Osječki matematički list 12 (2012), 139-149

Kako pomoći trgovačkom putniku

Velga Bosančić * Anka Golemac † Tanja Vojković ‡

Sažetak

Problem trgovačkog putnika, skraćeno *TSP* (*Traveling Salesman Problem*), je jedan od najpoznatijih i najproučavanijih problema kombinatorne optimizacije. Njegov matematički model je traženje Hamiltonovog ciklusa najmanje težine u težinskom grafu. Ovim radom se daje uvid u prirodu TSP-a te složenost i metode njegovog rješenja.

Ključne riječi: *TSP, Hamiltonov ciklus, algoritam*

How to Help a Traveling Salesman

Abstract

The Traveling Salesman Problem (TSP) is one of the most famous and most studied problems of combinatorial optimization. Its mathematical model consists of finding a minimum-weight Hamiltonian cycle in a weighted graph. This paper gives an overview of the TSP problem and its complexity throughout its history, variants, solution methods, and applications.

Keywords: *TSP, Hamiltonian cycle, Algorithm*

*Prirodoslovno-matematički fakultet Sveučilišta u Splitu, email: velbos@pmfst.hr
†Prirodoslovno-matematički fakultet Sveučilišta u Splitu, email: golemac@pmfst.hr
‡Prirodoslovno-matematički fakultet Sveučilišta u Splitu email: tanja@pmfst.hr

1 Uvod

Hassler Whitney (1907.-1989.), amer. matematičar, jedan od osnivača teorije singularnosti u matematici

William Rowan Hamilton, (1805.-1865.), u matematici prepoznatljiv po otkriću kvaterniona algebarske strukture tijela

Jedan od najsloženijih problema kombinatorne optimizacije je pronalazak u težinskom grafu ciklusa minimalne težine koji sadrži sve njegove vrhove. Poznata interpretacija tog problema je kako minimizirati ukupnu udaljenost koju trgovački putnik treba prijeći da bi posjetio svaki od n zadanih gradova točno jednom i vratio se u polazni grad. Otuda potječe njegov općenito prihvaćeni naziv "Problem trgovačkog putnika" kao i skraćenica **TSP** (*Traveling Salesman Problem*). Naziv je početkom 30-ih godina prošlog stoljeća uveo američki matematičar H. Whitney tijekom svojih seminarskih predavanja na Sveučilištu Princeton.

Problem trgovačkog putnika ima dugu povijest kroz koju je bio poticaj značajnim istraživanjima, prvenstveno u teoriji grafova i kombinatornoj optimizaciji [5], 1-16. Najraniji rezultat u vezi TSP-a seže u 1759. godinu, kada je Euler objavio rješenje za problem kretanja skakača po šahovskoj ploči (*Knight's Tour Problem*). Rane radove u ovoj tematici su imali A.T. Vandermonde (1771), T.P. Kirkman (1856) i W.R. Hamilton (1856) .

Rješavanje Problema trgovačkog putnika je veliki izazov, dijelom zbog njegove iznimne težine a dijelom zbog povezanosti sa zanimljivim praktičnim i teorijskim pitanjima. Potvrdu kompleksnosti, fundamentalnog značaja, popularnosti i primjenjivosti TSP-a nalazimo i na više od 1000 stranica u [1] i [5]. Danas je to referentni problem za ocjenu kvalitete raznih algoritama i optimizacijskih metoda o čemu ćemo govoriti u nastavku. Za početak dajemo kratki pregled uvodnih pojmova i činjenica vezanih uz grafove i algoritme.

Definicija 1.1. ***Graf*** *je uređena trojka* $G = (V, E, \varphi)$, *gdje je* $V = V(G)$ *neprazan skup čije elemente nazivamo* ***vrhovima***, $E = E(G)$ *je skup disjunktan s* V *čije elemente nazivamo* ***bridovima*** *i* φ *je funkcija koja svakom bridu* e *iz* E *pridružuje par* $\{u, v\}$, *ne nužno različitih vrhova* $u, v \in V$. *Graf skraćeno označavamo* $G = (V, E)$ *ili samo* G.

