

Received 29 March 2023; revised 6 July 2023; accepted 2 August 2023; date of publication 10 August 2023; date of current version 7 September 2023.

Digital Object Identifier 10.1109/TQE.2023.3303989

Analysis of the Vehicle Routing Problem Solved via Hybrid Quantum Algorithms in the Presence of Noisy Channels

NISHIKANTA MOHANTY¹ , BIKASH K. BEHERA² ,
AND CHRISTOPHER FERRIE³

¹Centre for Quantum Software and Information, University of Technology Sydney, Sydney, NSW 2007, Australia

²Bikash's Quantum (OPC) Pvt. Ltd., Mohanpur 741246, India

³Centre for Quantum Software and Information, University of Technology Sydney, Sydney, NSW 2007, Australia

Corresponding author: Nishikanta Mohanty (e-mail: nishikanta.m.mohanty@student.uts.edu.au).

This work was supported by the Centre for Quantum Software and Information, University of Technology Sydney and Sydney Quantum Academy.

ABSTRACT The vehicle routing problem (VRP) is an NP-hard optimization problem that has been an interest of research for decades in science and industry. The objective is to plan routes of vehicles to deliver goods to a fixed number of customers with optimal efficiency. Classical tools and methods provide good approximations to reach the optimal global solution. Quantum computing and quantum machine learning provide a new approach to solving combinatorial optimization of problems faster due to inherent speedups of quantum effects. Many solutions of VRP are offered across different quantum computing platforms using hybrid algorithms, such as quantum approximate optimization algorithm and quadratic unconstrained binary optimization. In this work, we build a basic VRP solver for three and four cities using the variational quantum eigensolver on a fixed ansatz. The work is further extended to evaluate the robustness of the solution in several examples of noisy quantum channels. We find that the performance of the quantum algorithm depends heavily on what noise model is used. In general, noise is detrimental, but not equally so among different noise sources.

INDEX TERMS Combinatorial optimization (CO), Ising model, quantum noise channels, variational quantum eigensolver (VQE).

I. INTRODUCTION

Quantum computers, the next generation of computing technology, are expected to solve complex optimization problems much faster than their traditional counterparts. As parallelism is quantum computing's most notable benefit [1], [2], it is only natural to turn to quantum computing to speed up calculations in complicated optimization problems (such as those described by a quantum approximate optimization algorithm (QAOA) [3], adiabatic computation (AC) [4], Grover's algorithm [5], and others). When applied to a multidimensional problem, classical optimization techniques in machine learning (ML) can take a long time to calculate global optimum and consume a lot of CPU and GPU power [6]. In higher dimensional problem spaces, classical algorithms have been shown to be less effective in general [7]. This is due to the fact that NP-hard optimization tasks are often assigned to ML algorithms [6].

VRP comes under the category of routing problems that try to address multiple issues related to fleet management [8]. The objective is always to optimize vehicle movement to minimize the cost or maximize the profit. Notwithstanding the difficulty in delivering quick and dependable solutions to the computationally hard VRP problem, several precise and heuristic techniques have been developed for solving it [8], [9]. Describing the VRP in its simplest form, a single vehicle is tasked to deliver goods at multiple customer locations; also, the vehicle needs to return to pick up additional items when it runs out of goods [10]. The goal is to minimize the cost of service by finding the best feasible combination of routes that begin and terminate at a central location (the depot) while maximizing the reward (often the inverse of the total distance traveled or the mean service time). This problem is computationally difficult to solve, even with only a few hundred customer nodes, and is classified as an NP-hard problem [6], [11].

Any VRP (n, k) involves $(n - 1)$ locations with k vehicles and a depot D [8], [12]. Its solution is a set of routes in which all k vehicles begin and terminate at the given depot D , ensuring that each place is visited only once. The route with the shortest sum total of distance traveled by k vehicles is the ideal one. For a long time, VRP is studied as an extension of the classical traveling salesman problem [9], [13], where now a group of k salesmen has to service collectively $(n - 1)$ locations, such that each location is serviced exactly once [8]. Constraints, such as vehicle capacity or restricted covering time, often complicate the VRP issue in practical settings. As a result, a plethora of conventional and quantum approaches have been presented in an effort to effectively solve the problem. Current quantum approaches for solving optimization problems include QAOA [3], quadratic unconstrained binary optimization (QUBO) [14], [15], quantum annealing [16], [17], [18], and variational quantum eigensolver (VQE) [12], which we will define in detail later.

In this work, we study the VRP in a different light. Here, we explore adding controlled noise to an adapted quantum solution to determine if it improves or degrades the overall results. Recent works in QAOA [19], [20], [21], [22] and VQE algorithms [23] studied the generic effects of noise in these hybrid algorithms. Our work complements these results by analyzing the effects of noise in a detailed gate-based simulation of an algorithm to solve VRP. We analyze the effect of various noise channels on an existing, yet variable, ansatz developed as a solution to VRP. We apply amplitude damping, bit-flip, phase-flip, bit-phase-flip, and depolarizing noise channels to VRP circuits, analyze the effects, and consolidate our findings.

The rest of this article is organized as follows. Section II discusses fundamental mathematical concepts, such as combinatorial optimization (CO), AC, QAOA, and the Ising model. Section III discusses the formulation of VRP using the concepts discussed in the previous section. Section IV covers the basic building blocks of circuits to solve VRP. Section V covers building an ansatz for VRP. Section VI covers the effects of applying noise models on the VRP circuit. Then, Section VII presents the observations from the simulation results. Finally, in Section VIII, we summarize the effects of various noise models on the VRP circuit and future directions of research.

II. MATHEMATICAL BACKGROUND

The fundamental concepts used to solve routing problems involve techniques and procedures from the field of CO. This is followed by converting the mathematical models to a quantum equivalent mathematical model for formulating the objective function. The solution of the objective function is often achieved by maximization or minimization of the function. In this section, we outline the key concepts.

