Seminar: Advanced Topics in Quantum Computing
On efficient encodings for QAOA solutions to vehicle routing problems

Eben Jowie Haezer

October 30, 2023

Abstract

This report details recent advances in the optimisation of computing resources for quantum approaches to solving the vehicle routing problem (VRP) and its variants. This set of problems is of significant importance with regard to logistical applications in industry. In accordance with the input constraints of the quantum hardware, the problem is formulated as a quadratic unconstrained binary optimisation (QUBO). A simple approach known as the full encoding results in each solution represented by a unique basis state, thereby requiring one qubit per classical variable. Due to this inefficiency in resource allocation when considering the limitations of noisy intermediate scale quantum (NISQ) devices at present, a more optimised minimal encoding is suggested that offers a logarithmic reduction in necessary computing power. In spite of certain drawbacks incurred by employing this minimal encoding, experiments have shown that the solution quality is not heavily impacted.

I. Introduction

The Vehicle Routing Problem (VRP) is concerned with finding optimal routes for vehicles to deliver goods to a set of customers with some geographical distance between them. It is obvious that this type of problem has far reaching practical applications in various contexts, in fact the need to find a reasonable solution is almost ubiquitous when dealing with logistical planning, for example to determine efficient road, rail, shipping, and air routes for commercial or public interest.

VRP, being itself a more general version of the Travelling Salesman Problem (TSP), is similarly an NP hard [1] combinatorial optimisation problem with a solution space scaling factorially in the number of customers, and thus finding the definitive optimal solution already becomes intractable at two digit customer counts. The most optimal classical algorithms known to date and used in practice employ heuristics and greedy methods to construct routes within single digit percentage tolerances.

More recently, the noisy intermediate scale quantum (NISQ) era [2] has paved the way for the development of variational quantum algorithms alongside the potent quantum annealing [3] method as a contender for computing reliable and near optimal solutions to combinatorial optimisation problems, exploiting the unique inherent property of quantum algorithms to operate on the entire solution space at once, albeit incrementally and probabilistically.

Initial tests have shown some promise in applying these quantum algorithms to the VRP against classical solvers, achieving comparable accuracy on small problem instances [4] [5]. As quantum hardware continues to improve, it makes sense to also look at utilising it as efficiently as possible. Solving VRP naively using quantum methods maps one classical variable to one whole qubit [5], quickly exhausting the already limited computing resources when introducing additional constraints such as vehicle capacity or time windows on larger problem scales more commonly encountered in practical scenarios.

This report discusses recent research into an idea to mitigate this uneconomical use of computing power through a more clever and refined encoding of the input problem, in order to achieve a logarithmic correlation between problem size and resource consumption. Section II outlines a formal description of the VRP and some of its variants as a graph problem. In section III, quantum algorithms for NISQ era processors that see use in combinatorial optimisation are briefly highlighted. Section IV explains the QUBO and its application to solving the VRP on quantum hardware, along with a brief mention of the closely related Ising model and

in particular its Hamiltonian function. From this, a formulation for the VRP can be derived. Section V details encoding approaches onto the quantum system and lists benefits and drawbacks for each, with a focus on managing computational resources. Published experimental observations describing the impact of the encodings are depicted and discussed in Section VI, and the report concludes in Section VII.

II. Vehicle Routing Problem

The vehicle routing problem (VRP) is a generalisation of the perhaps more well known NP hard Travelling Salesman Problem (TSP). VRP seeks the optimal route or routes for a number of vehicles to traverse in order to deliver certain goods to customers in various locations.

The problem is modelled intuitively by a graph $G = (V, E)$ where each node or vertex $v_i \in V$ represents the location of a customer and each edge $(i, j) \in E$ connecting two vertices corresponds to a path traversible by a delivery vehicle. $\|V\| = N$ is then the number of customers considered. Traversing an edge (i, j) typically incurs a cost represented by $c_{ij} \in \mathbb{R}_{\geq 0}$, of which the total value summed up across the entire journey is to be optimised, ie. made as small as possible. This cost may be set based on travel time, distance, or other concerns with economic consequences.

Vehicles may only travel across edges. For simplicity, one can assume G is a complete graph. Those edges directly connecting two nodes $(i, j) \in E$ that are absent in reality can be labelled with an edge cost $c_{ij} = c_{in_1} + ... + c_{n_m j}$ corresponding to the most efficient path available between them. This preprocessing step does not necessarily complicate matters, and classical algorithms exist for all pairs shortest path with known polynomial runtime performance, such as the Dijkstra algorithm. Furthermore, each graph contains a designated node $v_0 \in V$ known as the depot. Valid routes must always begin and end at the depot. A valid route is then a tuple $r = (v_n, v_{n+1}, ..., v_m)$ s.t. $v_n = v_m = v_0 \ \wedge \ \forall n.\ (v_n, v_{n+1}) \in E$.

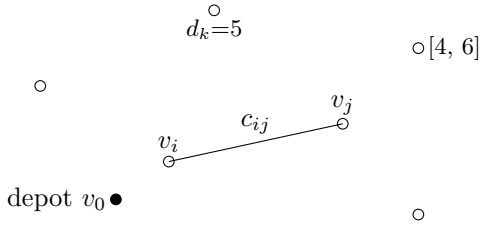

Figure 1: An illustration of the VRP as a graph problem. The white nodes represent customers, with an optional demand value d_k or a time window interval. One aims to find the most efficient route (represented by the cumulative edge cost $c_i j$) that delivers to all customers and respects the problem constraints.

For a problem to be classified as VRP, it should fulfil at least the above minimal constraints. However, additional constraints may be imposed as needed to better reflect a practical use case at the expense of slightly complicating the model. For example, the capacitated VRP or CVRP stipulates a fixed upper bound on the carrying capacity of each vehicle leaving the depot, where in most cases this value is consistent across all vehicles. Customers v_i are then assigned a score $d_i \in \mathbb{R}_{\geq 0}$ reflecting their demand quantity. This introduces the complication of optimising for capacity and demand as well as edge cost, and the ideal solution for a given graph will most likely differ from the simple VRP.