Za par vrhova u i v kažemo da su **susjedni** ako postoji brid e kojemu su oni krajevi. Pri tome kažemo da je brid e **incidentan** s vrhovima u i v i koristimo oznaku $e = \{u, v\}$ ili $e = uv$. **Stupanj** vrha v u grafu G je broj bridova grafa G incidentnih s v.

Šetnja u grafu G je niz $W := v_0 e_1 v_1 e_2 ... e_k v_k$ čiji članovi su naizmjenično vrhovi v_i i bridovi e_i tako da su krajevi od e_i vrhovi v_{i-1} i v_i, $i = 1, ..., k$. Broj k nazivamo **duljinom šetnje** W. Kažemo da je v_0 početak a v_k kraj šetnje W, da je W šetnja od v_0 do v_k ili (v_0, v_k) šetnja. Šetnja se, kada to ne umanjuje jasnoću, skraćeno zapisuje samo bridovima ili samo vrhovima. Šetnja je zatvorena ako je $v_0 = v_k$. Šetnju kojoj su svi bridovi međusobno različiti nazivamo **stazom**. **Put** je staza čiji su vrhovi međusobno različiti. **Ciklus** je zatvorena staza čiji su vrhovi osim krajnjih međusobno različiti.

Od značaja za našu temu su putovi i ciklusi koji sadrže sve vrhove grafa.

Definicija 1.2. ***Hamiltonov put (ciklus)*** *u grafu G je put (ciklus) koji sadrži sve vrhove grafa. Graf nazivamo* ***Hamiltonovim*** *ako sadrži Hamiltonov ciklus.*

Jedan od velikih neriješenih problema teorije grafova je naći nužan i dovoljan uvjet da bi graf imao Hamiltonov ciklus.

Definicija 1.3. ***Težinski graf*** *je uređeni par (G, w), gdje je $G = (V, E)$ graf i $w : E \to \mathbb{R}_0^+$ funkcija. Nenegativan broj $w(e)$ nazivamo* ***težinom*** *brida e.*

Težina zadanog brida može značiti visinu troška, udaljenost, vrijeme ili bilo koju drugu mjeru koja karakterizira taj brid.

Opširnije o grafovima može se naći u [4] i [10] .

Utvrđivanje je li graf Hamiltonov kao i pronalaženje u grafu Hamiltonovog ciklusa minimalne težine odnosno rješavanje TSP-a spadaju u kategoriju najtežih algoritamskih zadataka. Svakako je (barem u teoriji) iscrpnom pretragom konačnog grafa moguće u konačno mnogo koraka pronaći traženi ciklus ali u praksi je podjednako važno vrijeme u kojem se to postiže. Stoga su nam potrebni brzi i efikasni algoritmi. Vrijeme izvršavanja algoritma mjeri se ukupnim brojem osnovnih operacija (aritmetičkih operacija, usporedbi itd.) potrebnih za njegovu provedbu, a koji ovisi o veličini i prirodi ulaznih podataka.

Definicija 1.4. *Neka su $f, g : \mathbb{N} \to \mathbb{R}^+$ funkcije. Kažemo da je $f(n) = O(g(n))$ ako postoje $c, n_0 > 0$ takvi da za svaki $n \geq n_0$ vrijedi $f(n) \leq c \cdot g(n)$. Ako je $f(n) = O(g(n))$, onda za $g(n)$ kažemo da je gornja granica za $f(n)$ ili asimptotska gornja granica.*

