



UNIVERSITÀ DEGLI STUDI DI PADOVA

SCHOOL OF ENGINEERING DEPARTMENT OF INFORMATION  
ENGINEERING

MASTER DEGREE IN COMPUTER ENGINEERING

## Exact and heuristic approaches to the Travelling Salesman Problem (TSP)

Operations Research 2 : 2023/2024

**Supervisor:**

Matteo Fischetti

**Candidates:**

Fantin Luca (2119287)

Zanella Matteo (2122187)

# Contents

- 1 Introduction . . . . . 1**
  - 1.1 Problem formulation . . . . . 1
- 2 Project Setup . . . . . 3**
- 3 Heuristic methods . . . . . 4**
- 4 Exact methods . . . . . 5**
- 5 Bibliography . . . . . 6**

# Chapter 1

## Introduction

The *Travelling Salesman Problem*, also known as **TSP**, is one of the most famous and studied optimization problems in the Computer Science and Operations Research fields. Although its first mathematical formulation was proposed in the 19th century by the mathematicians **William Rowan Hamilton** and **Thomas Kirkman**, it received scientific attention from the 1950s onwards.

In 1972, **Richard M. Karp** proved the **NP-hard nature** of the TSP; this meant that the computation time for any solving algorithm can grow exponentially with the input size. Despite this, many different approaches have been developed over the years, yielding both exact and approximate solutions.

In this paper, several algorithms are explained, developed and tested against each other, both exact and approximate.

### 1.1 Problem formulation

In this thesis, we will consider an **undirected graph**  $G = (V, E)$ , where  $V$  is a set of  $|V| = N$  *nodes* (or *vertices*) and  $E$  is a set of  $|E| = M$  *edges*.

We define a **Hamiltonian cycle** of  $G$ ,  $G^* = (V, E^*)$ , as a graph whose edges form a cycle going through each node  $v \in V$  exactly once.

We also define a **cost function** for the edges  $c : E \rightarrow \mathbb{R}^+$ ,  $c_e := c(e) \forall e \in E$ .

The target of the TSP is finding an Hamiltonian cycle of  $G$  of minimum total cost, obtained by summing the costs of all edges in the cycle.

We can formulate this problem through *Integer Linear Programming (ILP)*. First, we define the following decision variables to represent whether or not a certain edge is included in the Hamiltonian cycle:

$$x_e = \begin{cases} 1 & \text{if } e \in E^* \\ 0 & \text{otherwise} \end{cases} \quad \forall e \in E$$

The ILP model is the following:

$$\left\{ \begin{array}{l} \min \sum_{e \in E} c_e x_e \\ \sum_{e \in \delta(h)} x_e = 2 \quad \forall h \in V \\ \sum_{e \in \delta(S)} x_e \leq |S| - 1 \quad \forall S \subset V : v_1 \in S \\ 0 \leq x_e \leq 1 \quad \text{integer} \quad \forall e \in E \end{array} \right. \quad \begin{array}{l} (1.1) \\ (1.2) \\ (1.3) \\ (1.4) \end{array}$$

Constraints 1.2 impose that every node of the graph must be touched by exactly two edges of the cycle. This group of constraints alone isn't enough to guarantee to find a valid Hamiltonian Cycle: we could find lots of isolated cycles.

Constraints 1.3, called **Subtour Elimination Constraints (SEC)**, guarantee that any solution found through this model is made up of only one connected component: every vertex  $v \neq v_1$  must be reachable from  $v_1$ .

Despite their importance, their number is exponential in  $N$ , thus, considering all of them at once is computationally expensive.

# Chapter 2

## Project Setup

# Chapter 3

## Heuristic methods

# Chapter 4

## Exact methods





# Chapter 5

## Bibliography