Combinatorial Optimization Using Quantum Computing

Gloria Oppong [1] & Lakmali Weerasena, Ph.D.[1]

[1] University of Tennessee at Chattanooga, 615 McCallie Avenue, Chattanooga, TN 37403-2598

Contact Information:
Gloria Oppong
Phone: (+1)423-475-1397
Email: xqs415@mocs.utc.edu

Research Question

How effective are QAOA-generated warm starts in improving the runtime and solution quality of classical TSP solvers?

Introduction

Combinatorial Optimization (CO) focuses on selecting the best solution from a finite set of feasible solutions. Many important CO problems are computationally difficult and belong to the class of NP-hard problems ([6]). Among these problems, the Traveling Salesman Problem (TSP) is one of the most widely studied optimization problems due to its theoretical importance and broad real-world applications. The TSP consists of:

- a set of cities $N = \{1, 2, \ldots, n\}$, and a pairwise distances d_{ij} between cities i and j,
- The objective is to determine a minimum-cost Hamiltonian tour that visits each city exactly once and returns to the starting city.
- The mathematical model of the TSP using the Miller–Tucker–Zemlin (MTZ) [4] formulation is

$$\min \sum_{(i,j)\in A} c_{ij}y_{ij} \quad \text{s.t.} \quad \sum_{j=1}^{n} y_{ij} = 1, \quad \forall i \in N, \quad \sum_{i=1}^{n} y_{ij} = 1, \quad \forall j \in N, \quad u_i - u_j + ny_{ij} \leq n-1,$$
$$\forall i \neq j,\ i,j \in \{2,\ldots,n\}, \quad u_i \geq 0, \quad \forall i = 2,\ldots,n, \quad y_{ij} \in \{0,1\}, \quad \forall (i,j) \in A.$$

- $y_{ij} = 1$ if the tour travels directly from city i to city j, and 0 otherwise.
- c_{ij} represents the travel cost (distance) from city i to city j.
- u_i are auxiliary variables used to eliminate subtours.
- N is the set of cities.
- A is the set of edges connecting the cities.
- The MTZ constraints ensure that the solution forms a **single Hamiltonian cycle** .

Applications of TSP

- Logistics and supply-chain routing
- Robotics and UAV path planning
- Healthcare home-visit routing
- Manufacturing optimization

Why TSP

- Canonical NP-hard benchmark
- Standardized datasets available
- Naturally maps into QUBO form
- Suitable for hybrid quantum workflows

Goal

Our goal is to investigate how the hybrid quantum–classical optimization methods, particularly QAOA-generated warm-start solutions, for improving the runtime and solution quality of classical TSP solvers.

Method

To implement the Quantum Algorithms for solving TSP, we represent TSP as a QUBO form.

Quadratic Unconstrained Binary Optimization (QUBO) Formulation [3]

- The QUBO formulation transforms the constrained TSP into an **unconstrained binary optimization** problem suitable for QAOA. The QUBO Hamiltonian is expressed as;

$$H_{QUBO} = H_{obj} + H_{city} + H_{pos}$$

$$H_{obj} = b\sum_{j=1}^{n}\sum_{u=1}^{n}\sum_{v=1}^{n} d_{uv}\hat{y}_{u,j}\hat{y}_{v,j+1}$$

$$H_{city} = a\sum_{v=1}^{n}\left(1-\sum_{j=1}^{n}\hat{y}_{v,j}\right)^2$$

$$H_{pos} = a\sum_{j=1}^{n}\left(1-\sum_{v=1}^{n}\hat{y}_{v,j}\right)^2$$

- $\hat{y}_{u,j} = 1$ if city u is visited at position j , 0 otherwise.
- d_{uv}: distance between cities u and v.
- H_{obj}: minimizes the travel distance of the tour.
- H_{city}: ensures each city is visited exactly once.
- H_{pos}: ensures each tour position is occupied by exactly one city.
- a: penalty coefficient enforcing feasibility constraints.
- b: weighting coefficient.

