Smart Traffic Optimization in the New Capital: Classical and Quantum Approaches

By

Team 1:
Alshimaa Sobhy Saad Salama
Basel Abdel-Maqsoud Abdel-Maqsoud Ziada
Enas Emad Ahmed Helal Shams Eldin
Mo'men Mohamed Ramadan Mohamed
Abdelrhman Zain Mohamed Ahmed

A Technical Report Submitted to In Partial Fulfillment of the Requirements for the Degree of The Quantum Hackathon

Mentors

Dr. Nouhaila Innan

Eng. Abdelrahman Elsayed

August 2025

Contents

Chapter 1

Problem Definition

1.1 Motivation

The Vehicle Routing Problem (VRP) represents one of the world's largest operational challenges [1], with its global market estimated at approximately USD 8.2 trillion in 2015. Typical VRP scenarios involve a fleet of vehicles (such as trucks, ambulances, or container ships), depots where these vehicles are stationed, and multiple client locations that must be served within given constraints. The central computational challenge is to design efficient routes that start and end at depots while visiting a set of locations, with the goal of minimizing total travel distance, travel time, or other performance metrics [2].

In this work, we consider a specialized instance of the VRP that is highly relevant to modern smart-city infrastructure: **Emergency Patient Transportation in the New Capital City.** In this scenario, traffic optimization plays a critical role in sustainability, fuel efficiency, and public well-being. We study the case of urgent patient transfers where a single ambulance must transport 5 patients to a central hospital.

Problem Statement. We consider the case of a single ambulance that must transport five patients to a central hospital under emergency conditions. The ambulance is allowed to perform multiple trips, but each trip can accommodate at most three patients due to capacity constraints. All patients must be delivered safely to the hospital, and travel distances are computed using real GPS coordinates for both the hospital and the patient locations.

Objective. The goal is to determine an optimal routing strategy that minimizes the total travel distance required to serve all patients, while respecting the capacity and feasibility constraints of the problem.

Dataset. The instance data (hospital and patient coordinates, pairwise distances) are provided in the JSON file: `OptimizationProblemData.json`. Distance calculations based on GPS coordinates[2].

1.2 Data Description

The dataset used in the problem corresponds to an emergency patient transportation scenario. The problem is formally defined as follows:

- **Title:** Emergency Patient Transportation System
- **Objective:** Minimize the total distance traveled while transporting all patients to the central hospital.
- **Constraints:**
  - Number of ambulances: 1
  - Maximum capacity per stop: 1 patient
  - Maximum stops per trip: 3
  - Total patients: 5

The central hospital is located at the coordinates (latitude: 29.9951, longitude: 31.6846) and serves as the final destination for all trips.

The dataset also contains the spatial information of five patients requiring transport:

1. Patient DT: (30.0004, 31.7396)
2. Patient GR: (30.0113, 31.7478)
3. Patient R2: (30.0303, 31.6692)
4. Patient R3_2: (30.0309, 31.6883)
5. Patient IT: (30.0128, 31.6938)

All patients are classified with a *normal* priority, and each trip is required to respect both the vehicle's capacity and the maximum number of stops, which is also defined. This structured dataset forms the basis for formulating and solving the optimization problem.

1.3 Data Extraction & Preprocessing

The raw input data consisted of geographic coordinates for the central hospital and five patient locations. Several preprocessing steps were applied to prepare the data for optimization:

1. **Coordinate Collection:** Each location (hospital and patients) was represented by its latitude and longitude values.
2. **Distance Matrix Construction:** Using the Open Source Routing Machine (OSRM) API, pairwise road-network distances were computed between all locations. This includes:
   - Hospital $\leftrightarrow$ each patient,

- Each patient ↔ every other patient.

Distances were returned in meters and converted to kilometers for consistency. The computed cost matrix represents directed road-network distances, meaning that the distance from location i to location j may differ from j to i.

This asymmetry arises due to road layouts, one-way streets, or routing constraints. Table 1.1 shows the directed distances (in kilometers), with rows denoting the **origin (From)** and columns denoting the **destination (To)**.

Table 1.1: Directed pairwise distance matrix between hospital and patients (in km).

