

Ciljni pristop pri reševanju nesimetričnega trgovskega potnika

Janez Brest, Aljaž Brest, Blaž Pšeničnik, Jan Popič, Klemen Berkovič, Borko Boškovič

Laboratorij za računalniške arhitekture in jezike, Inštitut za računalništvo,
Fakulteta za elektrotehniko, računalništvo in informatiko, Univerza v Mariboru,
Koroška cesta 46, 2000 Maribor, Slovenija
E-pošta: janez.brest@um.si

Solving the asymmetric traveling salesman problem using a target approach

In this article, we describe a technique for solving the asymmetric traveling salesman problem (ATSP) with a target-based approach. In the ATSP, an NP-hard problem, the goal is to find a Hamiltonian cycle of minimum total weight in a weighted complete graph. We have tested the heuristic algorithm, called RAI-2-opt, on problems from the TSPLIB library. In further experiments, we also applied the target-based approach, requiring the algorithm to find a solution of equal or better quality than a given benchmark value. We also checked whether the distribution of times required by our algorithm follows exponential trends.

1 Uvod

Problem trgovskega potnika (TSP) lahko preprosto opišemo takole: trgovski potnik mora obiskati seznam krajev (npr. mesta, vasi), ki so povezani vsak z vsakim. Povezave med kraji so različno dolge in imajo dodatne omejitve (zastoji, gost promet, slabo vozišče itd.). Vsaki povezavi priredimo utež, ki pomeni ceno, potrebno, da trgovski potnik opravi to pot. Trgovski potnik mora obiskati vsak kraj natanko enkrat in se vrniti v izhodiščni kraj; torej, trgovski potnik naredi cikel. Želja je, da pri tem minimizira stroške potovanja. Problem trgovskega potnika je močno raziskan in znan kombinatorični izziv v literaturi [1, 2, 3]. Njegovega reševanja so se lotili tudi z globokim učenjem [4].

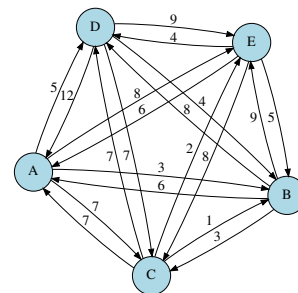
V članku se ukvajamo z reševanjem nesimetričnega problema trgovskega potnika (ATSP) s ciljnim pristopom. Pri ATSP, ki brez dodatnih predpostavk sodi v skupino NP-težkih problemov, želimo v uteženem polnem grafu poiskati minimalni Hamiltonov cikel. Hevristični algoritem, ki ga bomo nadgradili z lokalnim iskanjem, bomo preiskovali na nesimetričnih problemih trgovskega potnika iz knjižnice TSPLIB [5], ki vsebujejo od 17 do 443 vozlišč. Dalje bomo v eksperimentih uporabili tudi ciljni pristop, kjer želimo, da algoritem poišče rešitev, ki ima enako ali boljšo oceno rešitve od izbrane vrednosti. Osrednje razikovalno vprašanje je naslednje. Preveriti želimo, ali je porazdelitev časov zagonov pri našem hevrističnem algoritmu eksponentna.

Nadaljevanje prispevka je organizirano takole. V 2. poglavju predstavimo problem trgovskega pot-

nika. V 3. poglavju opišemo eksperiment in rezultate hevrističnega algoritma. Sledi sklepno, 4. poglavje, kjer podamo še smernice za nadaljnje raziskave.

2 Opis problema

V uteženem polnem grafu je treba poiskati minimalni Hamiltonov cikel. Hamiltonov cikel je sprehod v grafu, ki obišče vsako vozlišče, razen prvega in zadnjega, natanko enkrat. Cena sprehoda (ali skupna utež) je vsota uteži njegovih povezav.



Slika 1: Primer polnega usmerjenega grafa s petimi vozlišči. Vsako vozlišče je neposredno povezano z vsemi ostalimi vozlišči.

