf se prikazemo.
       plt.figure(figsize=(8, 8))
112
       nx.draw(najboljsi_graf_s_m, with_labels=True, font_weight=
113
            'bold', node_color='skyblue', node_size=800, font_size
            =10)
       plt.show()
114
```

5. Ugotovitve

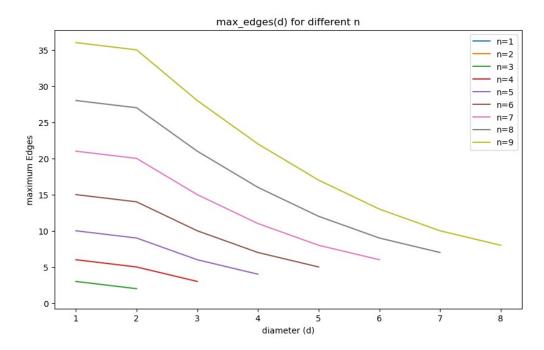
5.1. 1. FAZA - UGOTOVITVE:

Za d=1 ugotovimo, da je ne glede na izbiro števila vozlišč n, iskani graf ravno polni graf in ima posledično $\frac{n(n-1)}{2}$ povezav. V nasledjem koraku hitro ugotovimo, da se pri d=2 število povezav zmanjša le za 1, saj se z odstranitvijo katere koli poljubne povezave v polnem grafu premer poveča na d=2 in ker smo za to potrebovali odstraniti le eno samo povezavo je največje možno število povezav v grafu z n točkami in premerom d=2 enako $\frac{n(n-1)}{2}-1$. Podobno opazimo, da so grafi za premere d=n-1 ravno drevesa s stopnjo 2 in je zato število povezav enako n-1. Tako nas pri dani nalogi v resnici zanimajo predvsem grafi z $d\in\{3,\ldots,n-2\}$. V prvi fazi sva pričela reševati z opazovanjem in računanjem grafov z manjšim številom vozlišč n, pri tem sva si pomagala tudi s kodo iz 4.1.

Napisani algoritem je z uporabo funkcije nauty geng generiral vse povezane grafe na n vozliščih, izločeval tiste, katerih premer ni bil enak d, ter posodabljal spremenljivko z maksimalnim številom povezav ter tem povezavam ustreznemu grafu. Podatke o največjem možnem številu povezav za grafe do 10 vozlišč sva zbrala v tabeli 1 in vrednosti prikazala na spodnjem grafu.

$\mathbf{n} \setminus \mathbf{d}$			Šte	evilo	po	veza	av		
II \u	1	2	3	4	5	6	7	8	9
2	1								
3	3	2							
4	6	5	3						
5	10	9	6	4					
6	15	14	10	7	5				
7	21	20	15	11	8	6			
8	28	27	21	16	12	9	7		
9	36	35	28	22	17	13	10	8	
10	45	44	36	29	23	18	14	11	9

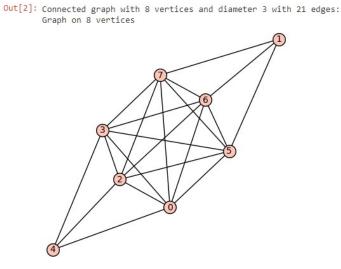
Tabela 1. Število povezav glede na število vozlišč in premer.



SLIKA 1. Maksimalno število povezav v odvisnosti od d pri različnih n.

Z opazovanjem tabele sva na podlagi vzorca uspela za grafe zd>1 zapisati formulo, ki nama pove maksimalno število povezav v grafu z n vozlišči in premerom d: $\frac{(n-d+1)(n-d)}{2}+n-2$. Te formule ne bova dokazovala in jo bova v nadaljnem raziskovanju uporabljala kot oceno, saj vanjo brez dokaza ne moreva biti popolnoma prepričana.

Z nadaljnim opazovanjem generiranih grafov sva opazila, da vsi grafi vsebujejo poln podgraf velikosti n-d+1. Na grafu prikazanem spodaj se to lepo vidi.



SLIKA 2. Graf z 8 vozlišči in premerom 3, ki vsebuje poln podgraf velikosti 6.