On the other hand, the VRP with time windows (VRPTW) introduces a secondary time parameter. Customers v_i are assigned a certain time window $[t_i^0, t_i^f] \subseteq \mathbb{R}_{\geq 0}$ in which they expect a delivery. For the depot v_0 this interval is $[0, \infty)$ for simplicity. Hence the edge costs c_{ij} represent travel time between two nodes in this formulation, and the objective shifts to finding the optimal route that serves all customers whilst respecting these time windows, or failing this attempting to maximise the number of customers or total goods supplied.

III. Quantum Optimisation

Traditionally, quantum solvers for combinatorial optimisation problems have been implemented using the quantum annealing approach. This method relies on the theory of adiabatic quantum computation to evolve a prepared system state of the Hamiltonian H_0 gradually towards the problem Hamiltonian H_p. With sufficiently slow evolution and noncommuting Hamiltonians $[H_0, H_p] \neq 0$ (see Appendix B), the system is guaranteed to arrive in the desired ground state of the problem by the adiabatic theorem.

More precisely, the system evolves according to

$$H(t) = \left(1 - \frac{t}{T}\right) \cdot H_0 + \frac{t}{T} \cdot H_p \qquad (1)$$

with $0 \leq t \leq T$, where the total algorithm runtime T is bound by $T \in \mathcal{O}(\frac{1}{g^2})$. Here g denotes the spectral gap of H, the minimal energy difference between the ground state and first excited state

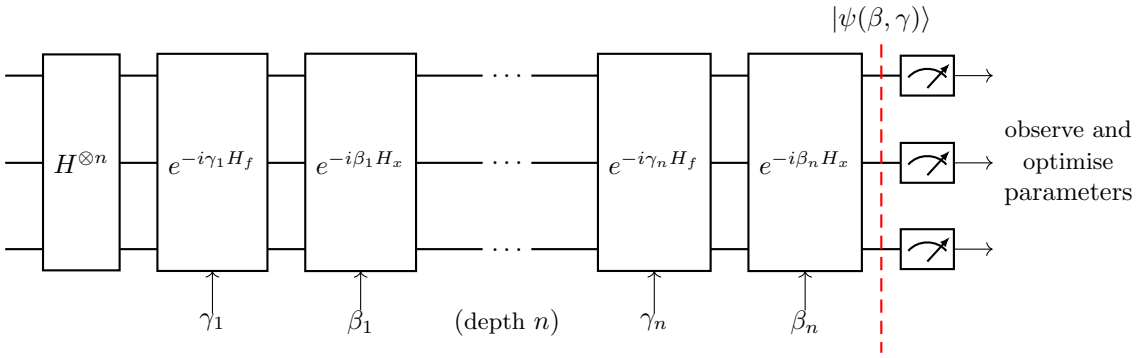

Figure 2: Sketch of the QAOA setup. The state is brought into uniform superposition after which repeated alternating application of rotation operators realises approximately the Hamiltonian $H = H_f + H_x$ according to (4). The result is measured in the $Z^{\otimes n}$ basis and the parameters β, γ are optimised classically for the next run.

throughout the evolution. Consequently, Hamiltonians with narrower spectral gaps have to be evolved more slowly for the sake of precision.

On the other hand, the more common circuit model processor operates based on a series of quantum gates connected in a circuit in order to manipulate the quantum state to a desired outcome. This principle of operation is founded upon the solution to the time dependent Schrödinger equation for time invariant Hamiltonians:

$$i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle = H|\psi(t)\rangle \iff U(t) = e^{\frac{-i}{\hbar}Ht} \tag{2}$$

where U denotes the unitary time evolution operator (see Appendix A). This unitary can be decomposed into several building blocks termed quantum gates, which are implemented in the hardware and operate on a small subset of qubits. The quantum circuit thus realises the Hamiltonian.

For complex Hamiltonians of the form $H = \sum_k H_k$, the following limit equation provides a key insight for $A, B \in \mathbb{C}^{m\times m}, m \in \mathbb{N}$:

$$e^{A+B} = \lim_{n\to\infty}\left(e^{\frac{A}{n}} \cdot e^{\frac{B}{n}}\right)^n \tag{3}$$

As a result, the time evolution can be approximated as:

$$U(t,n) = \prod_{i\leq n}\prod_k e^{\frac{-iH_k t}{n}} \tag{4}$$

ie. a series of smaller gate unitaries that may be repeated ad infinitum.

The advances in circuit model quantum processors that have brought about the NISQ era have offered an alternative approach to tackle optimisation problems, namely that of supplementing a quantum algorithm with classically tuned parameters to iteratively arrive at a good approximation of the ground state, known as variational algorithms. NISQ era processors are characterised by a primary limitation of restricted circuit depth due to their sensitivity to noise as well as the inability to correct the faults introduced. These practical limitations prevent the substitution high values of n in (4). However, by offloading some of the computation to a classical algorithm and proceeding with iterative refinement, this limitation can be worked around to a degree that allows one to obtain reasonable estimations of the ideal solutions to optimisation problems.

Those variational approaches specifically intended for combinatorial optimisation are given the name quantum approximate optimisation algorithm (QAOA), first described by Farhi et al. in [6]. Following the recipe in (4), a circuit as in Fig. 2) is constructed out of alternating operators H_f and H_x, where the former represents the objective function and the latter is termed a mixer Hamiltonian.

$$|\psi(\beta,\gamma)\rangle = \prod_i e^{-i\beta_i H_x} e^{-i\gamma_i H_f}\ |+\rangle^{\otimes n} \tag{5}$$

where $|+\rangle^{\otimes n}$ is the uniform superposition obtained through the Hadamard transform of the initial ground state, so as to begin optimising from a neutral position.

The series of unitary evolutions are commonly implemented in the gate model as qubit rotation operator arrays. The parameters β_i and γ_i represent angle values for these rotations H_x and H_f respectively, which can be interpreted as the discretised duration of time the system spends under a particular unitary evolution.