Vremenska složenost algoritma se obično definira pomoću funkcije $f : \mathbb{N} \to \mathbb{N}$, gdje je $f(n)$ ukupan broj elementarnih operacija koje su algoritmu potrebne za rješavanje problema ulazne veličina n, [9]. Kada vrijedi $f(n) = O(g(n))$ kažemo da algoritam ima vremensku složenost $O(g(n))$. Ovisno o funkciji g razlikujemo polinomijalne i eksponencijalne algoritme. Za problem kažemo da je polinomijalan ako postoji polinomijalni algoritam koji rješava taj problem. Polinomijalne algoritme smatramo efikasnima jer vrijeme njihove provedbe raste relativno sporo u odnosu na rast ulaznih podataka, no za mnoge čak i naizgled jednostavno formulirane probleme, nije poznato postoji li polinomijalni algoritam za njihovo rješavanje. Ovdje kratko spominjemo jedan od najvećih problema današnjeg teorijskog računarstva, pitanje je li $P = NP$. P i NP su skupovi problema razvrstanih prema vremenskoj složenosti i nazivamo ih klasama vremenske složenosti. U klasu P spadaju problemi rješivi efikasno, tj. polinomijalnim algoritmima, a u klasu NP problemi za koje se dano rješenje može verificirati u polinomijalnom vremenu. Zna se da je $P \subset NP$, a pitanje vrijedi li obrat je jedno od najvažnijih neriješenih pitanja teorije računarstva. TSP spada u NP klasu problema. Štoviše, on spada u posebnu klasu NP problema, takozvane $NP-$potpune probleme. Ako bi se za neki $NP-$potpun problem pokazalo da je polinomijalno rješiv, tj. da pripada klasi P, to bi značilo da su svi NP problemi u klasi P te da je $P = NP$. To je još jedan od razloga važnosti i popularnosti Problema trgovačkog putnika. Više o teoriji složenosti algoritama može se naći u [9].

2 Povijest i opis

Kao što smo spomenuli, Euler je 1759. godine riješio Problem skakačevog obilaska šahovske ploče (*Knight's Tour Problem*). Tražio je odgovor na pitanje postoji li put kojim skakač može posjetiti svako polje šahovske ploče točno jedanput polazeći s proizvoljnog polja i krećući se prema šahovskom pravilima. Euler je konstruirao takav put za standardnu ploču 8×8. Poopćenje pitanja za ploču veličine $m \times n$, $m \leq n$, uz stroži zahtjev (traženje ciklusa), u terminima teorije grafova je: Za danu veličinu ploče, postoji li na pridruženom grafu „skakačev" Hamiltonov ciklus? Vrhovi pridruženog grafa predstavljaju polja šahovske ploče, a dva su vrha susjedna ako i samo ako skakač može u jednom potezu prijeći s jednog na drugo odgovarajuće polje. Odgovor je potvrdan osim za slučajeve kada su m i n oba neparni, kada je $m = 1, 2$ ili 4, te za $m = 3$ i $n = 4, 6$ ili 8 (Schwenkov teorem, [7]).

Kroz 19. stoljeće rješavani su pojedini posebni slučajevi TSP-a. Za matematičku formulaciju tih problema zaslužni su irski matematičar William Rowan Hamilton i engleski matematičar Thomas Penyngton Kirkman. Proučavanje Problema trgovačkog putnika u njegovom općem obliku započelo je oko 1930. Prvi se time bavio austrijski matematičar Karl Menger u formi traženja najkraćeg obilaska svih točaka nekog konačnog skupa s poznatim udaljenostima između svake dvije točke. On sam je to nazvao problem glasnika (messenger problem). Uskoro će Menger kao gost na Harvardu potaknuti interes šire matematičke zajednice za problem te će se u to vrijeme njime baviti matematičari u Beču te na Harvardu i Princetonu. Prateći nastavak povijesti TSP-a svjedočimo velikom i zanimljivom poslu koji traje do danas.

U literaturi postoje brojne formulacije TSP-a. Njegov osnovni oblik se najčešće formulira na dva načina.

1. Trgovački putnik ima zadan skup gradova od kojih svaki mora posjetiti točno jedanput i vratiti se u početni grad. Udaljenosti među gradovima su poznate. Pitanje je kojim redoslijedom bi trebao obilaziti gradove da ukupna duljina puta bude minimalna.