Glede učinkovitosti sva ugotovila, da je najin algoritem učinkovit za grafe z številom vozlišč do 9. Od tod naprej traja enostavno preveč časa. Že pri številu vozlišč enako 8, je povezanih grafov kar 251548592

$5.2.\ 2.\ \text{faza}$ - Testiranje:

	, [Štev	ilo po	oveza	v																				_	\neg
l n	٩þ	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
n	0 1 1 2 2 3 3 1 1 1 1 2 2 3 3 3 3 3 3 3 3	1 3 6 10 15 15 22 36 45 55 66 67 89 1105 120 136 153 171 190 221 223 1253 227 3325 335 435 435 435 435 435 435 43	2 5 9 14 20 27 35 44 54 65 77 90 104 119 135 152 275 2275 2275 324 350 377 405 434 495 527	3 6 10 15 21 28 36 45 55 66 78 91 105 136 153 171 190 231 253 300 325 351 378 406 435 495 496	4 7 11 16 22 29 37 46 67 79 92 103 121 133 154 172 179 203 226 294 343 364 392 450	5 8 12 17 23 30 38 47 53 65 76 93 107 116 126 155 146 192 207 219 237 220 223 281 294 286 363 390 390 390 390 390 390 390 390 390 39	6 9 13 18 23 30 31 48 46 63 58 63 90 107 133 144 169 123 213 146 172 222 194 321 241 241 256	7 10 14 19 22 30 39 49 47 52 58 63 82 99 104 157 147 148 156 173 151 168 186 248 231 239	8 11 15 20 23 29 41 38 55 61 62 68 87 75 87 99 105 112 174 135 190 163 175 190 163 175 184 202 203 203 203 203 203 203 203 203 203	9 12 16 20 27 32 45 45 57 61 62 75 72 100 111 140 124 149 159 152 147 178 208	10 13 17 20 24 35 34 42 43 56 69 64 74 91 113 127 115 145 178 191 191	111 14 17 21 29 32 39 51 50 77 68 69 73 92 123 134 137 122 124 209 172	12 15 19 24 27 31 37 40 57 57 62 69 79 90 88 88 102 109 133 134	13 16 20 23 26 31 42 45 49 54 57 58 77 93 81 96 104 135 132	14 17 21 28 31 39 49 56 62 71 67 79 87 99 90 118 107	15 18 22 27 28 32 35 43 44 69 67 76 71 89 103 100	16 19 22 26 31 37 41 41 50 51 54 66 74 82 89 85	17 20 23 27 30 41 44 46 60 63 65 71 90 85	18 21 24 28 31 36 39 48 47 58 57 59 75 76 100	19 22 25 28 37 42 44 50 58 56 65 74 77	20 23 26 30 33 37 44 43 51 53 56 66 66 66	21 24 27 30 34 39 45 47 60 60 66	22 25 25 28 32 35 39 49 53 63	23 26 29 33 36 40 42 47 54 57	24 27 30 33 33 40 43 47 60	25 28 31 34 38 41 47 47	26 29 3 2 36 41 41 46	27 30 33 37 42 43	28 31 34 37 41	29 32 35 38	30 33 36	31 34	32		34	35	36	37	38	39	40	41	42	43	44
3	2 4	196 528 561	495 527 560	465 496 528	412 450 489	363	281	231 239 261	184 202 224	178 208 187	$191 \\ 211 \\ 205$	209 172 145	133 134 204	135 132 120	118 107 117	100 100 117	8.5	85 86 92	76	74	66	60 66 72	58 63 67	54	4.7	47	41	42	37	35 38 42	33		3 2 3 5	33											
3 3 3 3	3 6 7 6	59 5 53 0 56 6 70 3	629	630	480 547 579 599	399 394 390	328 295 435 363	281 288	219 239 245		187 192 195 215	157 203	138 158 196	138 163 147	139 135 138	123 136 133	107 114 123	97 108 121	91 98 100 111	79 89 108 119	78 78 96 99	83 79 80 90	72 76 82	65 69 74	62 69 77	60 60 78 71	54 60 62	52 55 63	48 51 54	45 49 51 59	43 46 50 53	41 44 49 50	38 41 45 50	36 40 43 45	40	35	36	27							
3 4 4 4) :	741 780 820	740 779		650 656 695 722	514 530 493 564 561	360 480 489 459	330 318 356	306 277 304 316 351	219 259 256 283 271	218 283 342 296	219 204	257 200 237 221	151 178 176 212 194	164 159 224 162 184	133 156 176 178 186	132 133 145 147 165	177	118	106 117	108 117 126 134	108	88 104 103 108 103	81 90 89 99 103	83 81 88 91 100	74 85 99 101	64 79 82 81 90	65 77 90 95 85	64 64 67 71 80	59 60 62 71 72	59 64 67 74	54 58 64 67	51 63 61 68	48 57 55 60	43 47 49 54 57	41 44 48 51 54	39 42 46 49 51	43 46	38 41 45 47	39 42 45	40 43	41			
4 4	3 9	903 946	$\frac{902}{945}$	8 52 8 86	73.7	661 569	477 506	390 419	361 379	296 309	249 297 307	267 246	243 295 257	206 196	237 221	210 178	176 196	151	172 162	138	14 2 14 1	126 138	1 16 1 19	114 127 124	$^{105}_{102}$	91 105 109	97 98	91 96	83	8 2 8 6	86 74 82	71 75	69 69	63 70	60 64	58 61	54 57	53 56	52	48 51	46 49	44 47		43 46	44