Since H_f is diagonal w.r.t. $Z^{\otimes n}$ as in (10), we typically choose the noncommuting mixer $H_x = \sum_k \sigma_k^x$, also known as the transverse field mixer (cf. (9)). However, this choice is by no means inconsequential. Further studies have yielded more

optimal choices for H_x and the initial system state tailored to the problem at hand [7].

In any case, the primary purpose of the mixer is to allow the system to escape undesirable local eigenstates should they be assumed in the course of running the algorithm. For this, the noncommutative relation between the two operators ensures that their individual eigenstates do not coincide (see Appendix B). From a macro standpoint, the mixer acts to dissuade convergence towards local minima by effectively reshuffling the state to an extent.

After each pass of the circuit, the system state should attain significant overlap to the desirable eigenstate. The optimised result is read out with high probability through a measurement and a classical optimiser attempts to refine the angle parameters β_i, γ_i. This procedure is then repeated until a sufficiently good solution is found. By the adiabatic theorem and (4), arbitrarily precise solutions are theoretically possible. In practice studies have highlighted certain limitations, for example correlating solution quality inversely with the ratio of constraints to variables in a problem [8].

IV. Problem Formulation

The quadratic unconstrained binary optimisation (QUBO) is concerned with finding a binary vector $|x^*\rangle \in \{0,1\}^n$, $n \in \mathbb{N}$ that fulfils the following optimal condition:

$$|x^*\rangle = \underset{|x\rangle \in \{0,1\}^n}{\operatorname{argmin}} \langle x|Q|x\rangle \tag{6}$$

where the linear operator $Q \in \mathbb{R}^{n\times n}$ is a symmetric matrix. This can be interpreted as an objective function:

$$f_Q(x) = \langle x|Q|x\rangle = \sum_{i=1}^{n}\sum_{j=i}^{n} Q_{ij}x_i x_j \tag{7}$$

In general, QUBO is also NP hard due to the exponential scaling of the solution space in $\|\{0,1\}^n\| \in \mathcal{O}(2^n)$ with respect to the number of dimensions n. Many combinatorial optimisation problems have conversions into QUBO [9] [10], not least the VRP and its variants. These conversions are useful to establish a uniform problem description for solvers to work with, however in the context of quantum solvers, they are further motivated by the equivalence between QUBO and the Hamiltonian of the Ising model for ferromagnetism in particle physics.

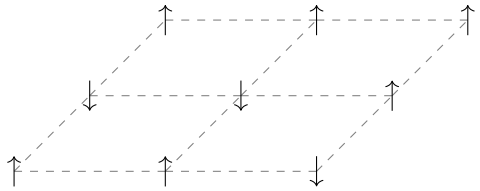

Figure 3: Lattice structure depicting the Ising model in two dimensions. At each lattice site $k \in \Lambda$ an arrow represents the atomic spin $\sigma_k \in \{-1, 1\}$ that influences the local dipole moment. Dashed lines are nearest neighbour interactions J_{ij}.

The Ising model describes a lattice structure Λ in which each lattice site houses a particle. The spin of each particle is represented by a discrete variable $\sigma_i \in \{-1, 1\}$, $i \in \Lambda$. This spin value governs the local magnetic moment of the particle (see Appendix C). Neighbouring lattice sites $\langle i\ j\rangle$. $i, j \in \Lambda$ influence each other, termed nearest neighbour interactions, whose interaction strength is represented by $J_{ij} \in \mathbb{R}$. Furthermore, one may consider the influence of an external field h_i at site $i \in \Lambda$, such that the spin wants to align with this field.

Thus the Hamiltonian reads:

$$H = -\sum_{\langle i\ j\rangle} J_{ij}\sigma_i\sigma_j - \mu\sum_i h_i\sigma_i \tag{8}$$

where μ is the magnetic moment.

Replacing σ with the Pauli operators yields the quantum mechanical description:

$$H = -\sum_{\langle i\ j\rangle} J_{ij}\sigma_i^z\sigma_j^z - \mu\sum_i h_i^z\sigma_i^z - \mu\sum_i h_i^x\sigma_i^x \tag{9}$$

where the second term with σ^z describes the external longitudinal field, and the final term with σ^x describes the transverse field per lattice site.

Notably, the Ising model is typically simplified to exclude the transverse field, resulting in a classical Hamiltonian where the constituent terms commute ie. a diagonal operator in the Z basis.

$$H = \sum_{z\in\{0,1\}^n} E_z\ |z\rangle\langle z| \tag{10}$$

This means the ground state E_0 is described by a corresponding eigenvector $|\phi\rangle$ with $\phi \in \{0,1\}^n$, allowing for simple measurement to obtain an eigenvalue mappable to a specific bit string solution.

Through the reversible transformation $\sigma \mapsto 2x - 1$, where $x \in \{0, 1\}$ s.t. $-1 \mapsto 0$ and $1 \mapsto 1$, one obtains the equivalent QUBO formulation for free, and hence optimising for the ground state of the Ising Hamiltonian is equivalent to optimising the QUBO objective function.

With this knowledge, one can formulate the VRP as follows. We define the set of constraints that a valid solution must adhere to:

once Each customer v_i in a route r_j is visited exactly once.

step For each time step, each vehicle is either at a customer node v_i or at the depot v_0.

Additional constraints may be specified depending on the problem variant. For CVRP the following is useful:

cap All routes $\{r_i\}$ respect the maximum capacity $m \in \mathbb{R}$ of a vehicle.

Note that for simplicity, we do not distinguish between vehicles and routes, under the premise that each route will be navigated by exactly one vehicle. Since in CVRP the routes are optimised for the vehicle capacity m, assuming homogeneous vehicles this simplification is unlikely to affect the final result. Furthermore, some CVRP variants consider multiple depot nodes. This variant will also not be further investigated here.