Slika 1 prikazuje primer polnega usmerjenega nesimetričnega grafa s petimi vozlišči (od A do E), kjer ima vsak par vozlišč dve povezavi s cenami, ki sta ponavadi različni. Poln usmerjen graf lahko predstavimo tudi z nesimetrično matriko, ki jo prikazuje enačba (1).

$$\begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{bmatrix} \infty & 3 & 7 & 5 & 8 \\ 6 & \infty & 3 & 8 & 9 \\ 7 & 1 & \infty & 7 & 2 \\ 12 & 4 & 7 & \infty & 9 \\ 6 & 5 & 8 & 4 & \infty \end{bmatrix} \end{matrix} \quad (1)$$

Na primer, Hamiltonov cikel $A \rightarrow D \rightarrow B \rightarrow C \rightarrow E \rightarrow A$, kjer se vsako vozlišče, razen vozlišča A, ki je hkrati startno in ciljno, ponovi natanko enkrat, ima ceno 20. A lahko morda najdemo cenejši Hamiltonov cikel? Odgovor je negativen. Iz vozlišča ne smemo opraviti sprehoda

v isto vozlišče, kar smo v matriki (1) označili s simbolom ∞ .

Matematično lahko zapišemo naslednjo definicijo za TSP: Za vsak par vozlišč c_i in c_j v polnem grafu z n vozlišči imamo podano razdaljo $d(c_i, c_j)$, ki jo lahko označimo z d_{ij} . Cilj pri reševanju problema trgovskega potnika je poiskati permutacijo vozlišč, π , ki minimizira:

$$\sum_{i=1}^{n-1} d(c_{\pi(i)}, c_{\pi(i+1)}) + d(c_{\pi(n)}, c_{\pi(1)}) \quad (2)$$

Vseh Hamiltonovih ciklov v polnem grafu z n vozlišči je $(n-1)!$, na primer, na sliki 1 je 24 ciklov. Glede na lastnosti razdalj d_{ij} lahko probleme trgovskega potnika razdelimo v tri skupine:

- evklidske (razdaljo med vozlišči lahko izračunamo s pomočjo metrike, npr. če so vozlišča grafa razporejena v dvodimenzionalnem prostoru, lahko kot ceno uporabimo evklidsko razdaljo),
- simetrične (za vsak par vozlišč velja $d_{ij} = d_{ji}$, kar pomeni, da je razdalja od vozlišča i do vozlišča j enaka razdalji od j do i) in
- nesimetrične ali asimetrične (pri teh problemih je lahko $d_{ij} \neq d_{ji}$).

V prispevku bomo uporabili algoritem RAI [6, 7], ki ga bomo nadgradili še z lokalno optimizacijo 2-opt [8, 9]. Ta novo nastali algoritem bomo poimenovali RAI-2-opt.

Algoritem RAI [6, 7] lahko na kratko strnemo takole. Iz cikla, ki predstavlja trenutno rešitev, odstranimo nekaj sosednjih vozlišč, ki jih nato eno za drugo naključno izberemo in ponovno na najcenejši način vstavimo v cikel. Dobljeni cikel primerjamo glede na ceno z začetnim in boljšega izmed njiju ohranimo. Postopek ponavljamo, dokler ni izpolnjen zaključni pogoj.

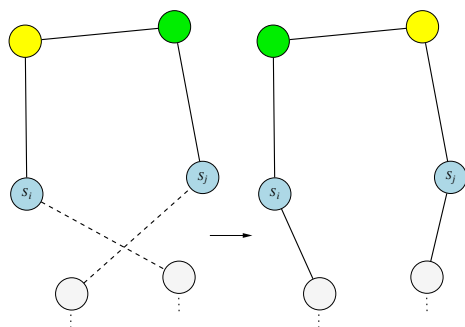
Algoritem 2-opt [8, 9] prištevamo k lokalni optimizaciji. Postopek išče dve povezavi, ki ju nadomestimo z dvema novima povezavama, tako da dobimo krajšo pot ter tako izboljšamo tudi cikel. Slika 2 prikazuje postopek lokalne optimizacije 2-opt. Lokalno optimizacijo 2-opt smo v algoritmu RAI-2-opt sprožili le, ko se je zgradil celotni cikel. Omenimo še, da smo pri lokalni optimizaciji omejili število vozlišč med indeksoma i in j na največ 5.