Tabela 2. max iteracij = 1000, zacetna temperatura = 1.0, stopnja hlajenja = 0.99

n\d 1 2 1 3 3 4 6 5 10 6 15	l B	2	3	4	5	6	7	8	9	10	11	12	13	14	15																													
3 3 4 6 5 10	6	2							9	10	11	14	13	14	10	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
4 6 5 10	6	2																																									\neg	П
5 10			-																																									
		5 9	3	١, ١																																		-					-	
		14	6 10	7	5																																							
7 21		20	15	ú	8	6																																						
8 28		27	21	16	12	9	7																															- 1						
9 36		35	28	22	17	13	10	8																																				
10 45		44	36	29	23	18	14	11	9																													l						
11 55		54	45	37	26	24	17	15	12	10					i i		İ					ı				ĺ						İ	İ	İ	i		l i	i	ı	l		ı		ιi
12 66		65	55	44	33	31	25	19	15	13	11																											ı		- 1				
13 78		77	66	56	4.5	35	28	22	19	16	14	12																																
14 91		90	78	67	57	46	40	33	27	20	17	15	13	1.6																													-	
15 10 16 12		104 119	91 105	79 92	66 80	58 60	39 47	32 43	28 37	25 35	21 26	19 22	16 20	14	15																													
17 13			120	106	87	72	68	53	37	37	30	25	25	20	18	16																						- 1		- 1				
18 15					107	84	58	71	54	44	34	31	26	26	21	19	17																					1						
19 17	71 1	170	153	137	95	104	77	58	55	50	4.7	34	30	28	24	23	20	18				ı										l		l			l i	i	ı	ı		ı		ı
20 19			171		129	123	103	76	61	50	4.7	42	35	35	30	26	24	21	19			ı										l		l			l I	ı		ı		- 1		ıΙ
21 21			190	165	145	131	85	75	69	61	54	57	43	36	36	32	26	25	22	20																								
22 23	31 2		210	191	161	150	1 26	95	101	74	73	59	48	42	40	33	31	28	25	23	21																							
	53 2 76 2		231 253		162 190	156 163	115 123		99 82	77 85	63 77	58 71	4.7 54	49 50	46 46	40 52	34 41	32	28 3 2	27 30	24 27	22 25	23																					
	0 2				200	177	130		96	98	84	72	62	57	58	48	58	40	44	32	31	29	26	24																				
	25 3				200	194	134		121	104	94	80	84	74	55	49	45	48	41	45	33	32	29	27	25													l						
27 35			3 25		268		143		119	103	101	88	71	69	68	60	60	54	46	39	39	35	33	30	28	26												l		- 1				
	78 3				271		158		119	120		81	93	83	66	65	63	53	56	50	42	40	36	34	32	29	27					l		İ			l I	l		ı		ı		Ιİ
29 40					299	28 7		160	161	143	103	97	99	86	81	67	64	63	58	53	46	46	39	37	34	32	30											- [- 1			- 1	
30 43			406		318		262			143		100	91	110	78	68	68	67	74	55	49	54	43	4.2	38	36	33		29															
31 46 32 49			435 465		3 23 3 63	23 1 36 3	257 225		138 156	157 163	1 21 1 59	120 124	94 111	91 108	93 103	78 83	78 95	68 70	64 76	59 83	54 59	54 60	48	44	43 44	38	36 39	34	32	30	31													
	70 4 28 5				3 56		281		227	180		150	130	1 24	1103	89	84	81	80	69	62	68	62 56	54 57	48	51	46	41		37	34	32												
	51 5		528	485		336	244			262		195	154	1 24	108	106	99	90	95	78	75	71	64	58	52	52	49	45	44	39	37	35	33							- 1				
35 59	5 5	594	561		409	308	260	264	237	203	163	144	212	142	109	134	94	98	90	99	75	68	67	75	60	55	54	49	4.7	43	41	38	36	34										
36 63			595	546	391	444		239	203	216	168	155	174	152	136	110	1 21	100	8.5	89	92	82	67	69	66	65	59	50	49	45	44	43	39	37	35									
	6 6		630		467		286					170		145		119	113	105	104	91	93	81	77	68	70	84	59	58	54	50	47	45	42	40	38	36								
38 70	3 7	702	666		513	365	287					183	186	157	149	128	138	108	104	97	90	87	75	74	97	64	6.2	61	55	57	51	48	46	44	41	39	37							
	11 7 30 7		703 740		549 553		314				195 201	196		164	181			130		98	89 98	94	92	81 91	79	74	72	64 76	61 69	59 60	54 60	53	49 53		44		40 43		39				-	