Let $x^r_{t,i} \in \{0,\ 1\}$ denote the assignment of the customer node v_i to be visited along route $r \in \mathbb{N}_0$ at time step $t \in \mathbb{N}_0$. The first two constraints can then be described mathematically as follows:

$$H_{\text{once}} = \sum_i (1 - \sum_{r,\,t} x^r_{t,i})^2 \tag{11}$$

$$H_{\text{step}} = \sum_{r,\,t} (1 - \sum_i x^r_{t,i})^2 \tag{12}$$

The idea is to coerce each squared term to evaluate to zero, implying each inner summation should evaluate to one. To illustrate, consider for example H_{step} which encodes the constraint **step**. We sum over all node indices i, such that for a given step t along the route r, precisely one customer node or the depot is assigned to the schedule. In general, once a constraint is violated it follows that the calculation yields a positive result, after which penalty values may be applied accordingly.

Encoding the constraint **cap** is a little more involved:

$$H_{\text{cap}} = \sum_r \left(m - (\Delta + \sum_{t,\,i} d_i x^r_{t,i})\right)^2 \tag{13}$$

where d_i represents the demand score of node v_i and

$$\Delta = \sum_{b=0}^{\lceil \log_2 1+m \rceil - 1} 2^b\ a^r_b \tag{14}$$

While the principle remains identical to the first two constraints, The additional term Δ in H_{cap} is a result of **cap** being somewhat more relaxed, in that any value $\sum_{t,\,i} d_i x^r_{t,i} \leq m$ for the loading of the vehicle is acceptable, instead of a hard requirement. This is conveyed through the auxiliary variables $a^r_b \in \{0,\ 1\}$, which may be freely set to perfectly zero out unused vehicle capacity for the route r. Intuitively, Δ represents the tolerable deviation of the amount loaded onto the vehicle on route r compared to maximum capacity. Naturally not fully loading the vehicle and forcing an early return to depot may be suboptimal in many cases, however the edge cost metric represented by the objective function H_c is sufficient to deter this.

$$H_c = \sum_{i,\,j,\,t,\,r} c_{ij}\ x^r_{i,t}\ x^r_{j,t+1} \tag{15}$$

Coupled with the objective function given by H_c, the complete CVRP Hamiltonian is the linear combination of these constraints:

$$H_p = H_c + p_1 H_{\text{once}} + p_2 H_{\text{step}} + p_3 H_{\text{cap}} \tag{16}$$

Here $p_k \in \mathbb{R}_{\geq 0}$ with $k \in \{1, 2, 3\}$ corresponds to a large penalty value that serves to mark a potential solution as undesirable, in the case that one or more of the corresponding constraints are violated. The exact values of p_k are dependent on the problem instance and must be heuristically determined and optimised.

Crucially, this description of the CVRP is only one of several possible formulations. Extensive research has been conducted into determining the most optimal description that benefits solution quality as well as speed of convergence. In particular, the formulation in (16) turns out not to be suitable with regard to current limitations of NISQ era processors, due to the large amount of free binary variables and comparative lack of computing resources available to account for them all.

As it stands, the number of binary variables needed is in the order of $\mathcal{O}(\|V\|^3)$, considering the number of customer nodes, time steps required, and routes or vehicles. One solution employed by [4] and [11] involves splitting the problem into two distinct phases, a clustering phase that groups customer nodes together similar to the knapsack problem, and a routing phase to determine the ideal route in each of these clusters as in the TSP.

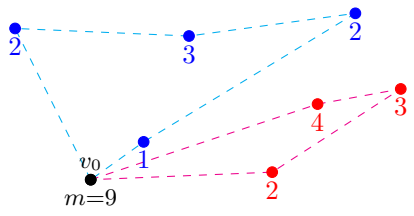

Figure 4: The clustering and routing phase. The clustering phase divides nodes into collections (shown in blue and red) where the total demand shown under each node does not exceed the vehicle capacity (in this case $m = 9$), similar to the multiple knapsack problem. The routing phase involves optimising each route individually for minimal edge cost as a TSP.

The clustering phase aims to discern routes by grouping nodes together in such a way that **once** and **cap** hold true for each cluster. Furthermore, intuition suggests neighbouring nodes should not be split apart but rather be assigned to the same cluster to minimise edge costs.

cluster The cumulative edge cost $c_i j$ between all pairs of nodes (i, j) within a cluster should be minimal.

This results in the following problem Hamiltonian:

$$H_{\text{kn}} = q_1 H_{\text{once}} + q_2 H_{\text{cap}} + q_3 H_{\text{cluster}} \tag{17}$$

with penalty values q_i, where H_{cluster} is defined as:

$$H_{\text{cluster}} = \sum_r \sum_{i,\, j} c_{ij}\; x_i^r\; x_j^r \tag{18}$$

and $x_i^r = \sum_t x_{i,t}^k$.

It is worth noting that the penalty value q_3 should be set more modestly in comparison to the others, given that **cluster** is not a hard constraint but rather a metric measuring the optimality of clusters. It should always be more appealing to relax **cluster** over violating **once** or **cap**.

Unfortunately, quantum solvers have been shown to perform relatively poorly when applied to the multiple knapsack problem (refs), hence at present the clustering phase should be deferred to a classical algorithm as suggested in [4].

The routing phase takes the partitioned clusters and determines the optimal route for each cluster locally. The problem is thus reduced to the TSP, which conveniently has a QUBO formulation given in [9] as follows

$$H_{\text{tsp}} = q_1 H_{\text{once}} + q_2 H_{\text{step}} + q_3 H_c \tag{19}$$

with the exception that, as we have broken up the problem and there is locally only one route to optimise, in all terms we do not sum over r.

Overall, the TSP requires $\mathcal{O}(\|V\|^2)$ binary variables to account for all possible nodes and all possible time steps. Here [9] suggests a resource optimisation saving to $(\|V\| - 1)^2$ variables by fixing the start node to be the depot v_0.