From ↓ / To →	Hospital	DT	GR	R2	R3_2	IT
Hospital	0.00	14.19	17.78	11.86	7.34	9.27
DT	8.63	0.00	7.75	19.67	12.17	9.40
GR	11.50	2.36	0.00	15.66	10.05	12.27
R2	9.45	10.92	11.81	0.00	4.07	7.32
R3_2	10.85	9.24	10.12	11.57	0.00	8.72
IT	9.67	9.43	10.48	11.54	5.93	0.00

From the directed matrix, the following insights can be drawn:

- The shortest hospital-to-patient leg is **Hospital** → **R3_2** at 7.34 km, whereas the reverse leg **R3_2** → **Hospital** is 10.85 km.
- The closest patient-to-patient direction is **GR** → **DT** at 2.36 km, but the reverse direction **DT** → **GR** is 7.75 km.
- The longest directional connection is **DT** → **R2** at 19.67 km.

This directed structure is essential for modeling realistic ambulance routes, since travel times and distances differ depending on the direction of travel.

This preprocessing pipeline ensured that the problem instance was fully specified with all necessary distances and cost structures, ready for optimization.

Chapter 2

Proposed Methodology

2.1 Overview of the Pipeline

The proposed pipeline provides an end-to-end framework for addressing the ambulance routing problem. The methodology is structured in two complementary phases: a classical baseline and a quantum-inspired approach.

In the first phase, we implement a **classical brute-force optimization** method. This exhaustive search examines all possible patient assignments and trip permutations within the specified capacity constraint, ensuring the discovery of the globally optimal solution. Although computationally expensive in larger problem instances, this step serves as a benchmark for validating and comparing the performance of quantum techniques.

In the second phase, we reformulate the problem as a **Quadratic Unconstrained Binary Optimization (QUBO)** model. The QUBO representation allows us to leverage quantum and quantum-inspired solvers. This phase is further divided into two components:

1. **Quantum Annealing:** The problem is solved using D-Wave libraries, where the annealing process searches for low-energy states that correspond to near-optimal routing plans.

2. **Quantum Approximate Optimization Algorithm (QAOA):** The problem is also implemented in a gate-based quantum framework, where QAOA iteratively optimizes parameterized quantum circuits to approximate the optimal solution.

This two-tiered design ensures both rigor and innovation: the brute-force method guarantees correctness on small instances, while the QUBO-based formulations enable scalability and exploration of quantum advantages in routing optimization.

2.2 Classical Brute-Force Approach

In addition to quantum methods, we implemented a classical brute-force optimization approach to solve the emergency patient transportation problem. The goal is to minimize the total distance traveled while transporting all patients to the hospital, considering a maximum of 3 stops per trip.

2.2.1 Methodology

The classical approach uses the following steps:

1. Generate all permutations of patients to consider every possible order of service.
2. Generate all valid splits of patients into trips, respecting the maximum capacity per trip.
3. Compute the total distance for each combination of permutation and trip split using the distance cost matrix $C(a, b)$:

$$\text{Cost(trip)} = C(\text{Hospital}, p_1) + \sum_{i=1}^{k-1} C(p_i, p_{i+1}) + C(p_k, \text{Hospital}) \tag{2.1}$$

where $p_1, \ldots, p_k$ are patients in a single trip.

4. Select the plan with the lowest total distance.

2.2.2 Results

Applying the brute-force approach on our dataset, the optimal plan and cost were obtained as follows:

- Optimal plan: $[(\mathrm{R2}, \mathrm{IT}), (\mathrm{R3\_2}, \mathrm{GR}, \mathrm{DT})]$
- Total distance: 57.30 km
- Computation time: 0.0065 s

2.3 Quantum Approaches

Quantum annealing is a way to solve optimization problems using qubits. At the start, the qubits are placed in a mixed state, and then the system is slowly adjusted so that it matches the problem we want to solve. By carefully reducing randomness, the qubits settle into the lowest-energy state, which represents the best solution. The final qubit values are then read out as the answer. Problems are usually written in forms like the Ising model or QUBO. [3].