A. COMBINATORIAL OPTIMIZATION

A classical CO problem is finding an optimal object from a finite set of objects. Exhaustive search is impractical in

finding the optimal object due to the potentially high number of objects. Mathematically defining, if s is a string in some set S and m number of clauses, where $s \geq m$, we have a maximization or minimization problem, known as a CO problem. Each clause expects a string parameter and returns a corresponding value [24]. It is the sum over the m clauses that constitute the string's total cost function. If we refer to the input string as z and clauses as C_α , we can write the total cost function as

$$C(z) = \sum_{\alpha=1}^m C_\alpha(z). \quad (1)$$

The objective is to identify $\bar{z} \in S$ such that $C(\bar{z}) \geq C(z)$ for all $z \in S$ (or, in the case of minimization, $C(\bar{z}) \leq C(z)$ for all $z \in S$). Here, \bar{z} is not required to be unique. $C_\alpha(z)$ can take two values 0 or 1 and z can be written as $z = z_0 z_1 z_2 \dots z_{n-1}$ for $z_i \in \{0, 1\}$. Also, considering maximization problems (assuming $C_\alpha(z)$ is a clause), the minimization problems can be studied as $C'_\alpha(z) = 1 - C_\alpha(z)$

$$\begin{aligned} C'(z) &= \sum_{\alpha=0}^{m-1} C'_\alpha(z) = \sum_{\alpha=0}^{m-1} (1 - C_\alpha(z)) \\ &= m - \sum_{\alpha=0}^{m-1} C_\alpha(z) = m - C(z). \end{aligned} \quad (2)$$

B. ADIABATIC QUANTUM COMPUTATION (AQC)

AQC is a theoretical framework of a quantum computer [4], [25]. The adiabatic theorem asserts that if the change to the Hamiltonian is sufficiently gradual, the system remains in the ground state of the given Hamiltonian [26]. The Hamiltonian is an energy operator of a system. In AQC, there are two Hamiltonians: the driver Hamiltonian (H_d) and the problem Hamiltonian (H_p). The driver Hamiltonian (H_d) is the energy operator whose ground state is easy to prepare, whereas the problem Hamiltonian (H_p) is the energy operator whose ground state is obtained after evolution [4]. Interpolation times are proportional to the energy gap between the two lowest states of the Hamiltonian being used.

The procedure begins with an easy-to-prepare ground state (i.e., the ground state of (H_d)) and ends (ideally) with the ground state of (H_p) , which is, in general, not directly characterizable. Mathematically, constitute function $s(t)$ on $[0, T]$ where $s(0) = 0$ and $s(T) = 1$. T is the value of time set high enough for the adiabatic theorem to hold. We define the Hamiltonian, $H(t) = (1 - s(t))H_d + s(t)H_p$. According to the adiabatic theorem, a system maintains its ground state of $H(t)$ across the whole interval $[0, T]$, provided a suitable $s(t)$; hence, the system is in the initial ground state (H_d) at time $t = 0$, and it will evolve into the intended ground state (H_p) at time $t = T$. In general, it is challenging to assess the integral describing the temporal evolution under this time-dependent Hamiltonian [27]

$$U(t) = \tau \exp \left\{ \frac{-i}{\hbar} \int_0^t H(T) dT \right\}. \quad (3)$$

It is possible to assess this Hamiltonian using Trotterization methods [28]. We divide $U(T)$ into intervals of δt small enough such that the Hamiltonian is almost constant over them. This permits us to use the much more streamlined formula for the Hamiltonian that is independent of time. Assuming $U(b, a)$ is the time evolution from instant a to instant b

$$\begin{aligned} U(T, 0) &= U(T, T - \delta t)U(T - \delta t, T - 2\delta t) \cdots U(\delta t, 0) \\ &= \prod_{j=1}^p U(j\delta t, (j-1)\delta t) \\ &\approx \prod_{j=1}^p e^{-iH(j\delta t)\delta t} \end{aligned} \quad (4)$$

where the approximation gets better as p gets larger (or, in other words, as δt gets smaller), and where δt is measured in \hbar . Now using the approximation $e^{i(A+B)x} = e^{iAx}e^{iBx} + O(x^2)$ and adding Hamiltonian $H(j\delta t) = (1 - s(j\delta t))H_D + s(j\delta t)H_P$, the integral $U(t)$ becomes

$$U(T, 0) \approx \prod_{j=1}^p e^{-i(1-s(j\delta t))H_D\delta t} e^{-is(j\delta t)H_P\delta t}. \quad (5)$$

AQC may be approximated by allowing the system to develop under H_P for a small $s(j\delta t)\delta t$ and then H_D for a small $(1-s(j\delta t))\delta t$, and unitaries can be derived for these operations using $U = e^{-i\alpha H\delta t}$. Here, α is an integer in the range $[0, 1]$, and this includes the scaling resulting from $s(j\delta t)$. AQC forms the theoretical basis of the variational quantum algorithm QAOA, which is discussed briefly in the next section.

C. QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

QAOA is a variational algorithm proposed by Farhi et al. in 2014 [3], [4]. This algorithm relies on the framework of AQC. Because of its use of both conventional and quantum methods, this algorithm is considered as hybrid algorithm. In the previous section, quantum AC drove the system from the eigenstate of driver Hamiltonian to that of the eigenstate of problem Hamiltonian.

In the context of optimization problems, the problem Hamiltonian can be written as

$$C|z\rangle = \sum_{\alpha=1}^m C_{\alpha(z)}|z\rangle. \quad (6)$$

The maximum energy eigenstate of C solves the CO problem. Similarly, for driver Hamiltonian, we use

$$B = \sum_{j=1}^n \sigma_j^x \quad (7)$$

where σ_j^x represents the Pauli operator σ^x on the bit z_j . B is also known as the mixing operator. Let us also define $U_c(\gamma) = e^{-i\gamma C}$ and $U_B(\beta) = e^{-i\beta B}$, which lets the system evolve under C for some γ amount of time and under B for

some β amount of time, respectively. QAOA then constructs a state

$$|\beta, \gamma\rangle = e^{-i\beta_p B} e^{-i\gamma_p C} \cdots e^{-i\beta_2 B} e^{-i\gamma_2 C} e^{-i\beta_1 B} e^{-i\gamma_1 C} |s\rangle \quad (8)$$

where $|s\rangle$ is a superposition state of all input qubits. The expectation value of the cost function $\sum_{\alpha=1}^m \langle \beta, \gamma | C_{\alpha} | \beta, \gamma \rangle$ gives the solution or the approximate solution of the problem after simplex or gradient-based optimization starting with some initial set of parameters [29].

D. ISING MODEL

The Ising model of ferromagnetism is a well-established mathematical model used extensively in the field of statistical mechanics [30], [31]. There are two possible states for the magnetic dipole moments of atomic ‘‘spins’’ (+1 and 1), each of which is represented by a discrete variable in the model. Each spin is able to communicate with its neighbors because of how they are organized in a graph, usually a lattice (where the local structure regularly repeats in all directions). The system tends toward the lowest energy state when neighboring spins agree, but heat interrupts this tendency, allowing for the emergence of alternate structural phases. The model serves as a simplification of reality that may be used to spot phase transitions [32]. Using the following Hamiltonian, we can describe the sum of the spin energies

$$H_c = - \sum_{\langle i, j \rangle} J_{ij} \sigma_i \sigma_j - h \sum \sigma_i \quad (9)$$

where J_{ij} represents the interaction between i and j , which are adjacent spins, and h represents an external magnetic field. If J is positive, the ground state at $h = 0$ is a ferromagnet. If J is negative, the ground state at $h = 0$ is an antiferromagnet for a bipartite lattice. Hence, for simplification and in the context of this document, we can write the Hamiltonian as

$$H_c = - \sum_{\langle i, j \rangle} J_{ij} \sigma_i^z \sigma_j^z - \sum h_i \sigma_i^x. \quad (10)$$