V. Encoding Approaches

We assume the use of a conventional NISQ processor to tackle a VRP involving n_c classical binary variables. A simple mapping from the VRP to QUBO uses what is known as the complete encoding, where each classical binary variable is represented by a single qubit. The quantum state takes the form:

$$|\psi(\theta)\rangle = U(\theta)|\psi_0\rangle \tag{20}$$

$$= \sum_{x \in \{0,\, 1\}^{n_c}} \alpha_x |x\rangle \tag{21}$$

In this case, sampling the quantum state yields any given bitstring solution $x \in \{0,\, 1\}^{n_c}$ with probability $\|\alpha_x\|^2$, where α is parametrised by a set of angles θ. In variational algorithms, θ are heuristically determined values that serve to tune arrays of rotation unitaries, which are adjusted after each pass to attain the optimised state $|\psi^*\rangle = U(\theta^*)|\psi_0\rangle$ (cf. (5)).

The QUBO objective function in the complete encoding is described by the expectation of the equivalent Ising Hamiltonian:

$$C_{\text{cpl}}(\theta) = \langle H \rangle = \langle \psi(\theta)|H|\psi(\theta)\rangle \tag{22}$$

The advantage of the complete encoding is its ability to capture all possible correlations between classical variables. This however sacrifices scalability, requiring computational resources that scale proportionally to the problem size as each binary classical variable is mapped to a single qubit, hence $n_q = n_c$. As a result, feasible applications of the complete encoding are limited to toy examples. Even as quantum hardware becomes more powerful, it is prudent to examine encoding schemes that can more efficiently take advantage of the available technology.

We consider the minimal encoding as presented in [12]. This scheme allows for n_c classical variables to be mapped to $n_q = 1 + \log_2 n_c$ qubits, using a $n_r = \log_2 n_c$ wide register and $n_a = 1$ auxiliary qubit.

Let $\{|\phi_i\rangle\}$ and $\{|0\rangle, |1\rangle\}$ denote a basis for the register and auxiliary qubits respectively. The system state $|\psi\rangle$ is then defined as:

$$|\psi(\theta)\rangle = \sum_{i=1}^{n_c} \beta_i (a_i|0\rangle + b_i|1\rangle)\; |\phi_i\rangle \tag{23}$$

where the coefficients β_i, a_i, b_i are parametrised by θ, as in the complete encoding.

This encodes in the register the probability β_i of measuring each register state $|\phi_i\rangle$ that corresponds to an individual classical variable x_i, and

in the auxiliary the probability of each of these binary variables taking on the value 0 or 1 according to the Born rule: $\Pr(x_i = 0) = \|a_i\|^2$ and $\Pr(x_i = 1) = \|b_i\|^2$. Taken together, the probability of measuring a certain bitstring solution x for a given θ is expressed by:

$$\Pr x = \prod_{i=1}^{n_c} \Pr x_i = \prod_{i=1}^{n_c} \|b_i\|^2 \tag{24}$$

As an example, consider $n_c = 4$, $n_r = 2$. The normalised system state representing a particular bitstring $x^* = (1,1,1,0)^\dagger$ would be $|\psi^*\rangle = \frac{1}{2}(|1\rangle|00\rangle + |1\rangle|01\rangle + |1\rangle|10\rangle + |0\rangle|11\rangle)$.

The variational algorithm is run for several trials to obtain an estimate of the coefficients a_i and b_i, from which the probability distribution for bitstring solutions as given in (24) may be constructed.

As stated previously, with the minimal encoding a logarithmic scaling of qubits to classical variables is achieved. However, this comes at the expense of a more limited expressiveness in the quantum state. As seen in (24), the probability distribution is sufficient to encode only statistically independent classical variables, ie. those s.t. $\forall i\ j.\ \Pr(x_i \wedge x_j) = \Pr x_i \Pr x_j$, which tend not to be the case for highly complex and intertwined problems as in the VRP. To illustrate, consider two nodes that both have a very high edge cost connecting them to other nodes and the depot, but a negligible edge cost between themselves. It is much more likely for these nodes to be visited sequentially along a route, and hence the probability of visiting any one of these nodes is greatly affected by whether its neighbour was visited at the previous time step.

Reference [12] claims this limitation is enough to limit quantum advantage, to the point where classical simulations may actually be more resource efficient. The independent probabilities governing the values of classical variables may be simulated by as many classical continuous variables. Furthermore, the number of trials required to obtain a solution of sufficient quality is necessarily increased. Depending on the exact problem configuration, particularly small coefficient values in the quantum state may result in certain bitstring outcomes being sampled too infrequently if at all while evaluating the objective function, leading to inaccuracies. A possible workaround for unsampled states is to assume $\Pr x_i = 0.5$ for an unsampled classical binary variable x_i as performed in [5], though this is a very rough estimate at best.

Substituting (24) into (7) and incorporating the probabilistic component, we obtain the following objective function for the minimal encoding:

$$C_{\min}(\theta) = \sum_{i \neq j}^{n_c} Q_{ij} \|b_i\|^2 \|b_j\|^2 + \sum_{i=1}^{n_c} Q_{ii} \|b_i\|^2 \tag{25}$$

$$= \sum_{i \neq j}^{n_c} Q_{ij} \frac{\langle P_i^1 \rangle}{\langle P_i \rangle} \frac{\langle P_j^1 \rangle}{\langle P_j \rangle} + \sum_{i=1}^{n_c} Q_{ii} \frac{\langle P_i^1 \rangle}{P_i} \tag{26}$$

expressed as projectors $P_i = |\phi_i\rangle\langle\phi_i|$ and $P_i^1 = |1\rangle\langle 1| \otimes P_i$.

Additionally, the minimal encoding scheme proposed here was originally conceived for solving graph maximum cut problems [12]. To this end, the scheme can be extended to capture two body correlations, as opposed to simply statistically independent variables as mentioned in this report. This proves useful in the context of maximum cut, since one is interested in the inclusion or exclusion of edges that connect two nodes rather than the nodes themselves. The ability to encode correlations in such a scheme also allows quantum advantage to be retained.

The two body correlations were encoded as follows:

$$\begin{aligned} |\psi(\theta)\rangle = \sum_{i,\ j} \beta_{ij} \big(& \alpha_{ij}^0 |00\rangle + \alpha_{ij}^1 |01\rangle \\ & + \alpha_{ij}^2 |10\rangle + \alpha_{ij}^3 |11\rangle \big) \, |\phi_{ij}\rangle \end{aligned} \tag{27}$$

While the overall scheme is similar, in this case the auxiliary is a $n_a = 2$ wide register which represents the values of pairs of binary variables specified by the main register.

