Drestanto Muhammad Dyasputro (1450107)
ddyasputro@student.unimelb.edu.au
d.dyasputro@unimelb.edu.au

Research Project – Experiment Write Up

Vehicle Routing Problem (VRP) and Graph Representation

The Vehicle Routing Problem (VRP) is a classic optimization challenge that involves determining the most efficient routes for one or more vehicles to serve a set of customers. For simplicity, we've focused on a scenario with a single depot and a single vehicle. Our input consists of customer and depot coordinates, which define their spatial locations.

To translate this spatial problem into a format suitable for computation, we treat each customer and the depot as a node in a graph. The routes connecting these nodes are represented as edges, with weights assigned to each edge to reflect the distance or travel time between the nodes. Classical algorithms, such as Dijkstra's or A*, can be used to construct this graph and calculate the edge weights.

For n customers and 1 depot, the graph can be represented as an NxN (square) matrix where N equals n+1, where each element in the matrix, row i and column j corresponds to the weight of the edge between nodes i and j. The matrix will be symmetric with zeros in the diagonal, meaning that the transpose will equal to the matrix itself

Quantum Approximate Optimization Algorithm (QAOA) Overview

QAOA is a quantum algorithm (or hybrid quantum-classical algorithm to be exact) designed to find approximate solutions to combinatorial optimization problems. Unlike exact quantum algorithms, QAOA aims to find good (but not necessarily best/optimal) solutions within a reasonable timeframe.

The QAOA process involves:

1. **Cost Function:** Defining a cost function that represents the problem's objective. In our VRP, this function measures the total distance or travel time of a route.
2. **Hamiltonian:** Encoding the cost function into a quantum Hamiltonian.
3. **Quantum Circuit:** Constructing a parameterized quantum circuit that alternates between applying the cost Hamiltonian and a mixing Hamiltonian.
4. **Optimization:** Iteratively adjusting the parameters (gamma and beta) of the quantum circuit to minimize the cost function, using a classical optimizer.
5. **Approximate Solution:** Extracting an approximate solution from the final quantum state.

Drestanto Muhammad Dyasputro (1450107)
ddyasputro@student.unimelb.edu.au
d.dyasputro@unimelb.edu.au

Research Project – Experiment Write Up

Key features of QAOA include:

- **Approximate Solutions:** QAOA provides approximate (not necessarily optimal) solutions.
- **Parameterized Circuits:** The quantum circuit's parameters (gamma and beta) are optimized classically.
- **Depth (p):** The depth of the QAOA circuit determines the number of iterations and affects the solution's quality.

VRP Representation in Quantum Qubits

To apply QAOA to the VRP, we need to represent the problem using quantum qubits. A naive approach is to represent all possible edges between nodes. However, this is not optimal.

- **Graph Representation in Matrix:** As mentioned earlier, we use an (n+1) x (n+1) matrix to represent the graph.
- **Qubit Allocation:** The key challenge is to efficiently encode the order of customer visits. Using n+1 qubits is insufficient. This is because VRP requires representing not just binary choices, but the order of visits. We found that $2^k \geq n!$, where k is the number of qubits, is sufficient. This is because there are n! possible permutations of order in n customers. The easiest way to find k is to use the matrix number of element. If $k = (n+1)^2$, 2^k will grow much faster than n! (We can proof this using mathematical ratio with techniques such as stirling's approximation and logarithm)

Cost Function

The cost function in our QAOA implementation of the VRP is designed to minimize the total travel distance while ensuring the solution represents a valid route. This involves two main components:

1. Distance Cost: This term represents the total travel distance, which we aim to minimize.
2. Constraint Penalties: These terms penalize invalid routes that violate the VRP's constraints, such as visiting a customer more than once or not visiting a customer at all.

Let's consider representing the connections between locations using binary variables x_{ij}, where $x_{ij} = 1$ if we travel from location i to location j, and $x_{ij} = 0$ otherwise. Use d_{ij} is the distance between location i and location j. We can formulate cost function to:

$$C(x) = A*\Sigma(d_{ij} * x_{ij}) + B*\Sigma((\Sigma(x_{ij}) - 1)^2) + C*\Sigma((\Sigma(x_{ji}) - 1)^2)$$

Drestanto Muhammad Dyasputro (1450107)
ddyasputro@student.unimelb.edu.au
d.dyasputro@unimelb.edu.au

Research Project – Experiment Write Up

Where A, B, and C are penalty parameters that control the trade-off between minimizing distance and satisfying constraints. In most cases, it makes sense to set B = C.

Quantum Circuit and Hamiltonian

The cost function C(x) is then translated into a quantum Hamiltonian H_C by replacing the binary variables x_{ij} with Pauli-Z operators. This Hamiltonian is used in the QAOA circuit to guide the quantum state towards solutions that minimize the cost.

The QAOA quantum circuit alternates between applying the cost Hamiltonian and a mixing Hamiltonian. The mixing Hamiltonian typically involves single-qubit rotations. The circuit's depth (p) determines the number of iterations and affects the solution's quality.

The gamma parameter (for cost function Hamiltonian $U_C(\gamma) = e^{(-i\gamma H_C)}$) as well as beta (for mixing Hamiltonian $U_M(\beta) = e^{(-i\beta H_M)}$) in the QAOA circuit control the evolution under the cost and mixing Hamiltonians, respectively. We can start with random initial values for these parameters and then optimize them using a classical optimizer.

Memory Problem

We encountered a significant memory problem when attempting to simulate the QAOA circuit for even a small VRP instance (5 customers, 1 depot). The "default.qubit" PennyLane simulator attempts to store the entire quantum state vector, which requires 2^k complex amplitudes. For our 6-node problem, this required more than 64 GB of memory, which was insufficient.

In our case, with 5 customers and 1 depot (6 nodes), we initially aimed to use a qubit encoding that resulted in $k = (n+1)^2$ qubits. With n = 5, this means we would have $k = (5+1)^2 = 36$ qubits. Therefore, the memory requirement would be $2^{36} * 16$ bytes. Calculating this:

$$2^{36} * 16 \text{ bytes} = 2^{40} \text{ bytes} = 1 \text{ Terabyte}$$

Therefore, using $(n+1)^2$ qubits would have resulted in a memory requirement of exactly 1 Terabyte. This is of course impractical. So we need to find another k where $2^k > n!$.

Comparison of Operation Number (Classical vs. Quantum)

- **Classical:** Classical algorithms for the VRP, such as exhaustive search, require exploring all possible permutations, resulting in approximately (n+1)! operations for n customers.
- **Quantum:** QAOA aims to find approximate solutions with potentially fewer operations, but the exact reduction depends on the problem instance and QAOA parameters.The

higher the depth p, the higher the complexity. The circuit qubit requirements and circuit complexity will also increase. In a controlled number of iteration s and number of depth p, the total quantum gate operation will be s*p*(n+1)^2 which is less than (n+1)!.

Other Notes: Using v Vehicles and d Depots

Introducing v dan d into account, gives (approximately):

$$T(n, d, v) \text{ (Classical)} \approx (n + d)!*v^n$$

$$T(n, d, v, p, s) \text{ (Quantum)} \approx s*p*v*(n+d)^2$$

- **Classical:** With v vehicles and d depots, the classical problem's complexity increases significantly, as we need to consider all possible assignments of customers to vehicles and depots.
- **QAOA:** QAOA can potentially handle this increased complexity, but the qubit requirements and circuit complexity will also increase. We would need to encode the vehicle assignments and depot selections into the quantum state.