2. U težinskom grafu pronaći Hamiltonov ciklus minimalne težine.

Dakle, formulacije problema su jednostavne, ali do rješenja nije nimalo lako doći. Naime, ako bismo tražili najbolji obilazak pretraživanjem svih mogućnosti, već za par desetaka gradova bi nam trebalo jako puno vremena. Za graf od n vrhova maksimalni broj mogućih Hamiltonovih ciklusa je $n!$. Ako je početni vrh odnosno grad unaprijed određen, a obilaske $v_1v_2..v_{n-1}v_nv_1$ i $v_1v_nv_{n-1}...v_2v_1$ ne razmatramo posebno, onda je ukupni broj različitih obilazaka koje treba provjeriti $(n-1)!/2$. Odatle je jasno da je TSP faktorijelne složenosti. Grubom pretragom svih obilazaka današnjim osobnim računalima bi trebalo $6.8 \cdot 10^6$ sati već za problem od 20 gradova, a za 40 gradova broj sati bi bio $2.3 \cdot 10^{36}$, [8]. Za dobiti rješenja u prihvatljivom vremenu bilo je nužno razvijati efikasnije metode.

Godina	Istraživački tim	Broj gradova
1954	Dantzig, Fulkerson, Johnson	49
1971	Held, Karp	64
1975	Camerini, Fratta, Maffioli	67
1977	Grotschel	120
1980	Crowder, Padberg	318
1987	Padberg, Rinaldi	532
1987	Padberg, Rinaldi	2392
1994	Applegate, Bixby, Chvatal, Cook	7397
1998	Applegate, Bixby, Chvatal, Cook	13509
2001	Applegate, Bixby, Chvatal, Cook	15112
2004	Applegate, Bixby, Chvatal, Cook	24978

Slika 1: Povijest rješavanja TSP-a

Na taj način su postizani sve bolji rezultati, te je do 2004. problem riješen za obilazak 24978 švedskih gradova i to je do danas najveći poznati potpuno riješeni TSP. Trenutno najveći postavljeni problem je za 1904711 gradova diljem svijeta. Slika 1 prikazuje bitne povijesne instance u rješavanju TSP-a.

2.1 Varijacije problema

Postoje različite podvrste Problema trgovačkog putnika. Na primjer, možemo zamisliti situaciju u praksi u kojoj nisu sve ceste dvosmjerne pa duljina puta između gradova A i B može biti različita ovisno krećemo li iz A ili iz B. Također, težina brida može biti i negativna; primjerice kada težina brida nije određena samo duljinom puta nego ukupnim troškom puta između dvaju gradova, a zarada od prodaje na putu između neka dva grada prelazi trošak tog dijela puta.

Najčešće se promatraju sljedeće tri podvrste problema.

1. **sTSP (simetrični TSP):** To je prethodno opisani osnovni oblik problema, čiji je model (neusmjereni) težinski graf.

2. **aTSP (asimetrični TSP):** Ako za bar jedan brid $\{u, v\} \in E(G)$ (par gradova u i v) težina brida (duljina puta) ima različite vrijednosti ovisno o smijeru obilaska, onda se radi o asimetričniom TSP-u. U tom slučaju se kao model koristi usmjereni graf. **Digraf** ili **usmjereni graf** D je uređena trojka $D = (V, A, \psi)$, gdje je $V \neq \emptyset$ skup vrhova, A skup disjunktan s V čije elemente nazivamo lukovima i ψ funkcija koja svakom luku a iz A pridružuje *uređeni par* (u, v) vrhova $u, v \in V$.

3. **mTSP (multiple TSP):** Ovo je generalizacija Problema trgovačkog putnika. m trgovačkih putnika kreće iz istog početnog grada, obilaze zadani skup gradova i vraćaju se na početak. Treba odrediti obilaske za sve trgovačke putnike, tako da svaki grad bude posjećen točno jedanput i ukupan trošak puta bude minimalan. Trošak može značiti udaljenost gradova, vrijeme potrebno za putovanje, cijenu prijevoza i slično. Vezano uz primjene pojavljuje se razne varijante mTSP-a, [6, 2].

   a) Postoji nekoliko polaznih gradova i određeni broj trgovačkih putnika kreće iz svakog od njih. Nakon što završe obilazak trgovački putnici se vraćaju ili svaki u svoj polazni grad ili u bilo koji od polaznih gradova uz uvjet da na kraju u svakom polazištu bude jednak broj trgovačkih putnika kao i na početku.