	20 8				592		362			278		212	198 188	172 165		144 152	133 150	13 7 13 6		11 2 13 5	98 123	108 116	96 98	91	81 84	78 81	76 81	76	67	69		57 60	55	50 53	48 53	46 48	46	41		40				
42 86			815		512		370					220	224	195	196	174		14 5	138	152	118	104	96	99	97	84	83	78	72	72	67	67	64	57	54	52		48			41			
43 90					587		4 24					233		201	187	175		161	139	136	136	110	109	106	120	100	90	88	77	72	73	67	64	61	59	55	53	51			44	42		
44 94	16 9	145	890	794	633	431	404	4 19	358	303	263	254	217	213		195	177	148	152		123	119	10.7	124	97	92	86	92	83	77	77	76	67			58	56	54	51	49	4.7	45	43	
4.5 99	9 0	89	9.24	811	647	512	437	402	318	308	284	265	262	204	200	195	174	160	151	151	130	143	138	120	104	115	99	95	93	85	85	76	70	75	66	67	59	57	55	53	50	48	46	44

Tabela 3. max iteracij = 1000, zacetna temperatura = 1.0, stopnja hlajenja = 0.99

	_																			Štev	ilo n	oveza	v																					\neg
n\d	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
n\d 2 3 4 5 6 6 7 8 9 10 111 122 133 144 155 166 177 18 18 19 20 21 22 23 24 25 26 27 28 29 30 31 31 32 24	171 190 210 231 253 276 300 325 351 378 406 435 465	2 5 9 14 20 27 35 44 54 65 77 90 104 119 135 152 170 189 209 230 252 275 299 324 43 44 44 46 46 46 46 46 46 46 46 46 47 47 48 48 48 48 48 48 48 48 48 48 48 48 48	3 6 10 15 21 28 36 45 55 66 78 91 105 120 136 153 171 190 231 253 276 300 3 25 3 55 3 55 4 66 4 55 4 55 5 5 5 5 6 6 6 10 10 10 10 10 10 10 10 10 10 10 10 10 1	4 7 11 16 22 28 37 46 67 79 88 81 100 121 137 154 168 179 193 224 244 263 289 306 334 379 403 403 404 404 404 404 404 404 404 404	5 8 12 17 23 30 38 47 54 68 80 93 99 119 132 145 148 136 188 221 145 258 290 275 323 38 345	6 9 13 18 21 31 31 31 46 51 69 77 130 122 127 130 174 151 132 244 172 293 193 193 252 238	7 10 14 18 25 30 32 39 44 44 68 80 65 83 81 167 155 134 149 147 188 165 185 207 207 207 207 207 207 207 207 207 207	8 111 15 18 23 33 48 60 57 62 70 78 113 94 122 124 121 126 148 156 168 200 197	9 12 16 21 24 34 37 51 45 52 80 72 92 96 121 106 112 147 147 166 143 155	10 13 17 22 28 31 35 42 49 53 54 68 73 75 94 1300 110 124 128 129 182 161	11 14 18 23 25 30 34 42 48 56 69 71 81 77 89 2 102 102 131 133 157	12 15 19 22 27 44 44 49 58 63 79 85 103 105 121 126	13 16 20 23 31 32 41 43 51 60 52 70 104 74 76 92 102 102 112	14 17 21 24 27 33 36 42 47 50 66 63 64 77 77 92 91 108 115	15 18 21 25 28 32 47 64 66 69 66 78 97 101 104	16 19 22 26 38 36 38 47 49 54 62 59 71 79 80 95	17 20 24 27 32 33 41 53 51 55 64 70 96	18 21 25 27 33 34 42 43 53 51 57 60 84	19 22 26 29 32 38 38 43 59 61 55 70 66	20 20 23 26 30 34 38 41 44 49 54 60 70	21 24 27 30 34 45 56 58 61	222 255 288 311 440 399 411 505 60	23 26 29 32 36 41 44 45 51 53	24 27 31 36 37 41 44 50	25 28 31 35 37 42 46	26 29 32 35 41 44	27 30 33 37 44	28 31 35 38	29 32 35	30	31		33	34	35	36	37	38	39	40	41	42	43	44
3 2 33 34 35	496 528 561 595	495 527 560 594	465 496 528 561	417 448 472 478	345 397 344 420	238 304 325 352	$207 \\ 253 \\ 313 \\ 310$	197 203 238 211	155 172 194 182	161 171 198 242	1 57 1 76 1 79 1 61	124 152 138 153	112 120 149 159	115 116 123 132	10 4 9 4 11 3 11 2	95 99 103 136	83 90 93 105	84 98 89 97	66 90 84 80	70 85 73 77	61 70 70 88	60 60 74 73	53 60 61 75	50 58 56 62	46 49 51 67	44 47 50 62	44 43 49 51	38 41 43 49	$\begin{array}{c} 35 \\ 41 \\ 42 \\ 45 \end{array}$	33 36 39 43	34 37 43	38		34										