V osrednjem delu članka želimo v eksperimentih uporabiti ciljni pristop, ki je opisan v [10], kjer so ga avtorji uporabili na problemu iskanja binarnih zaporedij. Omenjen problem tudi spada med NP-težke probleme in v [10] so eksperimenti pokazali eksponentno porazdelitev števila ovrednotenj in tudi časov zagonov pri vseh velikosti binarnih zaporedij. V tem članku pa nas zanima, ali se tudi pri ATSP pojavlja eksponentna porazdelitev časov zagonov, ko uporabimo ciljni pristop.

3 Eksperimenti in rezultati

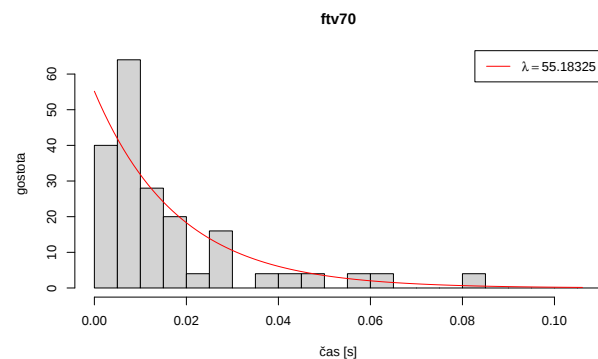
3.1 Število ponovitev gradnje cikla n^2 -krat

V eksperimentalnem delu najprej prikazujemo rezultate, ki jih je dosegel algoritem RAI-2-opt na asimetričnih



Slika 2: Lokalna optimizacija 2-opt. Pot trgovskega potnika pred (levo) in po uporabi premika 2-opt (desno). Povezavi, ki ju zamenjamo, sta označeni črtkano. Po uporabi 2-opt je sprehod $s_i, \dots, s_j$ obrnjen.

problemih trgovskega potnika iz knjižnice TSPLIB [5]. Algoritem je pri vsakem zagonu z različnim semenom ponovil gradnjo ciklov n^2 -krat, kjer je n število vozlišč v grafu. Dobljeni rezultati so zbrani v tabeli 1, kjer je prikazano: ime problema, njegova velikost (n), znana optimalna vrednost cikla, najboljša (min), najslabša (max), povprečna (avg) vrednost in standardna deviacija (std) ter (povprečni) čas, potreben za en zagon algoritma. Stolpci z oznako '%' prikazujejo, koliko odstotkov je pripadajoča vrednost dobljenega cikla nad optimumom. Opazimo lahko, da je algoritem v 21 primerih uspel najti optimalno rešitev. Povprečne vrednosti so znotraj meje 2,5 %, najslabše vrednosti pa zrastejo do 6,6 %. V štirinajstih primerih je maksimalna vrednost večja od 2 %.



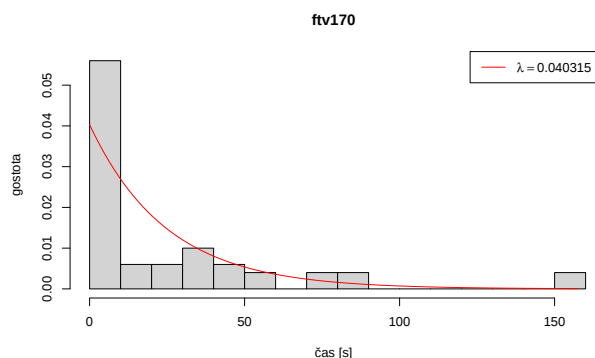
Slika 3: Histogram časov algoritma RAI-2-opt na asimetričnem problemu ftv70.

