Traffic Congestion Management During Urban Events

Submitted By:
Name - Garima
Roll No. - 21075033
Branch - Computer Science and Engineering

Problem Statement:

The goal is to manage traffic congestion during urban events by minimizing total travel time, balancing traffic load, and ensuring emergency vehicle prioritization. The system should dynamically adapt to real-time data to optimize traffic signal timings and reroute traffic based on evolving congestion patterns. The problem can be modeled as a Quadratic Unconstrained Binary Optimization (QUBO) problem and solved using a hybrid quantum-classical approach.

Solution:

We can employ a **hybrid quantum-classical approach** to solve the problem, involving:

1. **Classical Preprocessing:** Prepares road network, vehicle, event, and real-time traffic data.
2. **Quadratic Unconstrained Binary Optimization (QUBO):** Formulation of the problem to meet the objectives while keeping in mind the constraints of the problem.
3. **Quantum Optimization:** Use quantum annealing to minimize the QUBO function, optimizing vehicle routes and traffic signal timings.
4. Get optimized traffic signal timings and alternate routes for vehicles

This method is effective in large-scale, real-time traffic optimization scenarios, especially during high-demand events, leveraging quantum computing for efficient solutions.

Why QUBO?

The Quantum annealing technology provided by D-Wave Systems Inc., such as Quantum Processing Units (QPUs) and Hybrid Quantum Processing Units (HQPUs), is designed to solve complex combinatorial optimization problems. The QPU is specifically designed to solve the Ising model, which is an NP-hard problem.

This Ising model is equivalent to solving Quadratic Unconstrained Binary Optimization (QUBO) problems, represented as:

$$\mathrm{Obj}_{\mathrm{QUBO}}(x, Q) = x^T \cdot Q \cdot x$$

where x symbolizes a vector comprising binary variables, while Q is an N × N real-valued matrix delineating the interdependence among these variables. Each qubit in the QUBO problem corresponds to a variable in x, and the coefficients in Q represent the costs associated with qubit pairs. The QPU is a implementation of an undirected graph, with qubits as vertices and couplers as edges between them.

Reducing Congestion and load balancing:

Now, in order to solve this problem, we divide the problem into a chronological workflow:

1. Classical: preprocess map and GPS data.
2. Classical: identify areas where traffic congestion occurs.
3. Classical: determine spatially and temporally valid alternative routes for each car in the data set, if possible.
4. Classical: formulate the minimization problem as a QUBO (to minimize congestion in road segments on overlapping routes).
5. Hybrid quantum/classical: find a solution that reduces congestion among route assignments in the whole traffic graph.
6. Classical: redistribute the cars based on the results.
7. Iterate over steps 2–6 until no traffic congestion is identified.

Algorithm for reducing congestion at roads:

1. For each car i:
    a. Determine the current route
2. For each car i's current route:
    a. Map the source and destination to their nearest nodes in the road graph
3. For each with source/destination pair:
    a. Determine all simple paths from source to destination
    b. Find maximum (say 3) alternative paths that are maximally dissimilar to the original route and to each other using Dijsktra or A* algorithm.
4. For each car i, define the set of possible routes needed to form the QUBO.
5. Define the matrix Q with binary variables qij as described.
6. Solve the QUBO problem
7. Update cars with the selected routes.

QUBO function

The functional form of the QUBO that the QPU is designed to minimize is:

$$\mathrm{Obj}(x, Q) = x^T \cdot Q \cdot x,$$

A cost function that represents the cost of traffic congestion needs to be constructed to formulate the problem for a quantum annealer. From this formulation, we create a QUBO problem, which is suitable for a quantum annealer due to its equivalence to the Ising model. The cost function for a road segment s is defined as follows:

$$\text{cost}(s) = \left(\sum_{q_{ij} \in B_s} w_{ij} q_{ij} \right)^2$$

Here:

- qij represents the presence of vehicle i on segment j, where qij = 1 if the vehicle traverses segment j and qij = 0 otherwise.
- Bs denotes the set of vehicles traversing segment s.
- ws is the weight assigned to segment s, reflecting the congestion weight.

To ensure each vehicle is assigned exactly one route, we introduce the following constraint:

$$\sum_{j=1}^{3} q_{ij} - 2 \sum_{j=1}^{3} \sum_{k>j} q_{ij} q_{ik} = 1, \quad \forall i$$

This constraint guarantees that each vehicle i selects precisely one of the three available routes. The first term on the left-hand side sums the presence of vehicle i on each route, while the second term enforces the exclusion of scenarios where a vehicle simultaneously takes multiple routes. Hence,our global objective function for the QUBO problem is revised as follows:

$$\mathcal{O} = \sum_{s \in S} \left(\sum_{q_{ij} \in B_s} w_{ij} q_{ij} \right)^2 + \lambda \sum_{i} \left(\sum_{j} q_{ij} - 1 \right)^2$$

Here:

• S represents the set of all street segments.

• λ is a scaling parameter ensuring compliance with the route assignment constraint for each vehicle.

The second term penalizes scenarios where a vehicle takes more than one route, contributing to the overall optimization objective.

Given a large QUBO input, qbsolv partitions the input into important components and then solves the components independently using queries to the QPU. This process iterates (with different components found by Tabu search) until no improvement in the solution is found. The qbsolv algorithm can optimize subproblems using either a classical Tabu solver or via submission to a D-Wave QPU.