Quantum Approximate Optimization Algorithm (QAOA)

- QAOA begins with an equal superposition state: $|\psi_0\rangle = |+\rangle^{\otimes n^2}$
- The parameterized quantum state is constructed as: $|\psi_p(\vec{\gamma}, \vec{\beta})\rangle = \prod_{k=1}^{p} e^{-i\beta_k H_B} e^{-i\gamma_k H_C}|+\rangle^{\otimes n^2}$
- The classical optimizer updates the parameters $(\vec{\gamma}, \vec{\beta})$ to minimize: $E(\vec{\gamma}, \vec{\beta}) = \langle\psi_p|H_C|\psi_p\rangle$
- H_C is the cost Hamiltonian derived from the QUBO formulation.
- H_B is the mixer Hamiltonian used to explore the search space.

Feasibility Repair Algorithm for QAOA Outputs

1: Input: Raw QAOA samples S, distance matrix D, penalty parameter A, Top-K
2: Initialization: *feasible_solutions* $= \emptyset$
3: for each sample $s \in S$ **do**
4: Extract city–position assignment from s
5: *Constraint Check:* verify one-city-per-position and one-position-per-city
6: if constraints are violated **then**
7: Repair assignment using probability structure of s
8: Ensure a valid permutation matrix
9: verify Hamiltonian cycle feasibility
10: end if
11: Compute tour cost: cost $= \sum_{p=1}^{n} d_{s_p, s_{p+1}}$
12: Add repaired tour to *feasible solutions*
13: end for

Hybrid Quantum–Classical Workflow

- **Qiskit** is an open-source framework for building and running quantum circuits.
- We implement QAOA for the QUBO and evaluate the resulting cost.
- A classical optimizer updates $(\vec{\gamma}, \vec{\beta})$ and we decode the final bitstring into a tour.