3.2 Cilji pristop

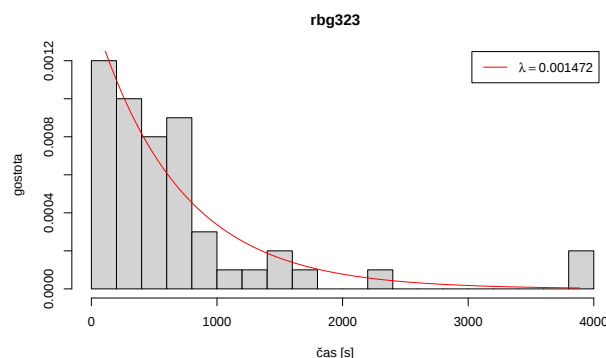
V nadaljevanju želimo uporabiti ciljni pristop pri reševanju problema trgovskega potnika. Pri tem pristopu izberemo predpisano vrednost cene cikla, c_t , potem pa algoritem teče, dokler ne doseže ali preseže vrednost c_t . Ker pri problemih iz knjižnice TSPLIB poznamo optimalne vrednosti rešitev, lahko kot vrednost c_t določimo odstopanje od optimalne vrednosti; v našem primeru smo izbrali 2 % in 1 % odstopanje od optimuma.

Tabela 1: Rezultati na asimetričnih problemih trgovskega potnika iz knjižnice TSPLIB pri številu ponovitev gradnje cikla n^2 -krat. Oznaka '%' prikazuje, koliko odstotkov je vrednost nad optimumom.

	problem	n	optimum	min.	%	max.	%	avg.	%	(std)	$t[s]$
1	br17	17	39	39	0	39	0	39	0	0	0,000167263
2	ftv33	34	1286	1286	0	1286	0	1286	0	0	0,00130825
3	ftv35	36	1473	1473	0	1490	1,15411	1477	0,271555	6,65459	0,00168288
4	ftv38	39	1530	1530	0	1557	1,76471	1540	0,653595	9,78685	0,00211487
5	p43	43	5620	5620	0	5621	0,0177936	5620	0	0,48	0,00414495
6	ftv44	45	1613	1613	0	1661	2,97582	1631	1,11593	15,0137	0,00366929
7	ftv47	48	1776	1776	0	1784	0,45045	1781	0,281532	3,68239	0,0045553
8	ry48p	48	14422	14422	0	14626	1,41451	14526	0,721121	55,9353	0,00401699
9	ft53	53	6905	6905	0	7017	1,62201	6930	0,362056	41,9886	0,00578466
10	ftv55	56	1608	1608	0	1651	2,67413	1619	0,68408	9,81307	0,00867284
11	ftv64	65	1839	1839	0	1899	3,26264	1855	0,870038	9,52347	0,0144043
12	ft70	70	38673	38836	0,421483	39583	2,35306	39126	1,17136	158,478	0,0159521
13	ftv70	71	1950	1950	0	1987	1,89744	1966	0,820513	7,68115	0,0202012
14	ftv90	91	1579	1579	0	1595	1,0133	1583	0,253325	3,2172	0,0567721
15	kro124p	100	36230	36241	0,0303616	37732	4,14574	36558	0,905327	540,72	0,063006
16	ftv100	101	1788	1788	0	1809	1,1745	1792	0,223714	3,70535	0,0827339
17	ftv110	111	1958	1958	0	1984	1,32789	1963	0,255363	4,97695	0,118732
18	ftv120	121	2166	2168	0,0923361	2249	3,83195	2192	1,20037	24,3089	0,176646
19	ftv130	131	2307	2307	0	2429	5,28825	2340	1,43043	32,2859	0,234944
20	ftv140	141	2420	2422	0,0826446	2592	7,10744	2441	0,867769	33,756	0,317074
21	ftv150	151	2611	2611	0	2765	5,89812	2667	2,14477	48,4954	0,420548
22	ftv160	161	2683	2683	0	2818	5,03168	2737	2,01267	41,7526	0,56605
23	ftv170	171	2755	2755	0	2903	5,37205	2813	2,10526	40,533	0,694078
24	rbg323	323	1326	1336	0,754148	1402	5,73152	1357	2,33786	9,89871	10,6171
25	rbg358	358	1163	1167	0,343938	1240	6,62081	1180	1,46174	12,2203	16,2293
26	rbg403	403	2465	2465	0	2538	2,96146	2467	0,0811359	10,2397	36,3748
27	rbg443	443	2720	2720	0	2729	0,330882	2721	0,0367647	2,29574	51,4779



Slika 4: Histogram časov algoritma RAI-2-opt na asimetričnem problemu ftv170.