Here, σ_z and σ_x represent Pauli z and x operator. For simplification, we can consider the following conditions to be ferromagnetic ($J_{ij} > 0$), $h = 0$ assuming no external influence on the spin. Thus, we can rewrite the Hamiltonian as follows:

$$H_c = - \sum_{\langle i, j \rangle} J_{ij} \sigma_i^z \sigma_j^z = - \sum_{\langle i, j \rangle} \sigma_i^z \sigma_j^z. \quad (11)$$

E. VARIATIONAL QUANTUM EIGENSOLVER

VQE is a hybrid quantum-classical technique used to determine the eigenvalue of a large matrix or Hamiltonian H [33]. The basic objective of this method is to search for a trial qubit state of a wave function $|\psi(\vec{\theta})\rangle$, which is dependent on a parameter set $\vec{\theta} = \theta_1, \theta_2, \dots$ also called as the variational parameters. By quantum theory, the expectation of an observable or Hamiltonian H in a state $|\psi(\vec{\theta})\rangle$ can be expressed as

$$E(\vec{\theta}) = \langle \psi(\vec{\theta}) | H | \psi(\vec{\theta}) \rangle. \quad (12)$$

By spectral decomposition, H can be written as

$$H = \sum_{i=1}^n \lambda_i |\psi\rangle_i \langle\psi|_i \quad (13)$$

where λ_i and $|\psi\rangle_i$ are the eigenvalues and eigenstates, respectively, of matrix H . Also, the eigenstates of H are orthogonal so $\langle\psi_i | \psi_j\rangle = 0$ if $i \neq j$. The wave function $|\psi(\vec{\theta})\rangle$ is represented as the superposition of eigenstates

$$|\psi(\vec{\theta})\rangle = \sum_{i=1}^n \alpha_i(\vec{\theta}) |\psi\rangle_i. \quad (14)$$

Thus, the expectation becomes

$$E(\vec{\theta}) = \sum_{i=1}^n |\alpha_i(\vec{\theta})|^2 \lambda_i. \quad (15)$$

Clearly, $E(\vec{\theta}) \geq \lambda_{\min}$. So in VQE algorithm, we vary the parameters $\vec{\theta} = \theta_1, \theta_2, \dots$ until $E(\vec{\theta})$ is minimized. This characteristic of VQE is important for addressing CO problems in which a parameterized circuit is used to construct the trial state of the algorithm, and $E(\vec{\theta})$ is the cost function, which is the expected value of the Hamiltonian in the trial state. On the assumption that the ansatz or parameterized quantum circuit may describe the ground state of the system, the ground state of the target Hamiltonian may be obtained by iterative minimization of the cost function. At each optimization stage, the optimization procedure employs a classical optimizer that employs a quantum computer to analyze the cost function and calculate its gradient.

III. MODELING VRP IN QUANTUM

To find a solution to the vehicle routing problem, we can map the cost function to an Ising Hamiltonian H_c [32]. The minimization of Ising Hamiltonian H_c gives the solution to the problem. To begin, let us consider an arbitrary connected graph of n vertices and a binary decision variable x_{ij} who has a value 1 if there exists an edge between i and j for edge weight $w_{ij} > 0$ else; the value is 0. To represent the VRP problem, we need $n \times (n - 1)$ decision variables. For every edge from $i \rightarrow j$, we define two sets of nodes source $[i]$ and target $[j]$. The set source $[i]$ contains the nodes j to which i sends an edge $j \in \text{source}[i]$. The set target $[j]$ contains the nodes i to which i sends an edge $i \in \text{target}[j]$. We define VRP as follows [12], [34]:

$$\text{VRP}(n, k) = \min_{\{x_{ij}\}_{i \rightarrow j \in \{0,1\}}} \sum_{i \rightarrow j} w_{ij} x_{ij} \quad (16)$$

where k is the number of vehicles and n is the total number of locations. Considering the starting location as 0th location or Depot D , we have $n - 1$ locations for vehicles to travel. This is subject to the following constraints:

$$\sum_{j \in \text{source}[i]} x_{ij} = 1 \quad \forall i \in \{1, \dots, n - 1\}$$

$$\sum_{j \in \text{target}[i]} x_{ji} = 1 \quad \forall i \in \{1, \dots, n - 1\}$$

$$\sum_{j \in \text{source}[0]} x_{0j} = k$$

$$\sum_{j \in \text{target}[0]} x_{j0} = k$$

$$u_i - u_j + Qx_{ij} \leq Q - q_j \quad \forall i \sim j, i, j \neq 0$$

$$q_i \leq u_i \leq Q \quad \forall i, i \neq 0. \quad (17)$$

The first two constraints impose the restriction that the delivering vehicle must visit each node only once. The middle two constraints enforce the restriction that the vehicle must return to the depot after delivering the goods. The last two constraints impose the subtour elimination conditions and are bound on u_i , with $Q > q_j > 0$, and $u_i, Q, q_i \in \mathbb{R}$.

For the VRP equation, we can form the Hamiltonian of VRP as follows [12]:

$$H_{\text{VRP}} = H_A + H_B + H_C + H_D + H_E$$

$$H_A = \sum_{i \rightarrow j} w_{ij} x_{ij}$$

$$H_B = A \sum_{i \in \{1, \dots, n-1\}} \left(1 - \sum_{j \in \text{source}[i]} x_{ij} \right)^2$$

$$H_C = A \sum_{i \in \{1, \dots, n-1\}} \left(1 - \sum_{j \in \text{target}[i]} x_{ji} \right)^2$$

$$H_D = A \left(k - \sum_{j \in \text{source}[0]} x_{0j} \right)^2$$

$$H_E = A \left(k - \sum_{j \in \text{target}[0]} x_{j0} \right)^2 \quad (18)$$

where $A > 0$ is a constant.