Reference [12] further generalises the construction to encompass many body correlations by way of subdividing binary variables into disjoint sets and expanding the encoding accordingly, however with the drawbacks of increased resource consumption and an increasingly difficult to compute cost metric. The total number of qubits required is then given by

$$n_q = n_a + \log_2 \frac{n_c}{n_a} \tag{28}$$

with the state then having the following representation:

$$|\psi(\theta)\rangle = \sum_i \beta_i \Big(\sum_{z \in \{0,1\}^{n_a}} \alpha_i^z |z\rangle \Big) \otimes |\phi_i\rangle \tag{29}$$

This results in n_a groups of variables. The limiting case $n_a = n_c$ gives the complete encoding. It remains to be seen whether VRP can better make use of this, though this is unlikely given the interconnected nature of the problem.

What follows in the next section are experimental results from various research papers analysing VRP solutions obtained through quantum algorithms on NISQ era devices. In order to examine the feasibility of the various encoding schemes for a

variety of problem sizes, we normalise the objective function in the following manner:

$$\overline{C} = \frac{C(\theta) - C^*}{\Delta C} \tag{30}$$

C^* is the best known solution, typically as found by classical algorithms. ΔC is the difference between C^* and the worst case ie. maximal C.

VI. Experimental Results

Reference [5] claims to have seen ideal convergence behaviour for VRPTW with the minimal encoding for small scale problems, on occasion surpassing that which was achieved when employing the traditional complete encoding.

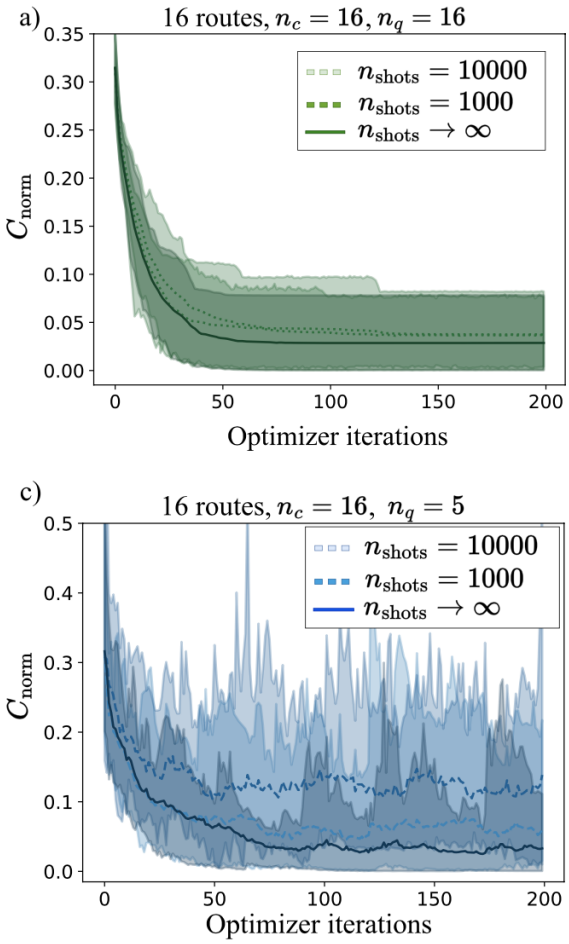

Figure 5: Figure taken from [5]. Performance comparison between the complete encoding (a, top) and the minimal encoding (c, bottom) with respect to the solution improvement after several optimisation runs. n_{shots} represents the curve obtained by sampling the state this many times. The limiting case is also shown. One observes markedly increased variance and slower convergence for the minimal encoding.

Figure 5 depicts how the optimiser iterations influenced the normalised cost compared to the complete encoding for a VRP with $n_c = 16$. One observes the expected reduction in solution quality using the minimal encoding, showing more variance even with increased optimisation runs and requiring substantially more sampling trials to get costs down.

Meanwhile, Figure 6 shows the cumulative distribution of solutions obtained as a function of the normalised cost (labelled C_{norm}) on several devices, plotted in colour. The black curve represents the distribution of all possible solutions, or alternatively $4 \cdot 10^8$ randomly generated solutions, where the former becomes too expensive to compute. A more extreme gradient nearer the vertical axis, where the normalised cost is kept at a minimum reflects well on the overall procedure, as this means a higher percentage of generated solutions are close to ideal.

Tests have also been conducted for larger problem instances that show convergent behaviour, although sufficient iterations were not performed in order for the convergence to manifest completely. This can be seen in the figure, which for $n_c = 3964$ classical variables was still a distance away from returning optimal solutions, limited by the relatively modest count of 10000 sampling trials compared to $n_q = 13, 2^{n_q} = 8192$ different eigenstates. The authors of [5] assumed $p = 0.5$ for unsampled variables and pointed to this as a reason for the suboptimal convergence with increasing optimisation runs.

In comparison, [11] had notably worse results for the two phase CVRP even for small problem sizes, using the complete encoding and QAOA to optimise the routing phase as a TSP. However, they achieved significantly better performance using a related variational algorithm termed variational quantum eigensolver (VQE).

VQE normally sees use in similar optimisation problems in fields such as quantum chemistry, again with the aim of finding the ground state of some problem Hamiltonian. Unlike QAOA, VQE refines the initial state classically, and formulates the problem Hamiltonian as a Pauli string $H = \bigotimes_i w_i \cdot \{\mathbb{1}_i, \sigma_i^x, \sigma_i^y, \sigma_i^z\}$, and measures the result in this manner as well.

In any case, the authors of [11] pointed to the problem formulation as a factor that potentially inhibited the QAOA, and suggested an alternate encoding might yield improved performance. Their version of the TSP additionally considered infeasible paths, ie. the problem was solved on an incomplete graph, and solutions connecting two nodes for which no direct edge between them existed were flagged invalid.