Graphical Representation of Qiskit Hybrid Workflow

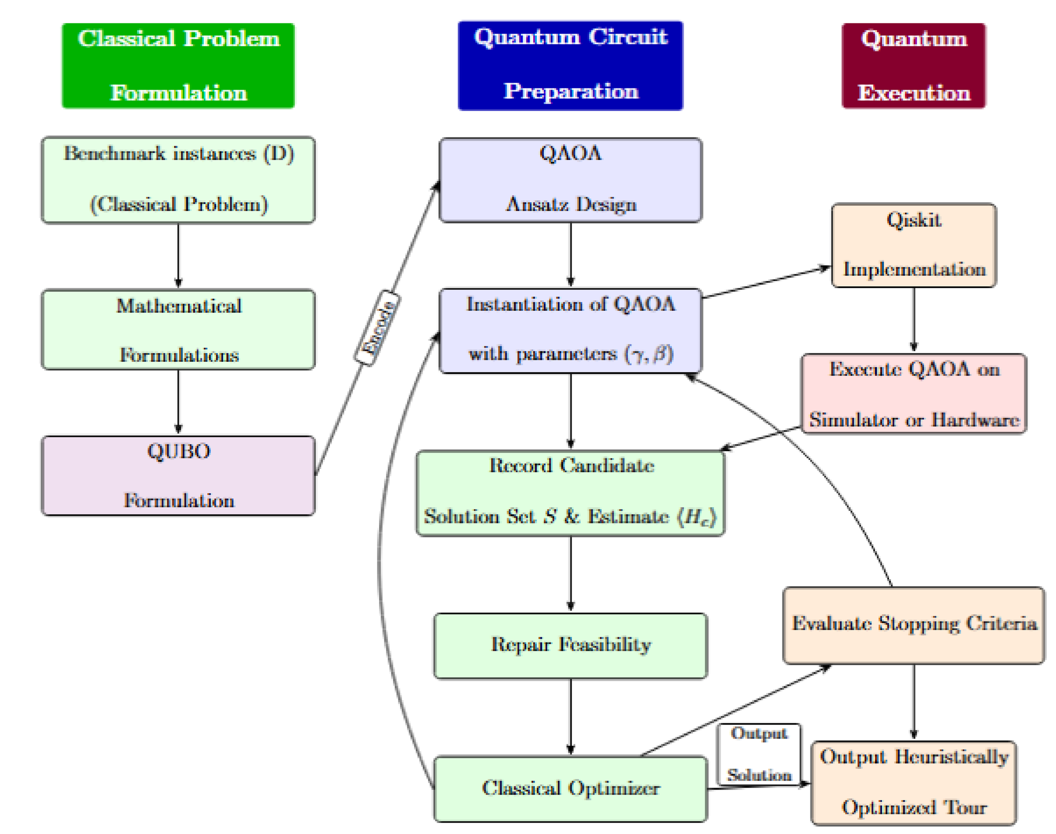

Figure 1: Qiskit-Based Quantum-Classical Hybrid Workflow

Computational Analysis

To evaluate the effectiveness of the proposed hybrid quantum–classical framework, we conducted computational experiments based on:

Datasets

- 49 UK city coordinates [2].
- Qoptlib benchmark instances [5].
- OR-Library `vrp8` dataset [1].

Experimental Setup

- All quantum experiments were implemented in **Qiskit**
- **QAOA**: Depth $p = 1$, 16–2048 shots.
- Exact solutions obtained using **Gurobi**.

Performance Metrics

- Objective value (tour cost)
- Optimality gap
- Primal gap
- Approximation ratio

Results

1. The hybrid quantum–classical workflow demonstrated that quantum-generated solutions can be incorporated into exact classical optimization procedures.
2. COBYLA consistently achieved the best balance between runtime and solution quality and remained scalable to larger problem instances, whereas several optimizers failed on larger cases.
3. Classical solvers (Gurobi) continued to outperform quantum approaches on large-scale instances; however, QAOA showed promise as a practical component of hybrid optimization workflows.
4. The results support the use of QAOA-generated warm starts and feasibility-repair strategies for near-term quantum optimization research.

Table 1: Comparison of QAOA optimizers using 16 shots

n	Qubits	COBYLA		NFT		SPSA		L_BFGS_B		SLSQP	
		Obj_val	T(s)	Obj_val	T(s)	Obj_val	T(s)	Obj_val	T(s)	Obj_val	T(s)
15	225	3398.333	859	3415.836	23,925	3052.266	34,849	2729.352	42,121	3307.930	20,846
20	400	4124.258	2,563	4849.531	75,969	4912.992	117,247	4409.706	123,617	3169.664	37,496
25	625	5453.940	19,527	5398.553	279,214	–	–	5392.846	159,885	5256.665	114,923
30	900	6353.938	35,797	–	–	–	–	–	–	6381.661	187,510
35	1,225	7877.402	75,521	–	–	–	–	–	–	–	–
40	1,600	8399.340	127,006	–	–	–	–	–	–	–	–

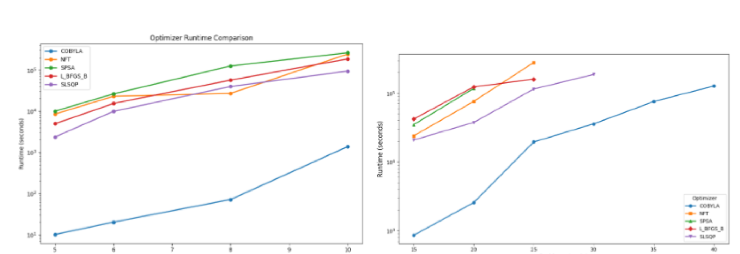

Figure 2: optimizer runtime comparison.