The set of all binary decision variables x_{ij} can be represented in vector form as

$$\vec{x} = [x_{(0,1)}, x_{(0,2)}, \dots, x_{(1,0)}, x_{(1,2)}, \dots, x_{(n-1,n-2)}]^T. \quad (19)$$

Using the above-mentioned vector, we can define two additional vectors for each node

$$\vec{z}_{S[i]} = \vec{x} \ni x_{ij} = 1, x_{kj} = 0, k \neq i$$

$$\forall j, k \in \{0, \dots, n - 1\}$$

$$\vec{z}_{T[i]} = \vec{x} \ni x_{ji} = 1, x_{jk} = 0, k \neq i$$

$$\forall j, k \in \{0, \dots, n - 1\}. \quad (20)$$

$$\sum_{j \in \text{source}[i]} x_{ij} = \vec{z}_{S[i]}^T \vec{x}$$

$$\sum_{j \in \text{target}[i]} x_{ji} = z_{T[i]}^T \vec{x}. \quad (21)$$

The aforementioned vectors will assist in the formulation of the QUBO model of VRP. For a linked graph $G = (N, V)$, the QUBO model [14], [15], [35], [36] is defined as

$$f(x)_{\text{QUBO}} = \min_{x \in \{0,1\}^{(N \times V)}} x^T Qx + g^T x + c \quad (22)$$

where Q is a quadratic coefficient of the edge weights, g is a linear coefficient of the node weights, and c is a constant. In order to find these coefficients in the QUBO formations of H_{VRP} given in (18), we first put in (21) in terms H_B and H_c , respectively, then expand and regroup the expression of H_{VRP} according to (22)

$$\begin{aligned} H = & A \sum_{i=0}^{n-1} [z_{S[i]} z_{S[i]}^T + z_{T[i]} z_{T[i]}^T] \vec{x}^2 \\ & + w^T \vec{x} - 2A \sum_{i=1}^{n-1} [z_{S[i]}^T + z_{T[i]}^T] \vec{x} \\ & - 2Ak [z_{S[0]}^T + z_{T[0]}^T] \vec{x} + 2A(n-1) + 2Ak^2. \end{aligned} \quad (23)$$

Hence, for QUBO formulation of (18), we get the coefficients $Q(n(n-1) \times n(n-1))$, $g(n(n-1) \times 1)$ and c

$$\begin{aligned} Q = & A \left[[z_{T[0]}, \dots, z_{T[n-1]}]^T [z_{T[0]}, \dots, z_{T[n-1]}] \right. \\ & \left. + (\mathbb{I}_n \otimes \mathbb{J}(n-1, n-1)) \right] \\ g = & W - 2Ak \left((e_0 \otimes \mathbb{J}_{n-1}) + [z_{T[0]}]^T \right) \\ & + 2A (\mathbb{J}_n \otimes \mathbb{J}_{n-1}) \\ c = & 2A(n-1) + 2Ak^2. \end{aligned} \quad (24)$$

Here, \mathbb{J} is the matrix of all ones, \mathbb{I} is the identity matrix, and $e_0 = [1, 0, \dots, 0]^T$.

The binary decision variable x_{ij} is transformed to spin variable $s_{ij} \in \{-1, 1\}$ as $x_{ij} = (s_{ij} + 1)/2$.

From the above-mentioned equations, we can expand (22) to form the Ising Hamiltonian of VRP [14]

$$H_{\text{Ising}} = - \sum_i \sum_{i < j} J_{ij} s_i s_j - \sum_i h_i s_i + d. \quad (25)$$

Here, the terms J_{ij} , h_i , and d are defined as follows:

$$\begin{aligned} J_{ij} = & - \frac{Q_{ij}}{2}, \quad \forall i < j \\ h_i = & \frac{g_i}{2} + \sum \frac{Q_{ij}}{4} + \sum \frac{Q_{ji}}{4} \\ d = & c + \sum_i \frac{g_i}{2} + \sum_i \sum_j \frac{Q_{ij}}{4}. \end{aligned} \quad (26)$$

IV. ANALYSIS AND CIRCUIT BUILDING

In this section, we create a gate-based circuit to realize the above-mentioned formulation using the IBM gate

model, which we have implemented using the Qiskit framework [37]. For any arbitrary VRP problem using qubits, we begin with the state of $|+\rangle^{\otimes n(n-1)}$ the ground state of H_{mixer} by applying the Hadamard to all qubits initialized as zero states, and we prepare the following state:

$$\begin{aligned} |\beta, \gamma\rangle = & e^{-iH_{\text{mixer}}\beta_p} e^{-iH_{\text{cost}}\gamma_p} \dots \\ & \dots e^{-iH_{\text{mixer}}\beta_0} e^{-iH_{\text{cost}}\gamma_0} |+\rangle^{n \otimes (n-1)}. \end{aligned} \quad (27)$$

The energy E of the state $|\beta, \gamma\rangle$ is calculated by the expectation of H_{cost} from (12). Once again, the H_{cost} term may be expressed in terms of Pauli operators using the Ising model, as

$$H_{\text{cost}} = - \sum_i \sum_{i < j} J_{ij} \sigma_i^z \sigma_j^z - \sum_i h_i \sigma_i^z - d. \quad (28)$$

Thus, for a single term of state in $|\beta, \gamma\rangle$ as β_0 and γ_0 , the expression reads $e^{-iH_{\text{mixer}}\beta_0} e^{-iH_{\text{cost}}\gamma_0}$. The first term H_{cost} can be expanded to following:

$$\begin{aligned} e^{iJ_{ij}\gamma_0\sigma_i\sigma_j} = & \cos J_{ij}\gamma_0 I + i \sin J_{ij}\gamma_0 \sigma_i \sigma_j \\ = & \begin{bmatrix} e^{iJ_{ij}\gamma_0} & 0 & 0 & 0 \\ 0 & e^{-iJ_{ij}\gamma_0} & 0 & 0 \\ 0 & 0 & e^{-iJ_{ij}\gamma_0} & 0 \\ 0 & 0 & 0 & e^{iJ_{ij}\gamma_0} \end{bmatrix} \\ = & M. \end{aligned} \quad (29)$$

Applying a CNOT gate on, before, and after the aforementioned matrix “ M ,” we can swap the diagonal elements

$$\text{CNOT}(M)\text{CNOT} = \begin{bmatrix} e^{iJ_{ij}\gamma_0} & 0 & 0 & 0 \\ 0 & e^{-iJ_{ij}\gamma_0} & 0 & 0 \\ 0 & 0 & e^{iJ_{ij}\gamma_0} & 0 \\ 0 & 0 & 0 & e^{-iJ_{ij}\gamma_0} \end{bmatrix}. \quad (30)$$

Observing the upper and lower blocks of the matrix, we can rewrite

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} e^{iJ_{ij}\gamma_0} & 0 \\ 0 & e^{-iJ_{ij}\gamma_0} \end{bmatrix} = I \otimes e^{iJ_{ij}\gamma_0} \begin{bmatrix} 1 & 0 \\ 0 & e^{-2iJ_{ij}\gamma_0} \end{bmatrix}. \quad (31)$$

$\begin{bmatrix} 1 & 0 \\ 0 & e^{-2iJ_{ij}\gamma_0} \end{bmatrix}$ is a phase gate. Looking at the second term of H_{cost} , we get

$$\begin{aligned} H_{\text{cost}} = & \sum_i h_i \sigma_i^z \\ e^{ih_i\gamma_0\sigma_i^z} = & \cos h_i\gamma_0 I + i \sin h_i\gamma_0 \sigma_i^z \\ = & \cos h_i\gamma_0 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + i \sin h_i\gamma_0 \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \\ = & \begin{bmatrix} e^{ih_i\gamma_0} & 0 \\ 0 & e^{-ih_i\gamma_0} \end{bmatrix} \\ = & e^{ih_i\gamma_0} \begin{bmatrix} 1 & 0 \\ 0 & e^{-2ih_i\gamma_0} \end{bmatrix}. \end{aligned} \quad (32)$$

Fig. 1(a) depicts the basic circuit with two qubits along with gate selections for H_{cost} .