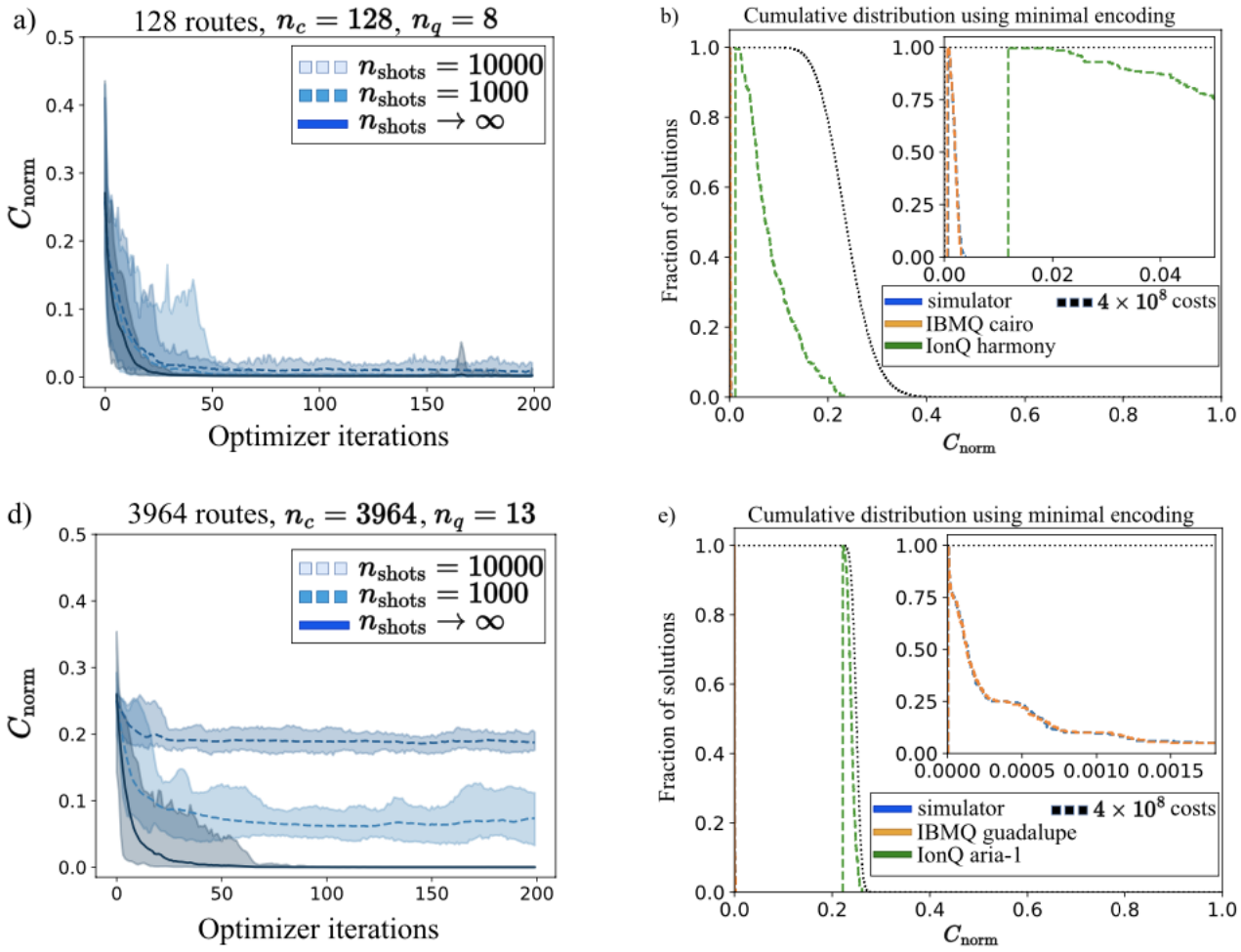

Figure 6: Figure taken from [5]. The left column (a, d) shows optimisation runs with the minimal encoding for increasing problem sizes. Insufficient sampling led to convergence not being attained for $n_c = 3964$ classical variables. The right column (b, e) reflects the cumulative distribution of solutions obtained with the minimal encoding on various NISQ devices, compared with $4 \cdot 10^8$ randomly generated solutions charted in black.

Due to this, a large percentage of proposed solutions were of little value and subsequently discarded. The researchers observed improvements in the feasibility of returned solutions only as the approximated ground state was within 97% of the best known solution.

Based on these preliminary findings, it seems that the success that quantum algorithms enjoy upon tackling VRP is heavily dependent on an advantageous problem formulation. In light of this, continued research towards optimising this for various algorithms with an eye on resource efficiency may yet prove fruitful as NISQ era processors become only more potent with time.

VII. Conclusion

The report summarises some recent research into using quantum algorithms feasible on current NISQ era processors to derive approximate solutions to the NP hard vehicle routing problem and its variants. Particular attention has been drawn to the ways in which VRP can be cleverly formulated and encoded in order to reduce the amount of computing resources required while still maintaining an adequate solution quality.

The minimal encoding first described in [12] uses logarithmically as many qubits as classical binary variables in the problem, at the expense of sacrificing the ability to encode correlations within variables. It has been shown [5] to perform comparably to conventional encodings on both small and large scales for certain problem formulations.

Acknowledgements. This report was compiled

as part of a quantum seminar held at Technische Universität München during the 2023 winter semester. I would like to thank Prof. Christian B. Mendl and my advisor Keefe Huang of the quantum research department at the university for their support in making this project possible.

References

[1] P. Toth and D. Vigo, *The Vehicle Routing Problem*. Society for Industrial and Applied Mathematics, 2022.

[2] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, 2018.

[3] B. Apolloni, N. Cesa-Bianchi, and D. de Falco, "A numerical implementation of quantum annealing," in *Stochastic Processes, Physics and Geometry*, 1988.

[4] S. Feld, C. Roch, T. Gabor, C. Seidel, F. Neukart, I. Galter, W. Mauerer, and C. Linnhoff-Popien, "A hybrid solution method for the capacitated vehicle routing problem using a quantum annealer," *Frontiers in ICT*, 2019.

