Quantum Integer Programming for the Capacitated Vehicle Routing Problem

Master's Degree in Data Science

Anthony Palmieri (2038503)

December 9, 2023

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Table of Contents

Motivations

- **Quantum Computing (QC)** a compelling alternative to classical computing paradigms.
- QC harnesses the unique phenomena of quantum mechanics.
- Speedups have already been demonstrated for numerous classical algorithms.
- What about **Combinatorial Optimization (CO)**?
- In this thesis, Quantum algorithms to tackle CO problems.
- Primary focus is on the **Capacitated Vehicle Routing Problem (CVRP)**.

Table of Contents

Quantum Computing Essentials

- In Quantum Computing (QC) the fundamental unit of information is the **qubit**.
- Unlike classical bits, qubits can exist in a **superposition** of states, simultaneously being both 0 and 1

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \qquad |\alpha|^2 + |\beta|^2 = 1, \qquad \alpha, \beta \in \mathbb{C}.$$

- Superposition allows for *parallelization!*
- Qubits can also become **entangled**: the state of one qubit becomes dependent on the state of another.

Quantum States

- A quantum system made of n qubits $|\psi_1\rangle, \ldots, |\psi_n\rangle$ is described by the *tensor product* of the individual qubits

$$|\psi_1\psi_2\cdots\psi_n\rangle := |\psi_1\rangle \otimes \cdots \otimes |\psi_n\rangle.$$

- A general quantum state $|\phi\rangle$ is a superposition of all classical states $|x\rangle$

$$|\phi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x|x\rangle, \qquad \sum_x |\alpha_x|^2 = 1, \qquad \alpha_x \in \mathbb{C}\ \forall x.$$

- $|\alpha_x|^2$ is the probability of observing the classical state $|x\rangle$.
- Thus, a quantum system can be in multiple classical states simultaneously, with probabilities determined by the **amplitudes** α_x.

Table of Contents

3 Combinatorial Optimization via QC

QUBO Formulations

- Combinatorial Optimization: finding the best solution from a finite set of possibilities.
- CO problems can be challenging due to vast number of potential solutions.
- To apply quantum algorithms, we need to convert them into a “compatible format”.
- The bridge between CO and QC is the **QUBO (Quadratic Unconstrained Binary Optimization)** formulation

$$\min_{x \in \{0,1\}^n} x^T Q x.$$

- **Ising formulation** is derived from the QUBO via change of variables $\{0, 1\} \ni x_i \to s_i \in \{-1, 1\}$, making it *equivalent*

$$\mathcal{E}(s_1, \ldots, s_n) = -\sum_{i<j} J_{ij} s_i s_j - \sum_{i=1}^{n} h_i s_i .$$

- It serves as a quantum-compatible representation of the original CO problem.
- From $\mathcal{E}$ we get the **Ising Hamiltonian** H_p by replacing each variable s_i with the related **Pauli-Z** operator σ_i^z

$$\sigma_i^z = \underbrace{I \otimes I \otimes \cdots \otimes I}_{i-1 \text{ terms}} \otimes \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}}_{i\text{-th term}} \otimes \underbrace{I \otimes I \otimes \cdots \otimes I}_{n-i \text{ terms}} .$$

- $H_p = \mathcal{E}(\sigma_1^z, \ldots, \sigma_n^z)$ is a square matrix of order 2^n.

Ising Hamiltonians

- Ising Hamiltonians *mapped* on quantum computers through **minor embedding**.
- There is a connection between the eigen-structure of H_p and the original CO problem.
- Namely, the eigenvectors associated to the lowest eigenvalue corresponds to the optimal solutions of our CO problem.
- Our focus shifts from solving the original problem to finding the **ground-state** of H_p.