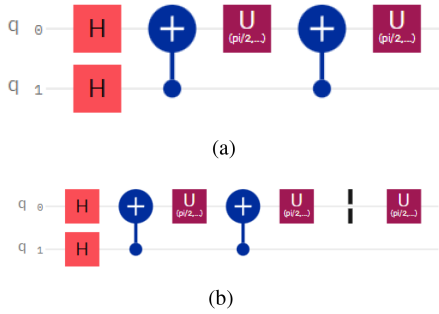


FIGURE 1. (a) Sample circuit showing gate selections for H_{cost} . (b) Sample circuit showing gate selections with additional U gate after barrier for H_{mixer} . Note: The sample circuits displayed in the figures represent the building blocks of the actual circuit and do not represent actual angles that are obtained as a solution of VRP using VQE.

Similarly, H_{mixer} can be derived as follows:

$$H_{mixer} = - \sum_i \sigma_x. \quad (33)$$

Considering a single term of H_{mixer} and taking the unitary

$$\begin{aligned} e^{-iH_{mixer}\beta_0} &= e^{-i(-\sigma_x)\beta_0} \\ &= e^{i\sigma_x\beta_0} \\ &= \cos \beta_0 \mathbb{I} + i \sin \beta_0 \sigma_x \\ &= \begin{bmatrix} \cos \beta_0 & i \sin \beta_0 \\ i \sin \beta_0 & \cos \beta_0 \end{bmatrix}. \end{aligned} \quad (34)$$

The IBMQ U gate is defined as follows:

$$U = \begin{bmatrix} \cos \theta/2 & -e^{i\lambda} \sin \theta/2 \\ e^{i\phi} \sin \theta/2 & e^{i(\lambda+\phi)} \cos \theta/2 \end{bmatrix}. \quad (35)$$

Comparing (31)–(35), we can establish relation of circuit parameters with γ and β to U gate, which will form building blocks of circuit. From Fig. 1(a), the circuit represents H_{cost} term, where the first U gate takes the parameters, $\theta = 0$, $\phi = -2J_{ij}\gamma_0$, and $\lambda = 0$, and the second U gate takes the parameters, $\theta = 0$, $\phi = -2h_i\gamma_0$, and $\lambda = 0$. Similarly, in Fig. 1(b), the U gate after the barrier represents the H_{mixer} term having the parameters $\theta = 2\beta_0$, $\phi = \pi/2$, and $\lambda = -\pi/2$.

V. VQE SIMULATION OF VRP

We construct the VRP circuit using the above-mentioned equations and create the Hamiltonians for three-city and four-city scenarios. Since we need $n(n-1)$ qubits, we end up with only Hamiltonians and circuits with 6 and 12 qubits. Beyond four cities, it is impossible to simulate in a classical desktop computer due to memory limitations. We create the ansatz using a quantum circuit defined in the previous section and run it across various VQE optimizers available in the IBM Qiskit framework: COBYLA, L_BFGS_B, SPSA, and SLSQP. We run the circuit up to four layers across all the optimizers and obtain the results depicted in Figs. 2 and 3. The primary difference between Figs. 2 and 3 is the while the

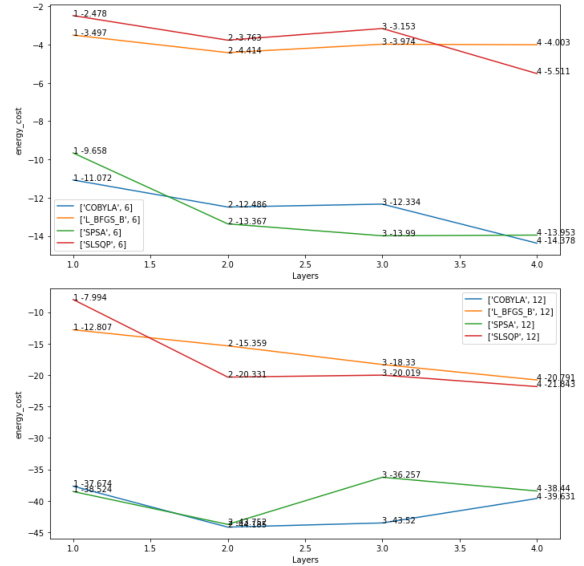


FIGURE 2. Plot illustrating the circuit simulation of VRP with five layers using various optimizers (COBYLA, L_BFGS_B, SLSQP, and SPSA). The plot consists of two separate graphs depicting the simulation output of 6 and 12 qubit circuits, respectively. Each plot, in turn, consists of four lines indicating energy values for different optimizers. The average value at each layer is represented in pairs with (layer, average energy) format.

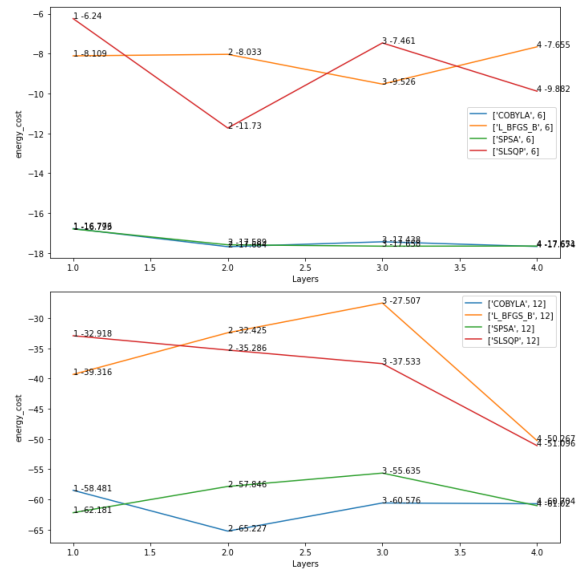


FIGURE 3. Plot illustrating the circuit simulation of VRP with five layers using various optimizers (COBYLA, L_BFGS_B, SLSQP, and SPSA). The plot consists of two graphs depicting the simulation output of 6 and 12 qubit circuits, respectively. Each plot consists of four lines indicating energy values for different optimizers. The minimum value at each layer is represented in pairs with (layer, minimum energy) format.

former represents average energy value the later represents minimum energy value of 15 consecutive runs at each layer.