36 37 38 39 40 41	703 741 780	665 702	630 666 703 741	695	3 76 4 78 517 543 551 588	417		249	218 203 242 256 266 277	220 198 209 233 255 331	184 223	140 167 215 191 207 238	149 167 160 179 195 208	145 145 144 164 190 178	122 127 141 154 158 201	120 136 143	107 115 112 123 163 142	93 109 116 135 120 132	1 14 1 13	94 94 109 114 116	83 88 95 107 105 113		70 81 86 89 91 112	69 83 82 86 103	61 72 77 85 83 81	66 63 71 68 78 81	60 61 72 67 72 75	50 54 59 64 65 73	52 55 59 70	50 57 59 61	46 51 56 56	45 49 52	4 2 4 6 4 8 5 4	40 43 46 54	35 38 41 44 47 50	$rac{4}{4}rac{2}{5}$	43		39 42	40				
4 2 4 3 4 4 4 5	861 903 946 990	860 902 945	8 19 8 57	720 755 781	571 713 608 682	443 509 477	427 426 409	384 337	283 301 319	293 263 269 303	243 256 254	219 284 285 258	$212 \\ 221 \\ 197$	180 185 205 222	179 186 179 222	18 2 1 5 5 1 9 7 1 8 0	137 146 176	139 157 169 174	132 132 153	118 130 129 154	116 124 130 138	115 129 114	108 128 114 121	101 98 108 123	85 103 105 111	84 91 102 103	88 95 87 94	85 79	73 77 82	72 72 85	68 70 73	64 67 72	62 68 71	57 62	57 57 65	55 56 60	50 53 57	47 50 57	45 49 52	43 47 49	47		43 46	44

Tabela 4. max iteracij = 1000, zacetna temperatura = 1.0, stopnja hlajenja = 0.99

<u> </u>																					Šte	/ilo p	ovez	av																					_
n\d	1	2	3	- 4	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18	1 1 3 6 10 15 21 28 36 45 55 66 78 91 105 120 136 153	2 5 9 14 20 27 35 44 54 65 77 90 104 119	3 6 10 1 2 20 36 4 4 5 60 75 9 10	55 11 11 12 13 13 13 13 13 13 13 13 13 13 13 13 13	4 7 7 1 6 22 29 17 16 16 17 16 17 16 16 17	5 8 12 17 23 30 37 47 57 68 80 81 107	6 9 13 18 24 31 33 48 58 61 72 86	7 10 14 17 24 28 40 49 47 56 61	8 8 11 15 20 26 33 33 50 60 53	9 9 12 16 19 25 34 34 39 54	10 10 13 17 20 28 31 38 37	11 11 14 17 23 24 32 36	12 12 15 18 23 26 31	13 13 16 20 22 27	14 17 20 24	15 15 18 21	16 16 16 19	17	18	19					24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
19 20 21 22 23 24 25 26 27 28 29 30	171 190 210 231 253 276 300 325 351 378 406 435	170 189 209 230 252 275 299 324 350 377 405	15 17 19 21 23 25 27 30 32 35 37 40	3 13 13 14 15 16 17 18 18 18 18 18 18 18 18 18 18 18 18 18	37 54 70 91 11 24 48 77 95 19 31	116 115 146 142 192 208 167 220 258 302 259 3 28	108 87 139 156 162 139 208 147 205 225 208 270	87 95 87 103 113 123 143 167 157 176 193 179	59 96 72 79 97 91 123 147 141 142 162 157	51 59 59 80 96 126 99 149 174 157 170 154	50 53 69 74 74 105 118 115 106 137 148 182	42 53 48 57 73 73 69 85 92 102 107 119	40 44 49 55 64 65 75 78 90 104 97 113	31 38 38 53 61 60 69 70 82 89 86 98	29 35 47 46 49 53 60 76 93 75 81	25 28 33 37 46 50 52 60 71 67 70 75	23 28 31 37 37 40 51 50 57 66 78	20 24 27 31 34 39 44 47 49 72 62 65	18 21 24 30 34 35 40 48 49 53 59 62	19 22 25 29 32 38 40 51 48 53 59	20 23 26 32 35 41 40 48 51 52	21 24 27 31 33 37 48 46 51	22 25 28 31 35 38 42 45	23 26 30 32 36 40 45	24 27 30 34 36 43	25 28 32 34 38	26 29 32 36	27 30 33	28 31	29															
31 32 33 34 35 36 37 38 39 40 41 42 43 44 45	703 741 780 8 20 8 61 9 03 9 46	49 5 52 7 56 0 59 4 62 9 66 5 70 2 74 0 77 9 81 9 86 0 90 2	46 49 52 56 59 63 66 70 74 78 82 85 85	5 40 6 43 88 46 1 5: 5 5: 0 50 6 5' 3 6: 0 6' 0 70 90 7: 3 78	04 54 39 117 114 30 74 24 78 02 19 81	285 331 358 376 395 490 461 493 578 503 603 611 608	317 230 330 325 319 288 353 343 384 424 462 446 421 523	194 206 297 243 259 281 373 312 339 350 417 406 397	24 4 22 4 24 5 26 1 26 4 34 5 33 0 34 4 36 0 33 7	238 296 259 272 303 309 346	185 143 177 191 196 190 202 202 292 297 270 250	133 160 182 199 203 209 215 198 274 276 3 28 284	194 224 214 222	127 131 209 160 182 180 178 195 224 191 223	163 174 171 200 204 198	103 100 142 123 128 160 193 152 175 150 179 187	103 149 129 143 134 156 162 160 165 218	136 146 153 159 170	112 137 131 134 166 168	68 79 76 78 94 111 105 109 146 134 128 153 147	56 69 83 81 84 83 85 110 98 110 116 121 131 126	129 126 130 122 130	103 97 100 102 106 125	50 55 59 62 70 73 84 86 94 101 135 109	10.5	41 45 47 56 57 81 72 76 79 80 98 105 106	40 47 45 47 53 57 61 63 70 88 84 94 10 2 10 2	36 40 44 46 52 53 59 67 68 76 77 89 95	34 38 41 45 50 53 54 58 63 67 73 78 80 95	38 42 50 51 52 61 63 67 75 75 75	30 33 36 39 43 46 50 54 57 70 67 75 71	37 40 43 47 50 56 64 63 71 72	38 42 44 47 51 55 58 63 70 73	39 43 45 49 52 56 59 63 69	40 44 47 50 52 56 61 65	41 44 48 53 53 57 61	42 45 49 52 57 61	43 46 50 56	45 47 51 54	46 48 52			42 45 48	43 46	44