Table of Contents

Adiabatic Quantum Computation

- AQC is rooted in the **Adiabatic Theorem** of quantum mechanics.
- Consider a system evolving under the Schrödinger equation

$$i\hbar\frac{\partial|\psi(s)\rangle}{\partial s} = H(s)|\psi(s)\rangle$$

  with $H(s) = (1-s)H_0 + sH_p$, $s \in [0,1]$.
- Assume at $s = 0$ the system is in the ground-state of H_0.
- If the evolution is *slow enough*, at $s = 1$ the system is in the ground-state of H_p.
- *Slow enough?* An upper bound on the total annealing time is given by

$$\frac{1}{\min_{s\in[0,1]}(\epsilon_1(s) - \epsilon_0(s))^2}$$

  with $\epsilon_0(s), \epsilon_1(s)$ lowest and second lowest eigenvalues of $H(s)$.

Quantum Graver Augmentation

- QGA relies on **Graver test set** $\{x \mid Ax = 0, x \in \mathbb{Z}^n, x \neq 0\}$, with A constraint matrix.
- Test sets are sets of directions such that starting from any feasible solution and moving along these directions leads to the optimum.
- Classical computers struggle with test set computation. *Turn to quantum!*
- Use AQC to compute Graver test set

$$||Ax||^2 = x^T A^T Ax \stackrel{!}{=} 0.$$

- Use AQC to compute starting feasible solutions

$$||Ax - b||^2 = (Ax - b)^T (Ax - b) \stackrel{!}{=} 0.$$

- VQAs are **hybrid** quantum-classical algorithms.
- They leverage a quantum-classical feedback loop.
- Quantum computers compute a parameterized quantum state

$$|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\psi_0\rangle$$

  using a a parameterized unitary $U(\boldsymbol{\theta})$ and a starting state $|\psi_0\rangle$.
- The classical optimizer tunes the parameters to minimize

$$C(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|H_p|\psi(\boldsymbol{\theta})\rangle.$$

- Once the optimal θs are found, recompute $U(\boldsymbol{\theta})$ and start again.
- Repeat until convergence or stopping criterion is met.

VQE and QAOA

- VQE and QAOA are VQAs.
- VQE a very flexible approach.
- In VQE, $U(\boldsymbol{\theta})$ often consists of rotations and entangling transformations.
- QAOA is specifically thought for CO problems.
- QAOA is a time-discretization of AQC via *Trotterization* $(e^{A+B} \approx e^{A}e^{B})$.
- QAOA structure

$$U(\boldsymbol{\beta}, \boldsymbol{\gamma}, L) = \prod_{j=1}^{L} e^{-i\beta_j H_0} e^{-i\gamma_j H_p}, \qquad |\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle .$$

Table of Contents

5 Capacitated Vehicle Routing Problem

CVRP

- CVRP an extension of the classic TSP.
- The most basic and simple variant of *Vehicle Routing Problems*.
- K vehicles, all starting at the depot node D, with maximum load capacity C.
- n customers to be served, each with a demand d_i, $i = 1, \ldots, n$.
- *Goal:* serve all customers while respecting capacity constraints and minimizing the overall routing cost.
- Vehicles start and end their routes at the depot node.

Solution Approach

- Full CVRP formulation impractical on current *NISQ* devices; *too many variables!*
- Use the heuristic approach: *Clustering + Routing*.
- First, cluster the customers. One cluster per vehicle.
- Then, within each cluster, model a TSP.
- Clustering: **Multi-Knapsack Problem, Modularity Maximization, Louvain Community Detection**.
- Routing: add *subtour elimination constraints* as needed (**recursive DFJ strategy**).