   b) Broj trgovačkih putnika u obilasku ne mora biti fiksan. Od m trgovačkih putnika koji su na raspolaganju treba napraviti izbor onih koji će sudjelovati u obilasku. U tom slučaju obično svaki trgovački putnik ima svoje fiksne troškove koji se uzimaju u obzir pri odlučivanju koliko će trgovačkih putnika biti aktivirano i koji će to biti, s ciljem minimiziranja ukupnog troška.

c) Neke gradove treba posjetiti u točno određenim vremenskim intervalima. Taj problem se može izravno primijeniti u rješavanju koordinacije trajekata, obilazaka školskog autobusa ili organizaciji letova u zračnom prometu. Ova varijanta se obično označava sa mTSPTW (multiple Traveling Salesman Problem with Time Windows).

d) Mogu se uvesti i druge restrikcije kao što su ograničen broj gradova koje pojedini trgovački putnik može obići, minimalna ili maksimalna udaljenost koju neki od njih mora prijeći i slično.

3 Metode rješavanja

Metoda koja se prva nameće kad razmišljamo o rješavanju ovakvih problema je jednostavno pronaći sve moguće obilaske, izračunati njihove duljine i odabrati najbolji. Međutim, kako smo već komentirali, ova metoda zahtjeva previše računalnog vremena da bi bila korisna za veći broj gradova. Zato su razvijene mnoge aproksimativne metode koje relativno brzo daju prihvatljivo dobra rješenja. Modernim metodama moguće je u razumnom vremenu naći rješenje i za probleme od nekoliko milijuna gradova, koje je s velikom vjerojatnošću vrlo blizu optimalnog rješenja. Ovdje ćemo ukratko i vrlo općenito prezentirati neke od najčešćih metoda rješavanja. Više informacija i detalja mže se naći u [6].

3.1 Egzaktne metode

To su metode koje rezultiraju takozvanim egzaktnim algoritmom čiji ishod je sigurno najbolje rješenje. Njihov nedostatak je dugo vrijeme izvođenja. Osim već spomenute pretrage svih mogućih obilazaka, koja je zbog svoje $O(n!)$ složenosti nepraktična već za 10 i više gradova, još se koristi i metoda grananja i ograničavanja (Branch and Bound). Kod nje se vrši procjena svih mogućih rješenja i odbacuju loša na temelju unaprijed postavljene gornje i donje granice. Korisna je za oko 40-60 gradova.

3.2 Aproksimativne metode

Ove metode koriste algoritme koji daju približna rješenja. U relativno kratkom vremenu mogu se dobiti dovoljno dobra rješenja. Dakle, radi se o algoritmima manje vremenske složenosti ali koji općenito ne jamče očekivanu kvalitetu rješenja.

Metoda najbližeg susjeda. Riječ je o najjednostavnijoj i najizravnijoj aproksimativnoj metodi. Ideja je uvijek posjećivati sljedeći najbliži neposjećeni grad i kada su svi gradovi posjećeni vratiti se u početni. Vremenska složenost ovog algoritma za problem od n gradova je $O(n^2)$.

Pohlepni algoritmi. Staza se postupno izgrađuje dodavanjem uvijek najkraćeg mogućeg brida. Pritom ne smije nastati ciklus duljine manje od broja gradova niti se pojaviti vrh stupnja većeg od 2, tj. posjetiti neki grad više od jedanput. Također, treba paziti da se isti brid ne doda više puta. Složenost pohlepnog algoritma je $O(n^2 \log_2(n))$ za problem od n gradova.

Metoda umetanja gradova. Započinje se najkraćim obilaskom nekog podskupa od n zadanih gradova. Najčešće se uzima trokut, a može se početi i samo s jednim gradom ili s 2 grada povezana najkraćim bridom. Sljedeći grad koji se dodaje je onaj najbliži bilo kojem od gradova s prethodnog obilaska, a dodaje se na optimalno mjesto u obilasku. Postupak se ponavlja dok se ne dodaju svi gradovi. Složenost metode je $O(n^2)$.