The tables for these figures are presented in the Appendix with Tables 4 and 5. The figures are derived from 15 consecutive runs of the VRP circuit, each with all the optimizers and four layers. While the average energy values of VRP

simulations decrease as optimizations increase across layers for most of the six qubit circuits, the same trend is not observed for 12 qubit circuits. Also, the energy curves vary significantly across all optimizers. Similarly, for minimum energy graphs in Fig. 3, energy values have no clear decreasing trend as optimization layers increase. Yet from the minimum energy graphs, we can reliably say that the protocol has achieved the minimum energy value. However they do not always follow the downward trend or stay at the same level as optimization layers increase. This, of course, is heavily dependent on the optimizer. Thus, when selecting an optimizer for simulation, we chose the optimizer that achieves the lowest minimum, the fewest number of optimization layers. In summary, we have found that COBYLA is the best-performing optimizer, followed by SPSA, L_BFGS_B, and SLSQP. However, in the following, when we pass the circuit through various noise models, we will use only the COBYLA optimizer.

VI. NOISE MODEL SIMULATION OF VQE

In a noisy quantum environment, a pure input state will be transformed into a mixed state represented as a density matrix [21], [38]. In the case of a 6-qubit state pure state $|\psi\rangle_{q_0q_1q_2q_3q_4q_5}$, the density matrix can be defined as $\rho = |\psi\rangle_{q_0q_1q_2q_3q_4q_5} \langle\psi|_{q_0q_1q_2q_3q_4q_5}$. After the implementation of the noise model, the density matrix takes the following form:

$$\xi_r(\rho) = \sum_m \left(E_m^{rq0} \right) \left(E_m^{rq1} \right) \dots \left(E_m^{rq5} \right) \rho \times \left(E_m^{xq0} \right)^\dagger \left(E_m^{xq1} \right)^\dagger \dots \left(E_m^{xq5} \right)^\dagger \quad (36)$$

where $r \in \{A, B, W, F, D\}$. The elements of the noise channels are described as follows, A is amplitude damping noise, B is bit-flip noise, W is phase-flip noise, F is bit-phase-flip noise, and D is depolarizing noise. We apply these noise channels to our VRP circuit and ansatz, which is variable based on the number of qubits (6 or 12) and layers (1 to 5). For simulation purposes, we choose the optimizer COBYLA as it has the best performance characteristics in the simulation of VQE. We restrict the noise probability to 0.5 as noisy environments beyond this noise level are unlikely and irrelevant in practice. The following sections discuss the noise channels and operators we experimented on in the VRP circuit.

A. AMPLITUDE DAMPING

The energy dissipation is a consequence of the interaction of the quantum system with an amplitude-damping channel. A quantum system gaining or losing energy from or to its environment is described as a change in amplitude rather than phase [38], [39]. If κ_A is the probability of gain or loss of amplitude or decoherence rate, the Kraus operators of amplitude damping channel can be described as follows:

$$E_0^A = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1 - \kappa_A} \end{bmatrix}$$

$$E_1^A = \sqrt{\kappa_A} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}. \quad (37)$$

B. BIT-FLIP NOISE

Random bit-flip errors characterize bit-flip noise [39] with probability κ_B . Thus, the Kraus operators of the bit-flip noise channel can be described as

$$E_0^B = \sqrt{1 - \kappa_B} I$$

$$E_1^B = \sqrt{\kappa_B} X = \sqrt{\kappa_B} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \quad (38)$$

C. PHASE-FLIP NOISE

Phase-flip noise alters the phase parameter of the quantum system without exchange of energy [38], [39]. The decoherence rate or the phase-flip noise parameter also follows the simple Bernoulli distribution with probability parameter κ_W . Thus, the Kraus operators of phase-flip noise channel can be defined as follows:

$$E_0^W = \sqrt{1 - \kappa_W} I$$

$$E_1^W = \sqrt{\kappa_W} Z = \sqrt{\kappa_W} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \quad (39)$$

D. BIT-PHASE FLIP NOISE

Bit-phase flip noise channel is characterized by a combination of random bit-flip errors and a change in the quantum system's phase information without energy loss [39]. Like other noise channels, the decoherence rate or the combined probability of bit-phase flip error follows the distribution κ_F . The Kraus operator of the bit-phase flip channel could be given by

$$E_0^F = \sqrt{1 - \kappa_F} I$$

$$E_1^F = \sqrt{\kappa_F} Y = \sqrt{\kappa_F} \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}. \quad (40)$$

E. DEPOLARIZING NOISE

A depolarizing noise channel leaves the system untouched or replaces it with a maximally mixed state of I/d for a d -level quantum system. The decoherence rate or the depolarization noise probability follows the distribution with parameter κ_D . The Kraus operators are as follows:

$$E_0^D = \sqrt{1 - \kappa_D} I$$

$$E_1^D = \sqrt{\frac{\kappa_D}{3}} X = \sqrt{\frac{\kappa_D}{3}} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$E_2^D = \sqrt{\frac{\kappa_D}{3}} Y = \sqrt{\frac{\kappa_D}{3}} \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$E_3^D = \sqrt{\frac{\kappa_D}{3}} Z = \sqrt{\frac{\kappa_D}{3}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \quad (41)$$

It is to be noted that, in all cases, the noise channel is applied locally to each qubit in the circuit.

TABLE 1 VQE Simulation of Amplitude Damping, Bit-Flip, Phase-Flip, Bit-Phase-Flip, and Depolarizing Channel for 6 and 12 Qubits With Four Layers Involving Optimizer COBYLA, Where Energy Costs are Averaged Over Ten Simulations