Table 2: Solution Quality Metrics (UK Dataset)

n	Exact obj-val	QAOA obj-val	QAOA T(s)	Gap(%)	Approx. ratio	Primal gap(%)
5	1129.93	1296.32	10	12.20	1.12	10.87
6	1132.03	1220.76	20	7.84	1.08	7.27
8	1326.49	1573.88	71	18.65	1.19	15.72
10	1517.87	2044.52	1377	46.03	1.35	25.76
15	1473.16	3398.33	859	130.68	2.31	56.65
20	1676.85	4124.26	2563	145.95	2.46	59.34
25	1930.53	5453.94	19527	182.51	2.83	64.60
30	2196.98	6353.94	35797	189.21	2.90	65.42
35	2521.04	7877.40	75521	212.47	3.13	68.00
40	2561.39	8399.34	127006	227.92	3.28	69.50

Table 3: Solution Quality Metrics (VRP8 Dataset)

n	Exact obj-val	QAOA obj-val	QAOA T(s)	Gap(%)	Approx. ratio	Primal gap(%)
5	106.99	144.40	16.301	34.97	1.35	25.91
6	113.71	148.80	31.125	30.86	1.31	23.58
8	139.14	188.30	68.886	35.33	1.35	26.11
10	160.65	212.40	174.905	32.21	1.32	24.36
15	208.01	418.80	1086.785	101.34	2.01	50.33
20	243.18	565.50	3070.260	132.54	2.33	57.00
25	278.99	729.00	9989.428	161.30	2.61	61.73
35	335.10	1051.30	55889.220	213.73	3.14	68.13
40	373.08	1289.74	112050.523	245.70	3.46	71.07

Table 4: Solution Quality Metrics (QOptLib Dataset)

n	Exact obj-val	QAOA obj-val	QAOA T(s)	Gap(%)	Approx. ratio	Primal gap(%)
4	6700	6700	9.021	0.00	1.00	0.00
5	6786	8145.00	17.298	20.03	1.20	16.69
6	9814	12528.00	20.923	27.65	1.28	21.66
7	7245	9915.00	45.098	36.85	1.37	26.93
8	2762	4466.00	71.622	61.69	1.62	38.15
9	2134	2817.00	119.010	32.01	1.32	24.25
10	2822	5532.00	175.155	96.03	1.96	48.99
15	3237	7172.00	773.440	121.56	2.23	54.87
22	4106	10469.00	4403.874	154.97	2.55	60.78
25	26443	65862.00	7940.506	149.07	2.49	59.85

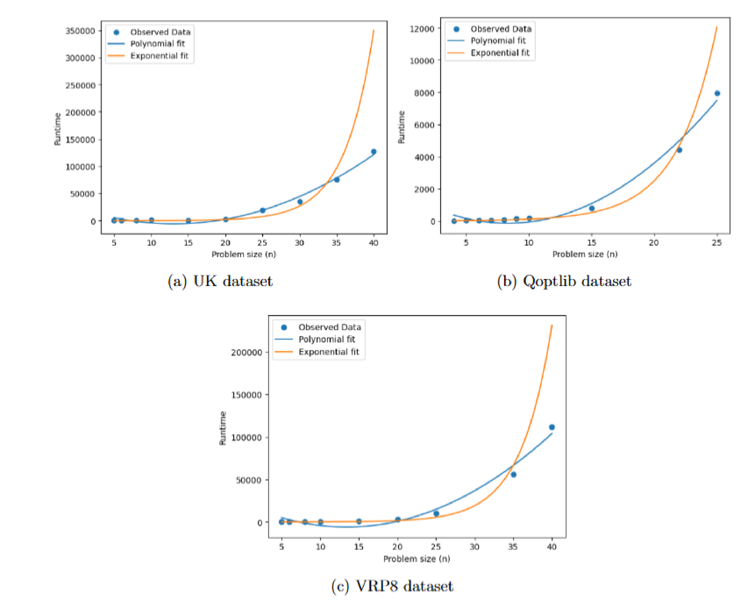

Figure 3: QAOA runtime comparison.

