**UNIVERSITY OF CATANIA**
Department of Mathematics and Computer Science
Bachelor's Degree in Computer Science

*Matteo Galletta*

# Minimum Weight Vertex Cover Problem

Artificial Intelligence Project

Professors: Mario Francesco Pavone
Vincenzo Cutello

Academic Year 2024 - 2025

# Contents

# Chapter 1

# Introduction

## 1.1 Problem

Given a problem instance $(G, \omega)$, where $G$ is a undirected graph $G(V, E)$ and $\omega : V \to \mathbb{R}^+$ a function that associates a positive weight value $\omega(v)$ to each vertex $v \in V$, the Minimum Weight Vertex Cover can formally be defined as follows:

$$\textbf{minimize} \quad \omega(S) = \sum_{v \in S} \omega(v), \quad S \in V$$

such that $\forall (v_i, v_j) \in E, v_i \in S \vee v_j \in S$.

Note that the MWVC is a NP-complete problem.

## 1.2 Algorithm choice

This project implements a solution to the MWVC problem using a *Genetic Algorithm (GA) with one-point crossover and k-tournament selection.*

The chosen algorithm facilitates parallel computing compared to Tabu Search, as well as allowing to reduce the risk of the local optima stagnation. Also, Branch-and-Bound has limitations concerning the problem size, while Genetic Algorithms are suitable for NP-Hard problems such as MWVC.

Regarding the chosen Genetic Algorithm variation, even though it is not immediate to state whether one-point crossover will achieve better results over uniform crossover without prior testing, it has been widely proved that k-tournament selection outperforms roulette wheel in most scenarios [1].

# Chapter 2

# Genetic Algorithms

## 2.1 Behaviour and Structure

Genetic Algorithms (GAs) are a class of evolutionary algorithms inspired by the principles of natural selection. They operate by iteratively evolving a population of potential solutions towards an optimal or near-optimal state. The process unfolds as follows:

**Initialization** An initial population of candidate solutions is randomly generated.

**Evaluation** Each individual in the population is evaluated based on a predefined fitness function, which quantifies its quality or suitability with respect to the problem at hand.

**Selection** Individuals from the populations is chosen from the population. The selected individuals are selected as parents of the following population. Selection is based on fitness, with fitter individuals having a higher probability of being chosen.

**Generation** Using crossover, selected parents genetic material are combined to create offspring. Then, with a certain probability, random mutations are introduced into the offspring. This is called mutation and it helps to maintain diversity within the population and prevents premature convergence towards suboptimal solutions. This newly generated population replaces the old one.

**Iteration** The cycle of evaluation, selection and generation is repeated until halting criteria are satisfied.

## 2.2 Selection

k-tournament detailed + general

## 2.3 Mutation

one-point detailed + general vedi se c'è solo one-point o se c'è altro da mettere

# Chapter 3

# Implementation

## 3.1 DEAP Framework

The chosen framework for the implementation is DEAP [2]. DEAP (Distributed Evolutionary Algorithms in Python) is a Python library that excels at rapid prototyping and testing of ideas, making the tool ideal for our needs.

## 3.2 Genes Structure

struttura geni

## 3.3 Testing

descrivi istanze test + pseudo per benchmark

# Chapter 4

# Benchmark

boh, vari tuning e benchmark

# Conclusion

# Bibliography

[1] Saneh Lata, Saneh Yadav, and Asha Sohal. Comparative study of different selection techniques in genetic algorithm. *International Journal of Engineering Science*, 07 2017.

[2] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.