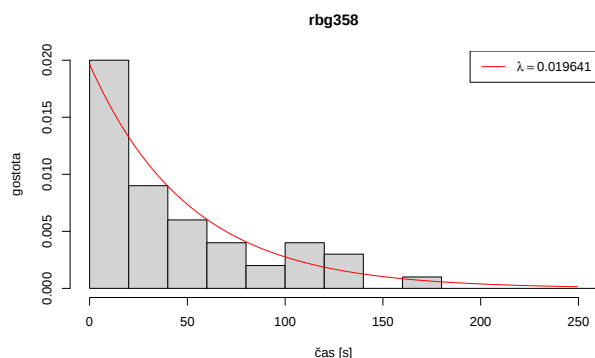
Slika 5: Histogram časov algoritma RAI-2-opt na asimetričnem problemu rbg323.

Algoritem RAI-2-opt smo na asimetričnih problemih iz knjižnice TSPLIB zagnali 50 krat in povprečne čase prikazuje tabela 2. Pri tem smo uporabili ciljni pristop z vrednostima 2 % in 1 %. V primeru ciljne vrednosti 1 % poudarjena p -vrednost označuje, da Kolmogorov-Smirnov test ($\alpha = 0,05$) ni zavrnil eksponentne porazdelitve časov zagonov algoritma. To se je zgodilo v 10 primerih: ftv33, ftv38, itd. Spomnimo, da so v [10] pri iskanju binarnih zaporedij zaznali eksponentno porazdelitev pri vseh velikostih problemov, pri ATSP pa se je to zgodilo v 10 primerih, kar znaša 37 %.

Slike 3, 4, 5 in 6 prikazujejo histograme časov algoritma RAI-2-opt v 50 neodvisnih zagonih na nesimetričnih problemih ftv70, ftv170, rbg323 in rbg358. Rdeča krivulja je eksponentno prileganje. Prikazana je tudi vrednost parametra λ . Na problemu ftv170 je statistični test Kolmogorov-Smirnov zavrnil trend eksponentne porazdelitve časov, medtem ko pa ga na problemih ftv70, ftv170 in rbg358 ni zavrnil. Najtrši oreh glede na potreben čas za doseg rešitev, kjer so ciljne vrednosti 1 % nad optimumom, našemu algoritmu predstavljajo problemi rbg323, rbg358 in ftv170. Pri njih je algoritem povprečno

Tabela 2: Povprečni časi algoritma RAI-2-opt na asimetričnih problemih trgovskega potnika iz knjižnice TSPLIB s ciljnim vrednostima 2 % in 1 %. Poudarjena p -vrednost označuje, da Kolmogorov-Smirnov test ni zavrnil eksponentne porazdelitve časov pri 50 zagonih algoritma.

	problem	n	$\bar{t}(2\%) [s]$	$\bar{t}(1\%) [s]$	p -vrednost
1	br17	17	0	0	0,003678
2	ftv33	34	6e-05	6e-05	0,2739
3	ftv35	36	4e-05	0,00132	0,0006613
4	ftv38	39	0,0001	0,00164	0,1681
5	p43	43	0	0	1,911e-08
6	ftv44	45	0,00154	0,05128	9,205e-05
7	ftv47	48	6e-05	0,00012	0,1154
8	ry48p	48	0,00028	0,00448	2,003e-07
9	ft53	53	0,00116	0,01006	2,954e-13
10	ftv55	56	0,00108	0,00722	0,01698
11	ftv64	65	0,00284	0,02266	0,3916
12	ft70	70	0,00464	0,195	1,024e-05
13	ftv70	71	0,00188	0,01812	0,3404
14	ftv90	91	0,00438	0,00818	0,06095
15	kro124p	100	0,1945	0,8622	8,882e-16
16	ftv100	101	0,00304	0,01118	0,6566
17	ftv110	111	0,0163	0,01822	0,356
18	ftv120	121	0,35272	4,4041	8,882e-16
19	ftv130	131	0,48564	3,64554	8,882e-16
20	ftv140	141	0,19618	0,29122	1,051e-07
21	ftv150	151	5,78496	8,2396	6,665e-07
22	ftv160	161	20,4824	18,6328	1,27e-07
23	ftv170	171	14,2059	24,805	0,0001055
24	rbg323	323	39,3286	679,174	0,3894
25	rbg358	358	7,89834	50,9152	0,6765
26	rbg403	403	0,05154	0,18816	0,003357
27	rbg443	443	0,12136	0,44824	0,0008194



Slika 6: Histogram časov algoritma RAI-2-opt na asimetričnem problemu rbg358.

potreboval 680, 51 in 25 sekund. Opazimo, da lahko časi pri ciljnim pristopu močno narastejo, npr. pri rbg323.