Qubits	Layers	Noise_Parameter	Noise_Model				
			Amplitude_Damping	Bit_Flip	Bit_Phase_Flip	Depolarizing_Noise	Phase_Flip
1	0.05	-12.053	-13.768	-4.247	-4.949	-2.033	
	0.1	-9.273	-9.577	-2.571	-2.437	-0.447	
	0.15	-10.121	-9.661	-3.383	-0.750	-0.036	
	0.2	-12.928	-9.614	-3.297	-0.608	0.049	
	0.25	-12.491	-10.965	-4.170	-0.416	0.058	
	0.3	-12.381	-10.883	-3.761	-0.261	0.059	
	0.35	-11.519	-9.809	-4.724	-0.045	0.059	
	0.4	-11.040	-9.881	-3.535	0.006	0.059	
	0.45	-9.440	-10.531	-2.724	0.034	0.059	
	0.5	-11.470	-11.999	-3.118	0.050	0.059	
2	0.05	-13.698	-4.262	-1.668	-2.377	-1.601	
	0.1	-12.773	-6.576	-1.118	-0.748	-0.313	
	0.15	-12.495	-6.282	-1.313	-0.305	-0.055	
	0.2	-13.909	-5.872	-1.411	-0.052	0.058	
	0.25	-13.171	-5.002	-1.621	0.029	0.058	
	0.3	-12.834	-1.198	-1.699	0.055	0.059	
	0.35	-13.162	-1.603	-1.468	0.058	0.059	
	0.4	-12.760	-3.024	-1.304	0.059	0.059	
	0.45	-13.789	-3.790	-1.236	0.059	0.059	
	0.5	-12.986	-3.075	-1.237	0.059	0.059	
3	0.05	-13.924	-2.332	-0.551	-1.167	-1.755	
	0.1	-14.026	-1.392	-0.605	-0.396	-0.476	
	0.15	-14.999	-3.239	-0.539	-0.031	-0.014	
	0.2	-12.909	-1.634	-0.492	0.053	0.058	
	0.25	-14.413	-2.651	-0.407	0.058	0.059	
	0.3	-13.029	-0.910	-0.600	0.059	0.059	
	0.35	-14.246	-1.864	-0.723	0.059	0.059	
	0.4	-13.519	-1.784	-0.574	0.059	0.059	
	0.45	-15.388	-1.621	-0.620	0.059	0.059	
	0.5	-14.377	-0.498	-0.693	0.059	0.059	
4	0.05	-15.659	-0.969	-0.411	-0.681	-1.574	
	0.1	-15.055	-1.494	-0.446	-0.852	-0.357	
	0.15	-15.560	-0.813	-0.343	0.047	-0.022	
	0.2	-14.661	-1.512	-0.101	0.059	0.059	
	0.25	-14.771	-0.502	-0.219	0.059	0.059	
	0.3	-16.147	-0.193	-0.368	0.059	0.059	
	0.35	-13.608	-0.238	-0.440	0.059	0.059	
	0.4	-13.916	-0.012	-0.059	0.059	0.059	
	0.45	-13.427	-0.886	-0.241	0.059	0.059	
	0.5	-15.757	-0.074	-0.221	0.059	0.059	
6	0.05	-42.626	-31.086	-1.522	-3.741	-1.056	
	0.1	-42.157	-33.947	-1.743	-0.919	-0.514	
	0.15	-38.465	-22.328	-2.430	-0.669	-0.660	
	0.2	-37.787	-38.835	-3.391	-0.637	-0.581	
	0.25	-46.766	-20.722	-2.142	-0.599	-0.600	
	0.3	-41.555	-30.384	-0.982	-0.653	-0.567	
	0.35	-37.722	-37.187	-2.105	-0.512	-0.444	
	0.4	-38.388	-35.172	-2.454	-0.618	-0.548	
	0.45	-40.835	-35.500	-3.122	-0.545	-0.482	
	0.5	-47.219	-48.876	-1.586	-0.565	-0.577	
2	0.05	-47.215	-13.251	-0.416	-0.718	-0.658	
	0.1	-43.644	-11.857	-0.419	-0.255	-0.410	
	0.15	-40.368	-7.993	-0.299	-0.208	-0.383	
	0.2	-45.283	-2.685	-0.417	-0.242	-0.178	
	0.25	-48.987	-7.242	-0.359	-0.231	-0.263	
	0.3	-35.422	-5.921	-0.372	-0.209	-0.192	
	0.35	-44.922	-0.137	-0.431	-0.454	-0.255	
	0.4	-43.944	-9.986	-0.400	-0.113	-0.313	
	0.45	-46.463	-0.931	-0.556	-0.444	-0.274	
	0.5	-43.890	-1.993	-0.434	-0.259	-0.211	
3	0.05	-46.368	-1.086	-0.409	-0.713	-0.662	
	0.1	-40.817	-0.656	-0.460	-0.690	-0.520	
	0.15	-37.797	-0.649	-0.493	-0.709	-0.786	
	0.2	-39.519	-4.338	-0.465	-0.649	-0.626	
	0.25	-48.781	-3.428	-0.402	-0.658	-0.823	
	0.3	-40.945	-1.416	-0.349	-0.576	-0.736	
	0.35	-38.358	-3.589	-0.288	-0.646	-0.529	
	0.4	-42.179	-3.328	-0.447	-0.605	-0.539	
	0.45	-41.264	-0.603	-0.585	-0.700	-0.523	
	0.5	-44.605	-2.647	-0.345	-0.613	-0.593	
4	0.05	-46.110	-0.565	-0.509	-0.643	-0.533	
	0.1	-42.539	-1.276	-0.630	-0.628	-0.577	
	0.15	-42.971	-0.596	-0.616	-0.650	-0.611	
	0.2	-44.977	-0.560	-0.553	-0.676	-0.653	
	0.25	-39.561	-0.781	-0.508	-0.567	-0.680	
	0.3	-44.023	-0.624	-0.531	-0.651	-0.539	
	0.35	-40.907	-0.657	-0.507	-0.599	-0.606	
	0.4	-43.012	-2.570	-0.597	-0.594	-0.585	
	0.45	-43.650	-0.604	-0.736	-0.569	-0.681	
	0.5	-42.357	-0.608	-0.608	-0.571	-0.686	

TABLE 2 For 6 and 12 Qubits With Four Layers and Using COBYLA as an Optimizer, the Table Above Shows the Average Deviation from the Classical Minimum (Over Ten Simulations) for VQE Simulations Utilizing Amplitude Damping, Bit-Flip, Phase-Flip, Bit-Phase-Flip, and Depolarizing Channels