Table of Contents

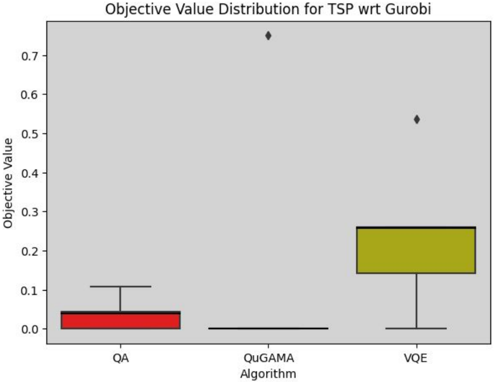
6 Numerical Experiments

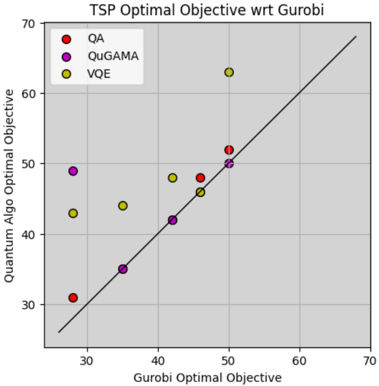
Numerical Experiments

- 25 randomly generated instances (**complete graphs**).
- 5 *TSP* instances $(n = 5, K = 1)$ and 20 *CVRP* instances $(n = 5, K = 2)$.
- AQC and QGA assessed using **DWave Advantage System 6.3**.
- VQAs evaluated through Qiskit's **statevector simulator**.