Podobne rezultate in grafe smo dobili tudi, če smo namesto časov uporabili število vstavljanj vozlišč pri algoritmu RAI-2-opt. Žal teh rezultatov zaradi prostorske omejitve ne moremo prikazati.

4 Sklep

V prispevku smo predstavili problem trgovskega potnika. Eksperimentalne rezultate smo na nesimetričnih proble-

mih iz knjižnice TSPLIB pridobili z algoritmom RAI-2-opt, ki smo ga dobili tako, da smo združili algoritem RAI z lokalnim iskanjem 2-opt. V eksperimentalnem delu smo uporabili tudi ciljni pristop. Histogrami časov zagonov programa v grobem kažejo na trende, ki so podobni eksponentni porazdelitvi. V 37 % primerov Kolmogorov-Smirnov statistični test potrdi, da trend časov zagonov sledi eksponentni porazdelitvi.

V prihodnje želimo opraviti eksperimente, kjer bi ciljne vrednosti še zaostriili, npr. 0,5 %, 0,1 %, itd. Eksponentno porazdelitev časov zagonov algoritmov želimo preveriti še na drugih NP-težkih optimizacijskih problemih.

Zahvala

J. Brest, A. Brest, J. Popič in B. Bošković priznavajo financiranje članka s strani Javne agencije za znanstvenoraziskovalno in inovacijsko dejavnost Republike Slovenije, raziskovalni program P2-0041 – Računalniški sistemi, metodologije in inteligentne storitve.

Literatura

- [1] Shi-Jie Jian and Sun-Yuan Hsieh. A niching regression adaptive memetic algorithm for multimodal optimization of the euclidean traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 27(5):1413–1426, 2022.
- [2] Petrică C. Pop, Ovidiu Cosma, Cosmin Sabo, and Corina Pop Sitar. A comprehensive survey on the generalized traveling salesman problem. *European Journal of Operational Research*, 314(3):819–835, 2024.
- [3] Panagiotis Karakostas and Angelo Sifaleras. A double-adaptive general variable neighborhood search algorithm for the solution of the traveling salesman problem. *Applied Soft Computing*, 121:108746, 2022.
- [4] Shengcai Liu, Yu Zhang, Ke Tang, and Xin Yao. How good is neural combinatorial optimization? A systematic evaluation on the traveling salesman problem. *IEEE Computational Intelligence Magazine*, 18(3):14–28, 2023.
- [5] Gerhard Reinelt. TSPLIB – A traveling salesman problem library. *ORSA Journal on Computing*, 3(4):376–384, 1991.
- [6] Janez Brest and Janez Žerovnik. An approximation algorithm for the asymmetric traveling salesman problem. *Ric. oper.*, 28:59–67, 1999.
- [7] Janez Brest, Janez Žerovnik in Viljem Žumer. Algoritem za nesimetrični problem trgovskega potnika. *Elektroteh. vestn.*, 70(1/2):40–45, 2003.
- [8] David S. Johnson. Local optimization and the traveling salesman problem. In *International colloquium on automata, languages, and programming*, pages 446–461. Springer, 1990.
- [9] Paulo R. de O. Costa, Jason Rhuggenaath, Yingqian Zhang, and Alp Akcay. Learning 2-opt heuristics for the traveling salesman problem via deep reinforcement learning. In *Asian conference on machine learning*, pages 465–480. PMLR, 2020.
- [10] Jana Herzog, Janez Brest, and Borko Bošković. Analysis based on statistical distributions: A practical approach for stochastic solvers using discrete and continuous problems. *Information Sciences*, 633:469–490, 2023.