**Problem trgovačkog putnika za neholonomske mobilne robote – upute za pokretanje programa**

**Autor:** Matej Petrović

U ovom dokumentu sadržane su upute za pokretanje programa napravljenih u sklopu diplomskog rada *Problem trgova kog putnika za neholonomske mobilne robote* za akademsku godinu 2024/2025 na Fakultetu elektrotehnike i računarstva Sveučilišta u Zagrebu.

Programi rješavaju probleme zadane u datotekama koje je napravila struktura Costs\_Data\_Holder.

# Sadržaj

1. Memetički algoritmi (GTSP\_Memetic\_algorithm)
2. Decoupled algoritam
   1. OR-Tools\_TSP\_solver.py
   2. GTSP\_Optimize\_headings.cpp

# Memetički algoritmi (GTSP\_Memetic\_algorithm)

**Kratak opis**

GTSP\_Memetic\_algorithm je program napisan u programskom jeziku C++ i implementira memetičke algoritme slične onomu kojeg su razvili G. Gutin i D. Karapetyan za *A Memetic Algorithm for the Generalized Traveling Salesman Problem (2010)*.

Algoritmi se razlikuju po tome koliko koriste funkciju optimizeHeadings(). Prva inačica poziva optimizeHeadings() tijekom svake generacije, a druga samo za posljednju iteraciju. Korisnik bira koji algoritam će pokrenuti unošenjem odgovarajuće zastavice.

**Pokretanje**

Program se pokreće na sljedeći način:

GTSP\_Memetic\_algorithm.exe optimize\_headings\_flag input\_file output\_file [backwards\_flag]

optimize\_headings\_flag može biti 0 ili 1 i određuje hoće li se provoditi optimizeHeadings() u svakoj generaciji ili ne.

* 0 označava da se optimizeHeadings() neće provoditi u svakoj generaciji već sam u posljednjom generaciji.
* 1 označava da će se optimizeHeadings provoditi u svakoj generaciji.

input\_file je datoteka (ime i put do datoteke) koju je napravio Cost\_Data\_Holder koja sadrži podatke o problemu.

output\_file je osnovni oblik naziva datoteke u koju će se spremati rezultati koji uključuju sve generacije i konačno rješenje. Datoteka će se uvijek spremati u mapu generations i imat će dodatak koji označava o kojoj je generaciji riječ. Ovisno o vrijednostima optimize\_headings\_flag i backwards\_flag, lokacija i ime imat će zajedno oblik:

* "generations/with\_optimizeHeadings/backwards/{output\_file}-with\_MA\_gen{i}.txt" za optimize\_headings\_flag = 1 i backwards\_flag = 1,
* "generations/with\_optimizeHeadings/no\_backwards/{output\_file}-with\_MA\_gen{i}.txt" za optimize\_headings\_flag = 1 i backwards\_flag = 0,
* "generations/without\_optimizeHeadings/backwards/{output\_file}-with\_MA\_gen{i}.txt" za optimize\_headings\_flag = 0 i backawrds\_flag = 1,
* "generations/without\_optimizeHeadings/no\_backwards/{output\_file}-with\_MA\_gen{i}.txt" za optimize\_headings\_flag = 0 i backwards\_flag = 0,

gdje je {i} redni broj generacije. Umjesto "{output\_file-with}\_MA\_gen{i}.txt" stajat će "{output\_file}-with\_MA\_FINISHED.txt" ako je riječ o datoteci koja sadrži podatke o konačnom rješenju ili će stajati "{output\_file}-end\_OH.txt" ako je riječ o datoteci s podacima o optimizaciji posljednjeg rješenja kad se ne koristi optimizeHeadings u svakoj iteraciji (optimize\_headings\_flag = 0).

[backwards\_flag] može biti 0 ili 1. 0 označava da se robot ne može kretati unatrag, a 1 da može. Ako se niti jedna vrijednost ne unese, [backwards\_flag] se automatski postavlja na 1.