2.3.1 Annealing-based Methods: Using quantum annealers (D-Wave).

Let $\mathcal{P} = \{DT, GR, R2, R3\_2, IT\}$ denote the set of patients. Let T be the number of trips (here $T = 2$) as. Let $y_{p,t} \in \{0, 1\}$ be a binary variable equal to 1 if patient p is assigned to trip t. Let $s_{t,k} \in \{0, 1\}$ be a slack binary variable for trip t, where $k = 0, 1, \ldots, K-1$ indexes the slack bits, and each bit has weight 2^k. These variables together encode the overflow amount beyond the vehicle capacity. Let d_{uv} denote the distance (cost) from node u to node v given by the cost_matrix.

Let $\text{saving}_{ij} \geq 0$ denote the computed pairwise saving for placing patients i and j in the same trip.

Binary quadratic model (BQM) energy. The energy function minimized in the BQM is

$$E(\mathbf{y},\mathbf{s}) = \underbrace{-\sum_{t\in\mathcal{T}}\sum_{i<j}\text{saving}_{ij}\, y_{i,t}\, y_{j,t}}_{\text{pairwise savings (encourage grouping)}} + \underbrace{B_{\text{assign}}\sum_{p\in\mathcal{P}}\left(\sum_{t\in\mathcal{T}} y_{p,t} - 1\right)^2}_{\text{assignment constraint (each patient assigned once)}}$$

$$+ \underbrace{B_{\text{cap}}\sum_{t\in\mathcal{T}}\left(\sum_{p\in\mathcal{P}} y_{p,t} - C - \sum_{k=0}^{K-1} 2^k s_{t,k}\right)^2}_{\text{capacity constraint (with slack bits)}} + \underbrace{M_{\text{onehot}}\sum_{p\in\mathcal{P}}\sum_{t<t'} y_{p,t}\, y_{p,t'}}_{\text{strong penalty preventing double assignment}} .$$

Variables.

- $\mathcal{P}$: set of patients.
- $\mathcal{T}$: set of available trips.
- $y_{p,t} \in \{0,1\}$: binary assignment variable, equal to 1 if patient p is assigned to trip t.
- $s_{t,k} \in \{0,1\}$: slack bit for trip t representing overflow capacity in binary, with weight 2^k.
- C: vehicle capacity (here $C = 3$).
- saving_{ij}: cost saving obtained if patients i and j share the same trip (defined in the pairwise savings step).
- $B_{\text{assign}}, B_{\text{cap}}, M_{\text{onehot}}$: penalty weights ensuring constraints dominate the objective.

Expansion of penalty terms. Each squared penalty $(\cdot)^2$ is expanded into linear and quadratic coefficients when building the BQM. For example,

$$\left(\sum_{t\in\mathcal{T}} y_{p,t} - 1\right)^2 = \sum_t y_{p,t} - 2\sum_t y_{p,t} + 1 + 2\sum_{t<t'} y_{p,t}y_{p,t'}.$$

Thus, the assignment penalty contributes linear terms $+B_{\text{assign}}$ for each $y_{p,t}$, and quadratic terms $+2B_{\text{assign}}\, y_{p,t}y_{p,t'}$ for $t \neq t'$.

Optimization procedure.

1. Compute pairwise savings saving_{ij} from the cost matrix.
2. Define binary variables $y_{p,t}$ for patient–trip assignments.
3. Determine required slack bits K and introduce slack variables $s_{t,k}$.
4. Construct the BQM energy as described above.
5. Sample the BQM using a solver (here: `SimulatedAnnealingSampler`).
6. Decode the best solution into patient groups $\{\, p \mid y_{p,t} = 1\,\}$ for each trip t.
7. Verify feasibility:
   - each patient is assigned exactly once,
   - each trip respects capacity after accounting for slack.
8. For each trip, compute the optimal visit order (classical sequencing) by enumerating permutations of assigned patients and choosing the one with minimum travel cost:

$$\text{trip\_cost}(\pi) \;=\; d_{\text{Hospital},\pi_1} + \sum_{i=1}^{|\pi|-1} d_{\pi_i,\pi_{i+1}} + d_{\pi_{|\pi|},\text{Hospital}}.$$

9. Sum the trip costs to obtain the total travel cost of the solution.

Remarks. This BQM formulation enforces feasibility through penalties, while savings terms encourage efficient pairings. Explicit route-ordering variables are not used in the quantum model; instead, sequencing is computed classically in a post-processing step.