[5] I. D. Leonidas, A. Dukakis, B. Tan, and D. G. Angelakis, "Qubit efficient quantum algorithms for the vehicle routing problem on nisq processors," 2023.

[6] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," 2014.

[7] S. Hadfield, Z. Wang, B. O'Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas, "From the quantum approximate optimization algorithm to a quantum alternating operator ansatz," *Algorithms*, 2019.

[8] V. Akshay, H. Philathong, M. E. S. Morales, and J. D. Biamonte, "Reachability deficits in quantum approximate optimization," *Physical Review Letters*, 2019.

[9] A. Lucas, "Ising formulations of many NP problems," *Frontiers in Physics*, 2014.

[10] F. Glover, G. Kochenberger, R. Hennig, and Y. Du, "A tutorial on formulating and using QUBO models," *Ann Oper Res*, 2022.

[11] L. Palackal, B. Poggel, M. Wulff, H. Ehm, J. M. Lorenz, and C. B. Mendl, "Quantum-assisted solution paths for the capacitated vehicle routing problem," 2023.

[12] B. Tan, M.-A. Lemonde, S. Thanasilp, J. Tangpanitanon, and D. G. Angelakis, "Qubit-efficient encoding schemes for binary optimisation problems," *Quantum*, 2021.

[13] M. Born and V. Fock, "Beweis des adiabatensatzes," *Zeitschrift für Physik*, 1928.

[14] T. L. Chow, *Introduction to electromagnetic theory: a modern perspective*. Jones & Bartlett Learning, 2006.

A. Schrödinger equation and unitarity

Solving the time dependent Schrödinger equation for time invariant Hamiltonians yields:

$$\begin{aligned} i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle &= H|\psi(t)\rangle \\ i\hbar\ \frac{1}{|\psi(t)\rangle}\ \partial|\psi(t)\rangle &= H\ \partial t \\ i\hbar\ \ln\frac{|\psi(t)\rangle}{|\psi(t_0)\rangle} &= Ht \\ \ln\frac{|\psi(t)\rangle}{|\psi(t_0)\rangle} &= \frac{-i}{\hbar}Ht \\ |\psi(t)\rangle &= e^{\frac{-i}{\hbar}Ht}|\psi(0)\rangle \end{aligned}$$

Since H is an observable describing the energy content of the system eigenstates, it is Hermitian $H^\dagger = H$ with real eigenvalues. Hence:

$$\begin{aligned} U^\dagger U &= (e^{\frac{-i}{\hbar}Ht})^\dagger e^{\frac{-i}{\hbar}Ht} \\ &= e^{\frac{i}{\hbar}H^\dagger t}e^{\frac{-i}{\hbar}Ht} \\ &= e^{\frac{i}{\hbar}Ht}e^{\frac{-i}{\hbar}Ht} \\ &= e^{\frac{i}{\hbar}(Ht-Ht)} \\ &= \mathbb{1} \end{aligned}$$

In this manner the unitary time evolution preserves inner products, ie.

$$\langle U\phi|U\psi\rangle = \langle\phi|U^\dagger U|\psi\rangle = \langle\phi|\psi\rangle$$

for all $|\psi\rangle, |\phi\rangle$, which in conjunction with the Born rule allows one to interpret the wavefunction as a probability density.

B. Commutativity and adiabatic theorem

Commutativity in this case is defined through the Lie bracket operator associated with the space of square matrices, ie. $[A, B] = AB - BA$. Interpreted literally, it defines a metric describing whether the two matrices A and B commute w.r.t. multiplication. A and B commute iff $[A, B] = 0$. Notably, two commuting Hamiltonians share an eigenbasis:

$$A|\phi_i\rangle = \lambda_i|\phi_i\rangle$$

Hence:

$$AB|\phi_i\rangle = BA|\phi_i\rangle = B\lambda_i|\phi_i\rangle = \lambda_i B|\phi_i\rangle$$

The informal version of the adiabatic theorem is stated as follows:

> A physical system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough and if there is a gap between the eigenvalue and the rest of the Hamiltonian's spectrum. [13]

As such, in the case that the initial and final Hamiltonians H_0 and H_p commute only the energies of the eigenstates are reassigned, and not the eigenstates themselves. This runs the risk of equalising the energy levels of the ground and excited states at some point during adiabatic evolution, meaning the system is obliged to evolve infinitely slowly to obtain a useful result due to $T \in \mathcal{O}(\frac{1}{g^2})$.

By the same reasoning, a noncommuting mixer is used in QAOA to allow exploration of the search space by inducing transitions between eigenstates.

C. Ising Model and Hamiltonian

Although the influence of a ferromagnetic particle is given as a magnetic field that weakens over distance, it is sufficient to consider only nearest neighbour interactions as a simplification. The magnetic flux density for a dipole is

$$\mathbf{B} = \frac{\mu_0}{4\pi}\left(\frac{3\mathbf{r}(\mathbf{m}\cdot\mathbf{r})}{r^5} - \frac{\mathbf{m}}{r^3}\right)$$

with $\mathbf{m}$ the magnetic moment [14]. One observes from the first term that for orientations orthogonal to the dipole the field strength drops off substantially, and otherwise decays cubically with distance as given by the second term. For an evenly spaced lattice any two particles have their influence reduced by a factor of $\frac{1}{8}$ for each particle located in between them, thereby justifying the nearest neighbour simplification.

When considering electrons, the spin is antiparallel to the magnetic moment, given by:

$$\mu = \frac{-e}{2m_e}L$$

where L represents angular momentum, so the Hamiltonian should be:

$$H = -\sum_{\langle i\ j\rangle} J_{ij}\sigma_i\sigma_j + \mu\sum_i h_i\sigma_i$$

however conventionally it is given as in 8, with the second term carrying a negative sign.

This results in the interactions being classified as ferromagnetic for $J_{ij} > 0$ and antiferromagnetic for $J_{ij} < 0$.