**Primjer pokretanja**

Pokretanjem

GTSP\_Memetic\_algorithm.exe 1 results\_costs/primjeri\_za\_usporedbu/paralel\_order.txt paralel\_order-out 1

pročitat će se problem iz datoteke results\_costs/primjeri\_za\_usporedbu/paralel\_order.txt. Tijekom rješavanja problema, tijekom lokalne optimizacije svake generacije pozvat će se funkcija optimizeHeadings(). Ako, na primjer, program obradi ukupno 3 generacije, rješenja će biti spremljena u:

* generations/with\_optimizeHeadings/backwards/paralel\_order-out-wiht\_MA\_gen1.txt
* generations/with\_optimizeHeadings/backwards/paralel\_order-out-wiht\_MA\_gen2.txt
* generations/with\_optimizeHeadings/backwards/paralel\_order-out-wiht\_MA\_gen3.txt
* generations/with\_optimizeHeadings/backwards/paralel\_order-out-wiht\_MA\_FINISHED.txt

# Decoupled algoritam

Decoupled algoritam sa sastoji od dva dijela: OR-Tools za TSP u datoteci OR-Tools\_TSP\_solver.py i optimizeHeadings za GTSP u datoteci GTSP\_Optimize\_headings.cpp.

## OR-Tools\_TSP\_solver.py

**Kratak opis**

OR-Tools\_TSP\_solver.py je program napisan u programskom jeziku Python i implementira OR-Tools alat za rješavanje TSP-a, odnosno, koristi njegov CP\_SAT solver.

Dobiveno rješenje sprema se u "GTSP\_results/TSP\_result.txt" kako bi ga dalje mogao koristiti GTSP\_Optimize\_headings.

**Pokretanje**

Program se pokreće na sljedeći način:

python OR-Tools\_TSP\_solver.py input\_file

input\_file je datoteka koju je napravio Costs\_Data\_Holder koja sadrži podatke o problemu.

Program će na osnovu tih podataka sebi zadati TSP problem koji će riješiti.

Rješenje će UVIJEK smjestiti u „GTSP\_results/TSP\_result.txt“.

**Primjer pokretanja**

Pokretanjem

python OR-Tools\_TSP\_solver.py results\_costs/primjeri\_za\_usporedbu/paralel\_order.txt

program će iz datoteke results\_costs/primjeri\_za\_usporedbu/paralel\_order.txt učitati podatke i na osnovu njih zadati TSP problem. Riješit će zadani problem i rješenje upisati u GTSP\_results/TSP\_result.txt.

## GTSP\_Optimize\_headings.cpp

**Kratak opis**

Program je napisan u programskom jeziku C++ i implementira A\* algoritam za rješavanje GTSP-a. A\* je implementiran kao funkcija optimizeHeadings().

Ovaj program uzima rješenje TSP-a koji je dao OR-Tools\_TSP\_solver.py, odnosno, niz čvorova, i za taj niz bira optimalan poredak stanja.

**Pokretanje**

Program se pokreće na sljedeći način:

GTSP\_Optimize\_headings.exe input\_file output\_file [backwards\_flag]

input\_file je datoteka (put i ime) koju je napravio Costs\_Data\_Holder koja sadrži podatke o problemu.

output\_file je datoteka (samo ime) u koju se sprema rezultat programa. Ona će biti smještena u "GTSP\_results/Decoupled-OH results/backwards/" ili "GTSP\_results/Decoupled-OH results/no\_backwards/", ovisno o vrijednosti [backwards\_flag].

[backwards\_flag] može biti 0 ili 1. 0 označava da se robot ne može kretati unatrag, a 1 da može. Ako se niti jedna vrijednost ne unese, [backwards\_flag] se automatski postavlja na 1.

**Primjer pokretanja**

Pokretanjem

GTSP\_Optimize\_headings.exe results\_costs/primjeri\_za\_usporedbe/paralel\_order.txt paralel\_order-out.txt 0

program će učitati problem iz datoteke results\_costs/primjeri\_za\_usporedbe/paralel\_order.txt i niz čvorova iz GTSP\_results/ TSP\_result.txt.

Rezultat će spremiti u GTSP\_results/Decoupled-OH results/no\_backwards/paralel\_order-out.txt.