Tabela 5. max iteracij=1000,zacetna temperatura =1.0,stopnja hlajenja =0.99

Tabela 6. max iteracij = 1000, zacetna temperatura = 1.0, stopnja hlajenja = 0.99

_	. T																				Štev	ilo n	ovez:	ıv																				\neg
n\	1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19					24	25	26	27	28	29	30	31	32	33	34	35	36	37 3	8 3	9 40	0 4:	1 42	2 4 3	3 44
n\\ 1	11 12 33 44 55 66 57 98 11 11 11 11 11 12 22 22 22 33 33 33 33 33 33 33 33 34 34 34 34 34	15 55 66 78 91 05 20 36 53 71 90 10 31 53 76 00 25	2 5 9 14 20 27 35 44 65 77 90 104 119 135 152 170 230 230 2252 252 252 252 252 252 252 253 377	3 6 10 15 21 28 36 45 55 66 78 91 105 120 136 153 171 190 210 231 253 276 300 325 351	4 7 11 16 22 29 37 46 56 67 79 92 106 113 134 150 172 207 217 227 248	10 7 11 8 13 4 11 7 15 4 17 6 20 6 18 8 21 1 24 8 27 0	6 6 9 13 16 24 31 38 51 69 81 7 99 4 1 142 6 142	7 10 14 17 25 32 33 49 41 62 60 70 82 86 92 145 126 118 167 121 7160	116 110 129 128 171	9 12 15 20 22 28 33 41 44 72 59 63 67 78 118 103 104 176 151	10 13 17 20 24 31 35 46 47 73 62 79 87 81 98 112 93 115 117	111 100 130	12 15 15 19 22 27 30 33 33 46 57 61 80 93 82 116 109 1139	13 16 20 23 31 30 46 45 54 46 63 72 75 81 91	14 17 21 22 33 35 46 45 51 57 70 79 95 86	15 15 18 21 24 30 35 43 46 58 52 63 74 66 79	16 16 19 22 27 29 34 37 44 48 51 69 62 67	17 20 23 27 32 34 42 47 47 57 59 68	18 21 24 27 30 36 39 44 51 52 73	19 22 26 29 32 34 47 44 52 62			22 22 25 28 32 35 40 46		24 27 31 34 38	25 25 28 31 34	26 26 29 32	27 27 30	28	29	30	31	32	333	34	335	36	37 3	88 3	9 40	0 4	1 42	2 4:	144
	4 4 4 5	06 35 65 96 28	405 434 464 495	378 406 435 465	3 44 3 49 3 86 4 20 4 29	31 5 33 3 34 5 31 4 34 3	5 200 3 205 5 272 4 241 3 269	186 194 227 209 214	16 2 19 1 21 4 19 1 22 6	1 50 1 74 1 78 1 66	117 142 147 157 166	130 136 123 137				79 85 79 107 116							46 46 53 60 62				32 35 42 43 45		28 31 35 38 41 43	29 32 36 38 42	30 33 37 39		32	33										
35 36 37 38 39	5 6 6 7	95 30 66 03 41	594 6 29 6 65 70 2 740	561 595 630 666 703	511 562 588 599 642	343 432 478 433 591	3 348 2 289 3 338 7 459 1 404	23 5 30 3 29 2 3 23 3 24	230 259 280 270 277	226 253 282 225 243	175 186 214 214 210	147 190 159 195 201	174 160 151 181 177	133 145 148 191 214	124 123 147 159 188	127 113 141 138 149	120 131 123 127 157	109 110 113 141 128	103 104 98 106 116	85 87 102 108 105	86 86 86 101 114	75 91 85 96 98	76 89 105	65 72 81 83 82	59 78 73 76 90	61 61 69 69 71	51 54 60 74 67 70	52 61 60 65 68	50 53 56 59 63	46 48 52 56 61	43 46 49 52 60	41 44 47 50 55	39 42 44 48 51	36 39 42 45 49	40 43 47	41 45	42	3.7 10 3						
40 41 42 43 44 45	8 8 9	03	8 19 8 60 9 0 2 9 4 5	816 854 888	719 718 733 787	51 5 64 7 60 8 60 3	7 486 3 457	341 370 464 448	336 359	291 313 346 323	225 290 261 311	238 279 257	24 1 20 1 22 7 22 6 24 9 25 7	230 217	160 166 212 186 230 246	162 188 190 226	146 151 171 170 186 225	144 148 151 169	129 159 151 179	114 120 128 142 147 144	1 24 1 26 1 14 1 42 1 25 1 52	137 125 138	103 102 112 135 131	101 96 116 128 115 125	98 98 102 112 112	87 88 88 98 104 120	74 80 84 93 99 103	74 87 78 93 95 105	66 80 82 85 83 92	75 72 77 82	74	66 66 69 74	62 61 70 71	54 61 64 69	54 57 59 67	50 54 60 61	48 4 52 1 56 1	50 4 53 5 58 5	4 4 8 4	2 40 5 43 9 46 2 49	3 41 6 44 9 48	4 4 2 8 4 5	43	