Optimizacija kolonijom mrava. Znanstvenici često pokušavaju riješiti složene probleme oponašanjem procesa u prirodi. Ideja rješavanja TSP-a oponašanjem kretanja kolonije mrava se pokazala jako uspješnom i vrlo brzo daje optimalna rješenja za male probleme, [3]. Kada istražuju nova područja, mravi ostavljaju trag feromona koji onda vodi ostale mrave do novih izvora hrane. Algoritam simulira situaciju koja bi se mogla opisati na sljedeći način: Započinjemo sa skupinom mrava, otprilike njih 20tak, postavljenih u različitim gradovima i šaljemo ih u druge gradove. Nije im dozvoljeno vratiti se u početni grad sve dok ne obiđu sve ostale gradove. Mrav koji je odabrao najkraći put ostavit će najjači trag feromona, jer je količina feromona obrnuto proporcionalna duljini puta. Kada ostali mravi budu odlučivali kuda krenuti iz nekog grada slijedit će put s najjačim feromonskim tragom. Postupak se ponavlja dok se ne pronađe najkraći obilazak.

3.3 Poboljšavanje rješenja

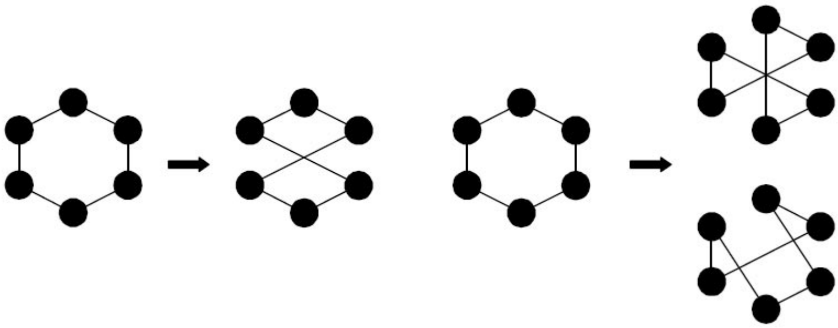
Nakon što je obilazak generiran nekim od aproksimativnih algoritama mogu se primijeniti metode poboljšanja rješenja.

2-opt, 3-opt i k-opt

2-opt algoritam bira nasumično 2 brida u ciklusu, uklanja ih i nanovo spaja dva nastala puta. Spajanje se vrši tako da se zadrže uvjeti obilaska, a dobiveni obilazak se dalje koristi samo ako je kraći od polaznog, Slika 2. Postupak se ponavlja dok daljnja poboljšanja nisu moguća, a dobiveni obilazak nazivamo $2-$optimalnim.

3-opt algoritam radi slično, ali uklanjaju se 3, umjesto 2 brida. Nakon uklanjanja 3 brida postoje 2 moguća načina ponovnog spajanja. Bira se onaj koji daje kraći obilazak. Dobivamo na kraju $3-$optimalni obilazak, a svaki

3−optimalni obilazak je i 2−optimalan. k-opt algoritam radi analogno za k bridova. Za veći k potrebno je i više računalnog vremena, pa se od $k > 3$ najčešće koristi još 4-opt.



Slika 2: 2-opt i 3-opt

Lin-Keringhan metoda

Glavna karakteristika osnovnog Lin-Kerninghan algoritma je da se za $k-$opt u svakoj iteraciji mijenja vrijednosti k. Drugim riječima, algoritam u svakom koraku iteracije za rastuće vrijednosti k ispituje hoće li izmjena k bridova rezultirati kraćim obilaskom. Ključ Lin-Kerninghan algoritma je ograničiti skup k-opt koraka u smislu da se razmatraju samo oni koraci koji su konstruirani kao niz od 2-opt koraka jer je u njihovom slučaju složenost najmanja. Algoritam nastavlja sve dok nisu zadovoljeni neki uvjeti zaustavljanja. Problem određivanja hoće li za neki k postupak dati kraći obilazak je sam po sebi dosta složen pa je ova metoda sporija od 2−opt metode i složenost joj je $O(n^{2.2})$.

4 Primjene i veza s nekim drugim problemima

Osim već navedene važnosti ovog problema za pitanje je li $P = NP$, postoje i brojni praktični problemi koji se mogu rješavati kao TSP ili se mogu na njega svesti. Ovdje navodimo neke od najčešće spominjanih, [6].