2.3.2 Quantum Approximate Optimization Algorithm (QAOA)

QAOA is a gate-model approach for solving optimization problems by mapping a Quadratic Unconstrained Binary Optimization (QUBO) formulation into a quantum Hamiltonian. The main steps of our formulation are as follows:

1. Problem Encoding (QUBO)

Each patient-to-trip assignment is represented by binary variables $y_{p,t}$.

- **Constraint A (assignment equality):** each patient must be assigned to exactly one trip:
$$B_{\text{assign}} \sum_{p} \left(\sum_{t} y_{p,t} - 1 \right)^2$$
- **Constraint B (capacity with slack bits):** each trip must respect capacity limits, using binary slack variables $s_{t,k}$ to enforce exact equality:
$$B_{\text{cap}} \sum_{t} \left(\sum_{p} y_{p,t} - C_t - \sum_{k} 2^k s_{t,k} \right)^2$$

The full QUBO objective is then:

$$E(\mathbf{y}, \mathbf{s}) = (\text{trip distances}) \;+\; \text{Constraint A} \;+\; \text{Constraint B}.$$

2. QUBO-to-Ising Conversion

The QUBO matrix Q is mapped into an equivalent Ising Hamiltonian expressed as weighted sums of Z and ZZ operators. This Hamiltonian forms the cost operator in QAOA.

3. QAOA Circuit Construction

We initialize all qubits in a uniform superposition using Hadamard gates. Then we apply alternating layers:

- **Cost operator:** rotations according to Ising terms, which encode the optimization problem.
- **Mixer operator:** RX rotations to explore new candidate solutions.

The circuit depth p determines the number of alternating layers.

4. Classical Optimization Loop

Circuit parameters (γ, β) are optimized using a classical optimizer (COBYLA in our case) to minimize the expected energy:

$$\langle \psi(\gamma, \beta) \mid H_{\text{cost}} \mid \psi(\gamma, \beta) \rangle.$$

5. Results Interpretation

The most frequent bitstrings measured from QAOA are decoded back into patient–trip assignments and slack values. Valid assignments (satisfying both constraints) yield feasible routing solutions, from which we compute the total travel distance. In our test case, QAOA identified valid assignments with feasible slack, and corresponding trip routes from the hospital to patients and back.

The problem instance was represented with a total of 14 binary variables, corresponding to patient–trip assignment indicators and slack variables: `['y_DT_0', 'y_DT_1', 'y_GR_0', 'y_GR_1', 'y_R2_0', 'y_R2_1', 'y_R3_2_0', 'y_R3_2_1', 'y_IT_0', 'y_IT_1', 's_0_0', 's_0_1', 's_1_0', 's_1_1']`. The constructed QUBO had shape $(14, 14)$ with 14 linear terms and 94 quadratic terms, which was mapped into 61 Ising terms (single-qubit Z and two-qubit ZZ operators). When running the simulation with Qiskit Aer, the QAOA parameters were optimized using COBYLA, converging after 34 function evaluations with a minimum cost value of -6311.86. The optimal parameter set was $\theta = [3.56, 3.40, 1.56, 6.23]$, yielding a best simulation distance of 58.30 km.

QAOA Results on Real Quantum Hardware

The QAOA optimization was executed on IBM's superconducting quantum processor `ibm_brisbane`. After transpilation and execution, the best measured bitstring corresponds to a valid assignment and sequencing:

$$\text{Best bitstring: } 10011001100000, \quad E = -15192.34, \quad \text{distance} = 67.20 \text{ km}.$$

$$\text{Trip 1: Hospital} \rightarrow R2 \rightarrow IT \rightarrow DT \rightarrow \text{Hospital} \quad (38.24 \text{ km}),$$

$$\text{Trip 2: Hospital} \rightarrow R3\_2 \rightarrow GR \rightarrow \text{Hospital} \quad (28.96 \text{ km}).$$

Comparison with Simulation

	Simulation	Real Hardware
Bestdistance	66.05 km	67.20 km
Feasibility	*Valid*	*Valid*
Difference	1.15 km	

The real hardware solution was within 1.15 km of the best simulation result, demonstrating that QAOA on quantum hardware can produce feasible and near-optimal vehicle routing solutions.