Tabela 7. max iteracij = 1000, zacetna temperatura = 1.0, stopnja hlajenja = 0.99

																				Št	e vi lo	DOV	ezav															_	—					\neg
n\	d 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
3 4 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	1 3 6 10 15 21 28	2 5 9 14 20 27	3 6 10 15 21	4 7 11 16	5 8 12	6 9	7																																					
1 1 1 1 1 1 1 1	0 45 1 55 2 66 3 78 1 91 5 101	44 54 65 77 90 5 104	36 45 55 66 78 1 91	22 29 37 46 56 67	23 30 38 47 57 68	18 24 31 39 48 58	10 14 19 25 32 40 49	8 11 15 20 26 33 41	9 12 16 21 27 34	10 13 17 22 28	11 14 18 23	12 15 19	13 16	14																														
1 1 1 1 2 2 2	7 136 3 153 9 173 0 190 1 210	3 13 5 3 15 1 1 17 0 1 18 9 0 20 9	5 1 20 2 1 36 0 1 53 0 1 71 0 1 90	106	93 107 122 138 2 155	81 94 108 123 139	1 24	50 60 71 83 96 110 125	42 51 61 72 84 97	35 43 52 62 73 85 98	29 36 44 53 63 74 86	24 30 37 45 54 64 75	20 25 31 38 46 55 65	17 21 26 32 39 47 56	15 18 22 27 33 40 48	16 19 23 28 34 41	17 20 24 29 35	18 21 25 30	19 22 26	20 23	21																							
2 2 2 2 2 2 2	3 253 1 276 5 306 6 321 7 353 3 378	3 255 5 275 9 299 5 324 1 350 3 375	2 231 5 253 9 276 1 300 9 325 7 351	211 235 254 277 301 3 26	192 2 212 2 233 7 255 278 3 302	174 193 213 234 256 279	157 175 194 214 235 257	141 158 176 195 215 236	126 142 159 177 196 216	11 2 127 143 160 178 19 7	99 113 128 144 161 179	87 100 114 129 145 162	76 88 101 115 130 146	66 77 89 102 116 131	57 67 78 90 103 117	49 58 68 79 91 104	42 50 59 69 80 92	36 43 51 60 70 81	31 37 44 52 61 71	27 32 38 45 53 62	24 28 33 39 46 54	22 25 29 34 40 47	23 26 30 35 41	24 27 31 36	25 28 32	26 29	27																	
3 3 3 3 3 3 3	1 46 1 46 2 49 3 528 1 56	5 434 5 464 6 495 3 525 1 560	435 5 465 7 496 5 528	407	3 53 7 3 80 6 4 0 8 6 4 3 7 7 4 6 7	3 28 3 54 3 81 4 09 4 38	3 29 3 55 3 8 2 4 10	305 330 356 383	306	238 260 283 307	198 218 239 261 284 308 333	180 199 219 240 262 285 309	163 181 200 220 241 263 286	164	165 183 202 222	118 133 149 166 184 203 223	105 119 134 150 167 185 204	1 20 1 3 5 1 5 1	82 94 107 121 136 152 169	72 83 95 108 122 137 153	63 73 84 96 109 123 138	55 64 74 85 97 110 124	48 56 65 75 86 98 111	42 49 57 66 76 87	37 43 50 58 67 77 88	33 38 44 51 59 68 78	30 34 39 45 52 60 69	28 31 35 40 46 53 61	29 32 36 41 47 54	30 33 37 42 48	31 34 38 43	32 35 39	33 36	34										
3 3 4 4	630 7 666 8 703 0 74:	6 29 6 66 8 70 2 1 74 0	595 630 2 666 703 741	561 596	2 530 563 597 632 668	499 531 564 598 633	469 500 532 565 599	440 470 501 533 566	412 441 471 502	38 5 413 44 2 47 2 50 3	359 386 414 443 473	334 360	310 335 361 388 416	28 7 31 1 33 6 36 2 38 9	265 288 312 337	244 266 289 313 338	224 245 267 290 314 339	205 225 246 268 291	187 206 226 247 269 292	170 188 207 227 248 270	154 171 189 208 228 249	139 155 172	125 140 156 173 191	11 2 126 14 1 15 7	100 113 127 142	89 101 114 128 143 159	79 90 102 115 129	70 80 91 103 116 130	62 71 81 92 104 117	55 63 72 82 93 105	49 56 64 73 83	59 44 50 57 65 74 84	40 45 51 58 66 75	37 41 46 52 59 67	35 38 42 47 53 60	36 39 43 48 54	44	38 41 45	39 42	40				
4 4 4	2 863 903 1 946	1 860 3 900	8 20 2 8 6 1 5 9 0 3	781 8 21 8 62	743 782 8 22	706 744 783	6 70 70 7 74 5	635 671 708	601 636 672	568 602 637	53 6 56 9	50 5 53 7	475 506 538 571	446 476 507	418 447 477 508	391 419 448	365	340 366 393	316 341 367	293 317 342 368	271 294	250 272 295	230 251 273	21 1 23 1	193 212 232	176 194	160 177 195	145 161 178 196	131 146	118 132	106 119 133	95 107	85 96 108 121	76 86 97 109	68 77 87 98	61 69 78 88	55 62	50 56 63	46 51 57	43 47 52	48		43 46	44