TSP Analysis

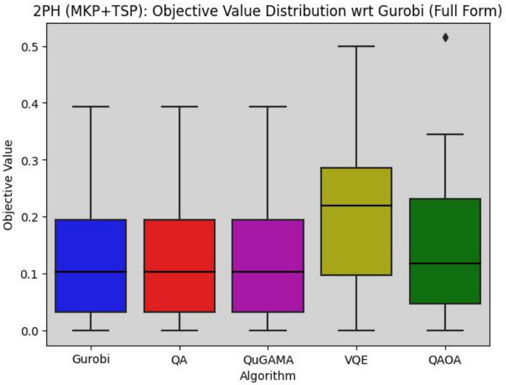
6 Numerical Experiments

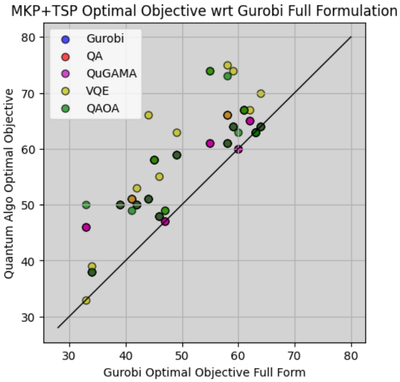




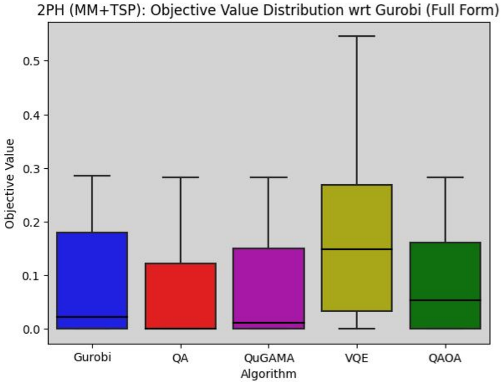
CVRP Analysis - MKP clustering

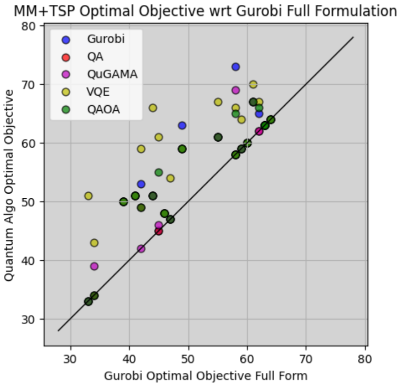
6 Numerical Experiments



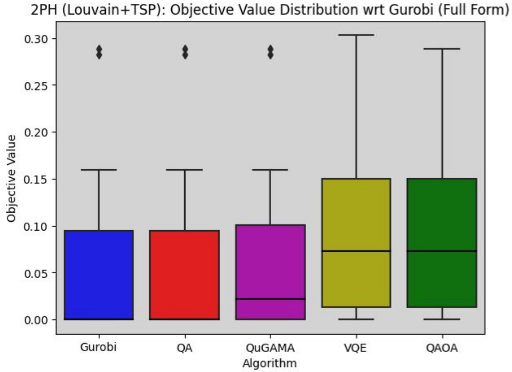


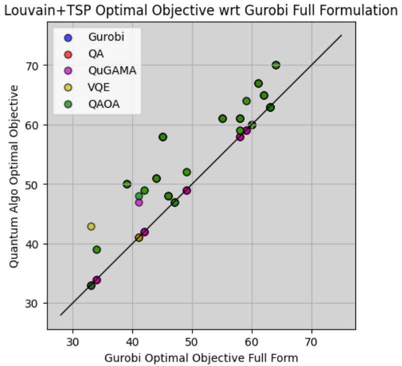
CVRP Analysis - MM clustering





CVRP Analysis - Louvain clustering





CVRP Final Comparison

6 Numerical Experiments

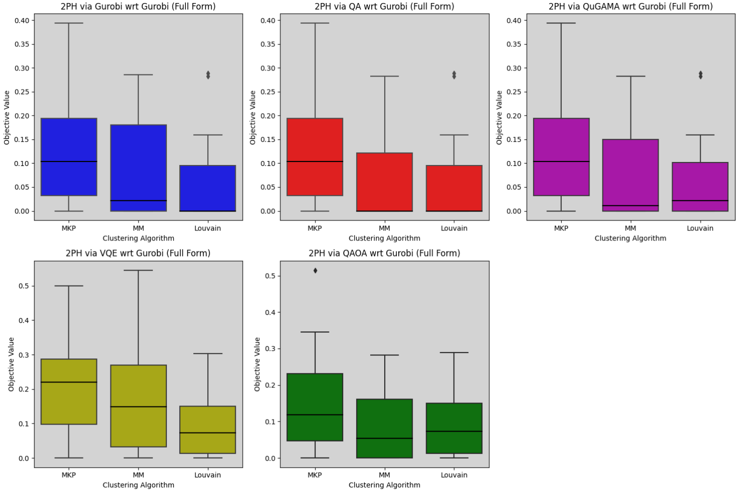


Table of Contents

7 Summary

Concluding Remarks

- AQC and QGA outperform VQAs, often matching Gurobi performance.
- VQAs satisfactory results, but may be impacted by noise on real quantum hardware.
- Louvain and Modularity Maximization clusterings better than MKP.
- Quantum Graver very good; great potential for its test set-based approach.
- Potential future developments: larger instances; warm-starting VQA with ML; refining the clustering phase and so on.

Quantum Integer Programming for the Capacitated Vehicle Routing Problem

Thank you for listening!

Any questions?

Table of Contents

CVRP Full Formulation

$$
\begin{aligned}
\min \quad & \sum_{(i,j)\in A} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{i\in\delta^-(j)} x_{ij} = 1 \qquad \forall j \in V \setminus \{0\} \\
& \sum_{j\in\delta^+(i)} x_{ij} = 1 \qquad \forall i \in V \setminus \{0\} \\
& \sum_{j\in\delta^-(0)} x_{j0} = K \\
& \sum_{j\in\delta^+(0)} x_{0j} = K \\
& u_i - u_j + C x_{ij} \le C - d_j \quad \forall i \ne j \in V \setminus \{0\},\ \text{with } d_i + d_j \le C \\
& d_i \le u_i \le C \qquad \forall i \in V \setminus \{0\} \\
& x_{ij} \in \{0,1\} \ \ \forall (i,j) \in A; \quad u_i \in \mathbb{Z}_{\ge 0} \ \ \forall i \in V \setminus \{0\}.
\end{aligned}
$$

$$
\begin{aligned}
\min \quad & \sum_{k=1}^{K} \sum_{(i,j)\in A} c_{ij} z_{ki} z_{kj} \\
\text{s.t.} \quad & \sum_{k=1}^{K} z_{k0} = K \\
& \sum_{k=1}^{K} z_{ki} = 1 \qquad \forall i \in V \setminus \{0\} \\
& \sum_{\substack{i\in V \\ i\neq 0}} d_i z_{ki} \leq C \quad \forall k = 1, \dots, K \\
& z_{ki} \in \{0, 1\} \qquad \forall k = 1, \dots, K;\ i \in V.
\end{aligned}
$$