Table 5: Solution Quality Metrics (QOptLib Dataset)

2*n	UK dataset		Qoptlib dataset		VRP8 dataset	
	Exact	Warm-start	Exact	Warm-start	Exact	Warm-start
5	0.7700	0.0551	0.1238	0.0613	0.0500	0.0410
6	0.6800	0.1194	0.1218	0.0522	0.0631	0.0398
8	3.2800	2.7222	0.3272	0.2270	0.0672	0.0454
10	5.1500	3.2443	0.7395	0.4750	0.0812	0.0583
15	6.0400	4.8081	10.6081	11.1664	0.0941	0.0653
20	14.2300	7.7272	41.8178	12.9225	0.2577	0.0907
25	26.5700	28.6493	241.5364	120.4260	0.5921	0.2539
35	27.4400	50.4104	NA	NA	2.0051	1.7239
40	59.2710	98.4776	NA	NA	2.0858	1.8260

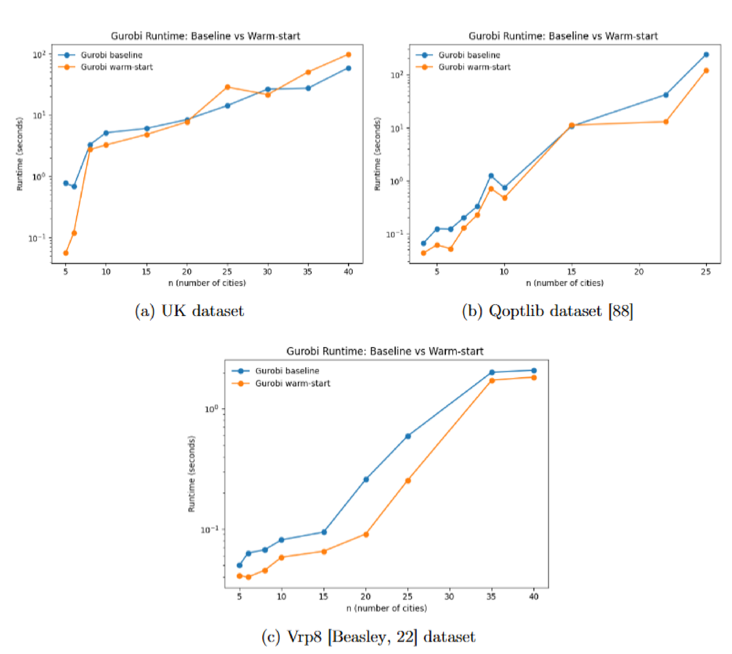

Figure 4: QAOA runtime comparison.

Conclusion and Future work

- Approximation ratios stay within a constant factor (below **3.5**) of the best-known solutions.
- QAOA simulator runtime increased sharply with n and became the dominant computational cost.
- The warm-start consistently improved Gurobi computational time across the tested cases.
- Future work will investigate larger benchmark instances, advanced QAOA variants, improved warm-start strategies, and execution on real quantum hardware.

References

[1] J. E. Beasley. OR-Library: Information page. https://people.brunel.ac.uk/ mastjjb/jeb/info.html. Accessed: 2026-02-09.

[2] Patrick Ford. Travelling salesman problem: Visit 49 uk cities., 2024.

[3] Andrew Lucas. Ising formulations of many np problems. *Frontiers in Physics*, 2:5, 2014.

[4] Clair E Miller, Albert W Tucker, and Richard A Zemlin. Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM)*, 7(4):326–329, 1960.

[5] Eneko Osaba and Esther Villar-Rodriguez. Qoptlib: a quantum computing oriented benchmark for combinatorial optimization problems. In *Benchmarks and Hybrid Algorithms in Optimization and Applications*, pages 49–63. Springer, 2023.

[6] Christos H Papadimitriou and Kenneth Steiglitz. *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998.

ACKNOWLEDGEMENTS : I would like to thank my supervisor, Dr. Lakmali Weerasena (Department of Mathematics, UTC), and all professors in the Mathematics Department, especially Deborah Barr, for their guidance and valuable contributions to this research.