Tabela 8. max iteracij = 1000, zacetna temperatura = 1.0, stopnja hlajenja = 0.99

																				Štev	ilo p	oveza	ıv															_					_	\neg
n\a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
16	1 3 6 10 15 21 28 36 45 55 66 78 91 105 120	2 5 9 14 20 27 35 44 54 65 77 90 104 119	3 6 10 15 21 28 36 45 55 66 78 91 105	4 7 11 16 22 29 37 46 56 67 79 92	5 8 12 17 23 30 38 47 57 68 80	6 9 13 18 24 31 39 48 56 65	7 10 14 19 25 26 40 42 59	8 11 15 18 26 31 36 49	9 12 16 21 23 30 31	10 13 17 20 24 31	11 14 18 23 25	12 15 19 23	13 16 19	14 17	15		17	18	19					24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
17 18 19 20	136 153 171 190 210 231 253 276 300	135 152 170 189 209 230 252 275 299 3 24 3 50 3 77 4 05	120 136 153 171 190 210 231 253 276 300 325 351 378 406	106 120 137 154 172 191 211 232 254 263 293 3 26	93 105 120 136 155 173 147 206 225 255 278 295 315 345	81 88 108 111 133 156 174 171 157	62 73 75 109 124 102 153 117 173 145 235 257 174 275	53 68 74 82 89 85 117 102 162 159 150 140 207 172	44 54 58 59 87 79 90 97 109 107 107 139 172 208	34 46 62 55 60 66 70 91 94 102 159 127 117 164	32 35 39 44 47 69 79 71 105 99 105 122 164	26 30 35 48 41 67 62 64	23 28 38 37 49 44 53 56 66 68 112 84 82 98	21 26 27 35 41 41 46 53 62 64 70 85 102 88	18 22 25 28 33 37 47 52 53 73 64 71 81 97	16 19 23 25 31 35 41 41 55 52 56 69 75 85	17 20 23 27 32 36 38 42 52 49 59 61 72	18 21 24 30 31 36 39 44 57 60 64	19 22 26 31 33 41 40 51 50 62	20 23 27 29 33 36 40 45 54	21 24 27 31 33 39 43 45 52	22 25 28 32 35 39 46 45	23 26 29 33 41 40 45	24 27 30 34 37 41	25 28 31 35 37	26 29 3 2 3 6	27 30 34	28 31	29															
31 32 33 34 35 36 37 38 39 40 41 42 43 44	465 496 528 561 595 630 666 703 741 780 8 20 8 61 9 03 9 46 9 90	464 495 527 560 594 629 665 702 740 779 819 860 902 945	43 5 46 5 49 6 52 8 56 1 59 5 63 0 66 6 70 3 74 1 77 9 81 7 85 1 88 8	391 404 466 464 519 534 586 596 635 678 724 731 774 779	373 408 352	299 252 403 371 440 458 305 364 437 539 570 625 519 526	30 7 29 3 24 0 3 2 7 30 1 26 8 44 0 33 8 39 7 38 9 47 7 38 7 37 8 47 5	184 222 320 247 297 406 278 311 333 454 363 363 584 432	184 231 199 229 228	162 175 219 173 175 193 276 327 249 272 259 289 339 385	140 117 158 154 174 210 250 302 259 268 259 250 348 307	150 154 144 138 158 150 198 228 219 214 315 246 306 232	1 29 1 48 1 35 1 38 1 46 1 49 1 75 1 86 2 07 1 80 2 54 2 07 2 76 2 49	103 10 2 11 7 12 7 13 9 15 2 14 8 15 7 26 1 17 1 16 9 20 8 19 7 25 4 23 2	103 96 104 117 128 140 137 169 163 170 180 171 202 200	77 94 90 143 122 113 138 154 132 240 160 163 180 204	90 85 95 124 115 115 144 128 139 143 158 149 168 211 226	71 73 76 85 98 112 126 140 111 169 166 174 155 171	64 75 75 93 90 110 98 119 120 140 132 144 155 149	66 62 67 78 122 99 103 102 108 114 117 128 146 143 191	58 63 79 85 78 88 96 100 112 128 145 125 128	49 56 65 63 72 83 82 91 98 97 94 118 125	48 51 54 63 69 72 77 80 90 91 111 110 120 117	45 46 57 60 86 73 87 79 94 109 99 107 106 125	42 45 52 54 57 66 63 77 81 83 96 105 99	39 43 46 49 58 60 68 64 82 78 84 94 97 124 103	37 39 45 46 51 56 56 59 70 71 73 79 88 89 111	34 38 41 43 52 52 55 71 69 63 74 80 83 87 97	32 36 38 41 47 48 52 58 59 70 71 77 80 89	43 48 49 53 62 62 68 71 77 88	37 41 43 47 51 53 58 61 63 68	38 41 45 50 51 55 62 66 64 68	45 48 53 58 64 62 65	37 40 43 46 51 55 58 63	38 41 45 48 51 55 57 63	52 55 60	43 46 49 52 59	4 5 4 7 51 53	$\frac{45}{48}$ 52	46 49	41 44 47 50		43 46	44

Tabela 9. max iteracij = 1000, zacetna temperatura = 1.0, stopnja hlajenja = 0.97