MM Clustering

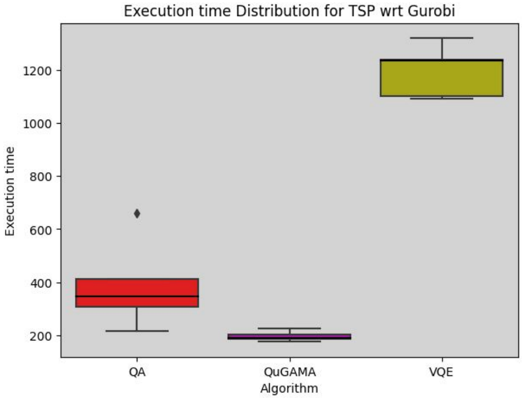
8 Appendix

$$\begin{aligned} \max \quad & \frac{1}{m} \sum_{v} \sum_{i,j} \left(A_{ij} - \frac{k_i^{in} k_j^{out}}{m} \right) z_{v,i} z_{v,j} \\ \text{s.t.} \quad & \sum_{v} z_{v,i} = 1 \quad \forall i = 0, \ldots, n-1 \\ & z_{v,i} \in \{0, 1\} \quad \forall v = 1, \ldots, k;\ i = 0, \ldots, n-1. \end{aligned}$$

$$
\begin{aligned}
\min \quad & \sum_{\substack{(i,j)\in A \\ i,j\in Cluster}} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{\substack{j\in N^+(i) \\ j\in Cluster}} x_{ij} = 1 && \forall i \in Cluster \\
& \sum_{\substack{i\in N^-(j) \\ i\in Cluster}} x_{ij} = 1 && \forall j \in Cluster \\
& \sum_{i\in S} \sum_{j\notin S} x_{ij} \geq 1 && \forall S \subset Cluster \\
& x_{ij} \in \{0, 1\} && \forall i,j \in Cluster.
\end{aligned}
$$

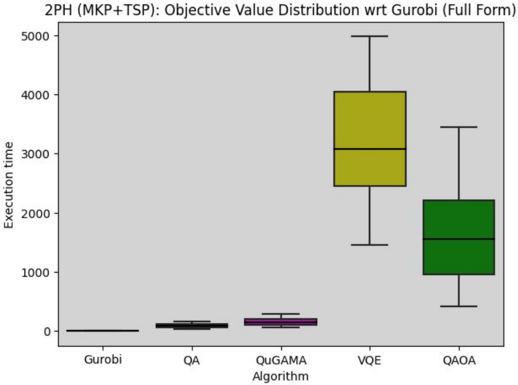
TSP Execution Times

8 Appendix



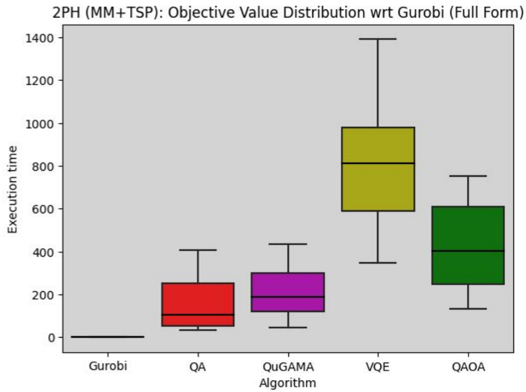
MKP+TSP Execution Times

8 Appendix



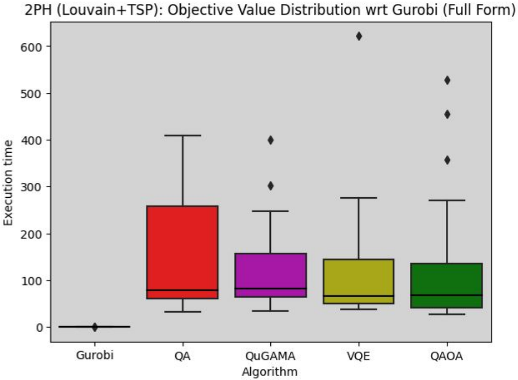
MM+TSP Execution Times

8 Appendix



Louvain+TSP Execution Times

8 Appendix



CVRP Execution Times Final Comparison

8 Appendix