Chapter 3

Discussion and Future Work

3.1 Strengths and Limitations

This work demonstrates the feasibility of applying both classical and quantum approaches to an ambulance routing problem formulated as a variant of the Vehicle Routing Problem (VRP). A key strength of the study is the two-phase pipeline design: the brute-force classical solver guarantees optimality in small-scale instances, while the QUBO-based quantum formulations enable exploration of scalable and innovative solution strategies. The integration of real GPS-based distances provides a realistic foundation, ensuring that the problem setup reflects practical urban conditions.

However, limitations remain. First, the brute-force baseline is computationally intractable for larger instances, restricting its applicability to very small problem sizes. Second, the QAOA implementation was constrained to simulator runs due to backend incompatibility with real quantum hardware. Additionally, the limited depth of the QAOA circuit and the choice of a single classical optimizer (COBYLA) may have restricted the quality of solutions. Finally, noise, connectivity restrictions, and qubit count limitations of current devices further hinder the scalability of the quantum solutions.

3.2 Potential Improvements

Several enhancements can be made to strengthen the framework:

- **Hybrid algorithms:** Combining quantum optimization with classical post-processing (e.g., local search, heuristics) could yield better-quality solutions for larger problems.
- **Better embedding:** In the annealing-based formulation, optimized embeddings onto D-Wave hardware could reduce overhead and improve solution quality.
- **Error mitigation:** Gate-model implementations (QAOA) can benefit from error mitigation techniques such as zero-noise extrapolation or measurement error calibration to reduce the impact of device noise.
- **Optimizer tuning:** Exploring different classical optimizers (e.g., SPSA, Adam, or Bayesian optimization) could improve convergence compared to COBYLA alone.

3.3 Future Research Directions

Looking ahead, several avenues for research emerge:

1. **Scaling to larger instances:** Extending the formulation to more patients, multiple ambulances, and realistic traffic conditions will test scalability.
2. **Multi-objective optimization:** Incorporating additional objectives (e.g., travel time, fuel consumption, or patient priority levels) to reflect practical ambulance dispatch scenarios.
3. **Dynamic routing:** Investigating real-time updates where new patient requests arrive dynamically, requiring online re-optimization.

4. **Quantum–classical synergy:** Developing hybrid methods that offload specific subproblems to quantum solvers while leveraging efficient classical heuristics for the remaining structure.

5. **Hardware experiments:** As quantum hardware matures, executing the formulations on real devices with error-mitigation strategies will be essential for validating practical performance.

3.4 Conclusion

This work addressed the emergency patient transportation problem in the framework of the Vehicle Routing Problem (VRP), focusing on a single ambulance serving five patients under capacity constraints. We constructed a realistic dataset using GPS-based distances, formulated the optimization task as both a classical brute-force search and a quantum optimization problem, and evaluated the performance of both approaches.

The brute-force solver provided the globally optimal solution with a total distance of 57.30 km, serving as a benchmark. The QUBO-based approaches, implemented via D-Wave (annealing) and QAOA (gate-model simulation), demonstrated the feasibility of encoding and solving such routing problems on quantum frameworks. In the QAOA simulation, a best route distance of 58.30 km was achieved, highlighting competitiveness with the classical optimum despite current limitations in hardware and algorithmic depth.

Overall, the study underscores both the promise and challenges of quantum optimization for logistics applica

Bibliography

[1] Paolo Toth and Daniele Vigo. *The vehicle routing problem*. SIAM, 2002.

[2] qiskit-community. "Vehicle Routing". In: *Qiskit Optimization 0.7.0* (2025).

[3] Thomas Krauss and Joey McCollum. "Solving the network shortest path problem on a quantum annealer". In: *IEEE Transactions on Quantum Engineering* 1 (2020), pp. 1–12.