Bušenje sklopovskih pločica

Da bi spojili vodiče jednog sloja s vodičima drugog sloja u sklopovskoj pločici treba izbušiti rupe. One mogu biti različitih veličina, a svako mijenjanje veličine glave stroja zahtijeva određeno vrijeme. Utrošak vremena bismo očito optimizirali ako izbušimo prvo sve rupe jednog promjera, onda sljedećeg itd. Dakle problem bušenja možemo shvatiti kao nekoliko TSP-a zaredom, za svaki promjer po jedan, u kojem su "gradovi" pozicije na kojima treba izbušiti rupu istog promjera. Težina bridova se mjeri udaljenošću rupa, odnosno vremenom koje je potrebno stroju da dođe od jedne do druge. Cilj je minimizirati ukupni put glave stroja.

Rendgenska kristalografija

Za analizu strukture kristala koristi se rendgenski difraktometar. To je mjerni instrument koji s različitih pozicija mjeri intenzitet refleksije rendgenskih zraka od kristala i tako analizira strukturu materijala. Samo mjerenje traje kratko ali promjena pozicije zahtjeva mnogo vremena jer treba precizno odrediti poziciju između stotina tisuća koje je potrebno proći u nekim eksperimentima. Vrijeme potrebno za prelazak s jedne pozicije na drugu može se vrlo precizno izračunati, a krajnji rezultat mjerenja ne ovisi o rasporedu pozicija s kojih se mjeri. Potrebno je pronaći raspored pozicija koji minimizira ukupno vrijeme mjerenja, a to nas vodi do Problema trgovačkog putnika.

Određivanje rute vozila (Vehicle Routing Problem)

Određivanje optimalne rute vozila je sam za sebe poznati kombinatorni problem. Neka u gradu postoji n poštanskih sandučića koje treba svaki dan isprazniti unutar nekog fiksnog vremena, na primjer unutar jednog sata. Treba odrediti minimalni broj vozila koji je za to potreban i rute svih vozila tako da ukupan trošak pražnjenja sandučića bude najmanji. Problem se često formulira i preko n kupaca kojima treba dostaviti robu, odnosno određivanja broja potrebnih vozila te koje vozilo će posjetiti kojeg kupca i po kojem rasporedu.

Spomenimo još da se primjerice postupak spajanja komponenti matične ploče računala, raspoređivanje materijala u skladištu, raspored poslova koje obavlja jedan ili više strojeva i brojni drugi problemi modeliraju kao Problem trgovačkog putnika.

Napomena. Članak je nastao pri izradi završnog preddiplomskog rada Velge Bosančić na studiju Matematike PMF-a u Splitu.

Literatura

[1] D.L. Applegate, R.E. Bixby, V. Chvátal, W.J. Cook, *The Traveling Salesman Problem: A Computational Study*, Princeton University Press, 2006.

[2] T. Bektas, *The multiple traveling salesman problem: an overview of formulations and solution procedures*, Science Direct, 2005.

[3] M. Dorigo, L.M. Gambardella, *Ant colonies for the travelling salesman problem*, BioSystems 43 (1997), 73–81.

[4] J.L. Gross, J. Yellen, *Graph Theory and Its Aplication*, CRC Press, 1999.

[5] E.L. Lawler, J.K. Lenstra, A.H.G Rinnooy Kan, D.B. Shmoys (eds.),*The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization,* Wiley, Chichester, 1985.

[6] R. Matai, S. Prakash Singh, M. Lal Mittal, *Traveling Salesman Problem: An Overview of Applications, Formulations, and Solution Approaches*, InTech, 2010.

[7] A.J. Schwenk, *Which rectangular chessboards have a knight's tour?,* Mathematics Magazine, 64(5) (1991), 325-332.

[8] S. Singer, *Numerička matematika*, FSB, Zagreb, 2004./05.

[9] M. Sipser, *Introduction to the Theory of Computation*, PWS Pub. Co., Boston, 1997.

[10] D. Veljan, *Kombinatorna i diskretna matematika*, Algoritam, Zagreb, 2001.