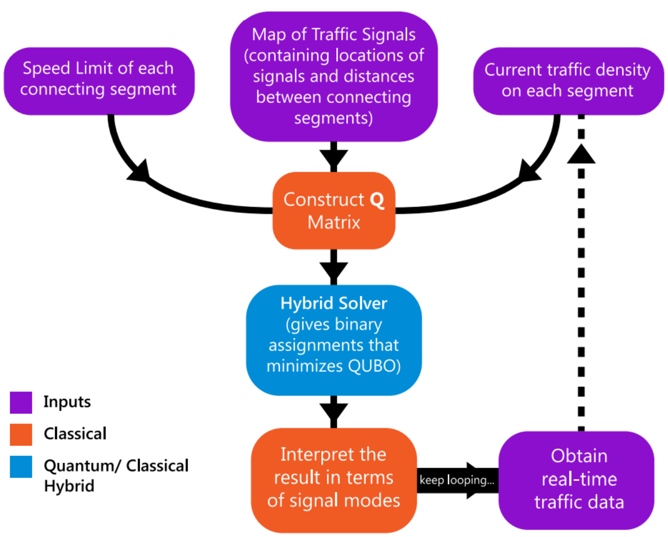
Traffic Signal Optimizartion

Algorithm for Traffic Signal Optimization:

Now, in order to solve this problem, we divide the problem into a chronological workflow:

1. Classical: Process map data, including the lengths of all segments and their respective speed limits.
2. Classical: Process traffic density data and information as to which signals to synchronize in this instance (if any).
3. Classical: Formulate the QUBO matrix.
4. Hybrid classical/quantum: Find a solution that provides smoothest flow of traffic across the route.
5. Classical: According to the resulting signals configuration, the simulation moves traffic on each segment resulting in updated traffic density data. The simulation also figures out which signals to coordinate for the next run.
6. Repeat steps 2–5.

Here, ‘Classical’ refers to calculations carried out on classical machines.



QUBO function

In the traffic signal control, the objective function to be minimized in the QUBO problem can be expressed as (by similar logic as above):

$$O = \sum_j C_{ij} q_{ij} + \varphi \sum_i \left(1 - \sum_j q_{ij}\right)^2$$

where qij are binary variables representing whether the traffic light signal mode j (green, yellow, red phase) at the i-th junction (traffic light group) is on. Provided with a properly chosen coefficient Cij, the first term in the right hand side would be associated with the potential the current set of signal modes can improve the overall traffic condition. In this work, we chose Cij as the total number of halting cars on inbound lanes controlled by signal mode j of the i-th junction. The objective function by design would optimize the system to clear out the maximum number of vehicles halting before the intersection. The second term with a large enough penalty factor φ is the constraint that ensures only one mode would be selected at a time at each junction.

Emergency Vehicle Routing:

Ensure that emergency vehicles follow the least congested, highest-priority routes.

Dynamic Feedback Loop:

- **Continuous Monitoring:** The system continuously monitors traffic conditions and updates routes and signal timings based on real-time data.
- **Real-Time Adaptation:** The quantum-classical approach allows for real-time adaptation, ensuring the system remains effective as traffic patterns change during the event.

Pipeline Detail:

1. Input Stage:

- **Vehicle Data:** Real-time locations, types (normal or emergency), and current routes.
- **Road Network Data:** Graph representation of roads and intersections.
- **Event Data:** Traffic increases due to urban events, leading to higher congestion in certain areas.
- **Traffic Signal Data:** Current signal timings at key intersections.
- **Capacity Data:** Defined road and intersection capacity limits.

2. Preprocessing Stage:

- **Route Generation:** Generate alternative routes for each vehicle.
- **Traffic Load Mapping:** Apply traffic density weights to road segments based on real-time congestion and event data.
- **Capacity Constraints:** Ensure that road segments and intersections do not exceed their capacity.
- **Prioritization of Emergency Vehicles:** Assign high-priority, congestion-free routes to emergency vehicles.

3. QUBO Formulation Stage:

- **Define binary variables** for car-route assignments.
- **Construct the QUBO objective function** with terms for minimizing travel time, balancing load, prioritizing emergency vehicles, and dynamic traffic signal adjustments.

4. Quantum Optimization Stage:

- **Solve the QUBO problem** using quantum annealing or hybrid quantum-classical solvers.
- **Output optimized routes** and **traffic signal timings** that minimize overall travel time and ensure balanced traffic flow.

5. Postprocessing Stage:

- **Vehicle rerouting:** Reassign vehicles to their optimized routes.
- **Traffic signal adjustments:** Implement the optimized signal timings at key intersections.

6. Output Stage:

- **Optimized traffic flow:** Vehicles are rerouted to reduce travel time and balance load across the road network.
- **Prioritized emergency routes:** Emergency vehicles follow priority routes with minimal congestion.
- **Optimized signal timings:** Traffic signals dynamically adapt to ensure smoother traffic flow and reduced wait times.

Conclusion:

This enhanced approach integrates multiple objectives—minimizing travel time, balancing traffic load, prioritizing emergency vehicles, and dynamically adapting to real-time data. By using a hybrid quantum-classical solution, we can efficiently optimize traffic signal timings, reroute vehicles, and ensure congestion-free paths for emergency vehicles, providing a scalable solution for managing traffic during urban events.