Qubits	Layers	Noise_Parameter	Noise_Model				
			Amplitude_Damping	Bit_Flip	Bit_Phase_Flip	Depolarizing_Noise	Phase_Flip
1	0.05	8.412	8.107	15.113	15.247	17.238	
	0.1	7.563	8.023	14.301	16.934	17.649	
	0.15	4.757	8.070	12.388	17.076	17.734	
	0.2	5.194	6.719	13.514	17.269	17.742	
	0.25	5.303	6.801	13.923	17.423	17.744	
	0.3	6.165	7.875	12.961	17.639	17.744	
	0.35	6.644	7.803	14.149	17.690	17.744	
	0.4	8.244	7.153	14.960	17.718	17.744	
	0.45	6.215	5.686	14.567	17.734	17.744	
	0.5	3.986	13.423	16.017	15.307	16.084	
2	0.05	4.911	11.108	16.567	16.936	17.371	
	0.1	5.189	11.402	16.372	17.379	17.630	
	0.15	3.775	11.813	16.273	17.632	17.742	
	0.2	4.513	14.683	16.063	17.714	17.742	
	0.25	4.851	16.487	15.985	17.739	17.744	
	0.3	6.081	16.814	16.217	17.742	17.744	
	0.35	4.925	16.292	17.080	17.289	17.308	
	0.4	4.925	14.661	16.380	17.744	17.744	
	0.45	3.895	13.894	16.449	17.744	17.744	
	0.5	4.698	14.610	16.447	17.744	17.744	
3	0.05	1.760	15.352	17.133	16.518	15.929	
	0.1	3.658	16.292	17.080	17.289	17.308	
	0.15	2.686	14.445	17.146	17.653	17.670	
	0.2	4.776	16.051	17.192	17.737	17.742	
	0.25	3.271	15.033	17.278	17.742	17.743	
	0.3	2.655	16.774	17.084	17.744	17.744	
	0.35	3.438	15.820	16.961	17.744	17.744	
	0.4	4.165	15.901	17.110	17.744	17.744	
	0.45	2.297	16.063	17.065	17.744	17.744	
	0.5	3.307	17.187	16.991	17.744	17.744	
4	0.05	2.025	16.715	17.273	17.003	16.111	
	0.1	2.629	16.190	17.238	17.632	17.328	
	0.15	2.124	17.472	17.342	17.732	17.662	
	0.2	3.023	16.373	17.583	17.743	17.743	
	0.25	2.914	17.182	17.465	17.744	17.744	
	0.3	1.537	17.492	17.316	17.744	17.744	
	0.35	3.438	16.076	17.447	17.744	17.744	
	0.4	3.769	17.672	17.246	17.744	17.744	
	0.45	4.257	16.798	17.443	17.744	17.744	
	0.5	1.927	17.610	17.463	17.744	17.744	
6	0.05	23.058	34.599	64.163	61.943	64.628	
	0.1	23.528	31.738	63.942	64.785	65.471	
	0.15	27.220	43.356	63.254	65.015	65.025	
	0.2	27.898	26.850	62.294	65.048	65.104	
	0.25	18.919	44.963	63.543	65.086	65.076	
	0.3	24.129	35.301	64.703	65.031	65.118	
	0.35	27.963	28.498	63.579	65.173	65.241	
	0.4	27.297	30.513	63.251	65.067	65.136	
	0.45	24.850	30.185	62.563	65.140	65.203	
	0.5	18.465	20.008	64.099	65.120	65.107	
2	0.05	18.469	52.433	63.268	64.967	65.026	
	0.1	22.220	53.827	63.266	65.429	65.275	
	0.15	23.528	31.738	63.942	64.785	65.471	
	0.2	20.402	62.999	63.268	65.443	65.507	
	0.25	16.737	38.443	63.255	65.453	65.322	
	0.3	29.263	59.764	63.313	65.475	65.493	
	0.35	20.763	65.548	63.254	65.251	65.430	
	0.4	21.740	55.698	63.285	65.292	65.372	
	0.45	19.222	64.753	63.129	65.241	65.411	
	0.5	21.795	63.692	63.251	65.425	65.474	
3	0.05	19.317	64.599	63.275	64.972	65.023	
	0.1	24.868	65.029	63.254	64.994	65.164	
	0.15	27.888	65.191	63.585	64.976	64.996	
	0.2	26.165	61.347	63.220	65.036	65.059	
	0.25	16.904	62.256	63.283	65.026	64.862	
	0.3	24.740	64.269	63.335	65.108	64.949	
	0.35	27.327	62.096	63.397	65.038	65.156	
	0.4	23.506	62.356	63.280	65.039	65.146	
	0.45	24.421	65.082	63.100	65.955	65.162	
	0.5	21.079	63.038	63.339	65.072	65.092	
4	0.05	19.575	65.120	63.175	65.042	65.152	
	0.1	23.145	64.408	63.055	65.057	65.108	
	0.15	22.713	65.088	63.068	65.074	65.074	
	0.2	20.707	65.124	63.132	65.008	65.031	
	0.25	26.124	64.903	63.177	65.117	65.004	
	0.3	21.662	65.061	63.153	65.033	65.145	
	0.35	24.778	65.028	63.178	65.086	65.079	
	0.4	23.672	63.115	65.088	65.090	65.100	
	0.45	22.034	65.080	63.949	65.116	65.004	
	0.5	23.328	65.076	65.077	65.114	64.998	

VII. INFERENCES FROM SIMULATION

In the experiment of simulating VRP across various noise channels, we vary the noise probability from 0.05 to 0.5 and observe the energy values of VQE. We execute the VRP circuit with 1–4 layers on our chosen optimizer COBYLA for both 6 and 12 qubit configurations. The experiment is repeated ten times for each noise model with different noise realizations. In the same experiment, we calculate the minimum eigenvalue of classical Hamiltonian and record the difference in energy cost after noise induction.

The state’s energy is recorded for each layer from 1 to 4 of the QAOA circuit. These values are averaged over the ten iterations to arrive at the average energy cost for each value of the noise parameter and each layer number. The results are given in Table 1. The deviation from the optimal value is shown in Table 2. Finally, Table 3 summarizes the inferences

on deviation from the classical minimum for various noise models.

We observe that VQE results are impacted due to the induction of noise. In the following sections, we will describe our observations briefly for each noise model.

A. AMPLITUDE DAMPING NOISE

Amplitude damping noise shows values range between 50% and 75% of classical minimum for both 6 qubit (−17.68) and 12 qubit (−65.684) circuits. There are a few outliers where the algorithm can reach very close to the classical minimum for 6 qubit circuits. For 12 qubit circuits, the values are mostly above 50% of classical minimum but never reach classical minimum as close as in 6 qubit circuits. This trend is seen across multiple layers for amplitude damping channels. It is noticed that the global minimum across layers is observed at the second layer at −48.947, but it is very close to the minimum of the first layer at −47.219. Hence, we

TABLE 3 Summary of the Inferences on Deviation From the Classical Minimum for VQE Simulation of VRP Using Amplitude Damping, Bit-Flip, Phase-Flip, Bit-Phase-Flip, and Depolarizing Channels for 6 and 12 Qubits

Noise channel	Qubits	Classical minimum	First Layer Dev.%	Second Layer Dev.%	Third Layer Dev. %	Fourth Layer Dev.%
Amplitude damping	6	-17.68	25%–50%	25%–50%	25%–50%	25%–50%
Bit flip	6	-17.68	50%	75%	Nearly 100%	Nearly 100%
Bit-phase flip	6	-17.68	Nearly 100%	Nearly 100%	Nearly 100%	Nearly 100%
Phase flip	6	-17.68	Nearly 100%	Nearly 100%	Nearly 100%	Nearly 100%
Depolarizing	6	-17.68	75%	Nearly 100%	Nearly 100%	Nearly 100%

Noise channel	Qubits	Classical minimum	1st Layer Dev. %	2nd Layer Dev.%	3rd Layer Dev. %	4th Layer Dev.%
Amplitude damping	12	-65.684	25%–50%	25%–50%	25%–50%	25%–50%
Phase flip	12	-65.684	50%	75%	Nearly 100%	Nearly 100%
Bit-phase flip	12	-65.684	Nearly 100%	Nearly 100%	Nearly 100%	Nearly 100%
Phase flip	12	-65.684	Nearly 100%	Nearly 100%	Nearly 100%	Nearly 100%
Depolarizing	12	-65.684	Nearly 100%	Nearly 100%	Nearly 100%	Nearly 100%

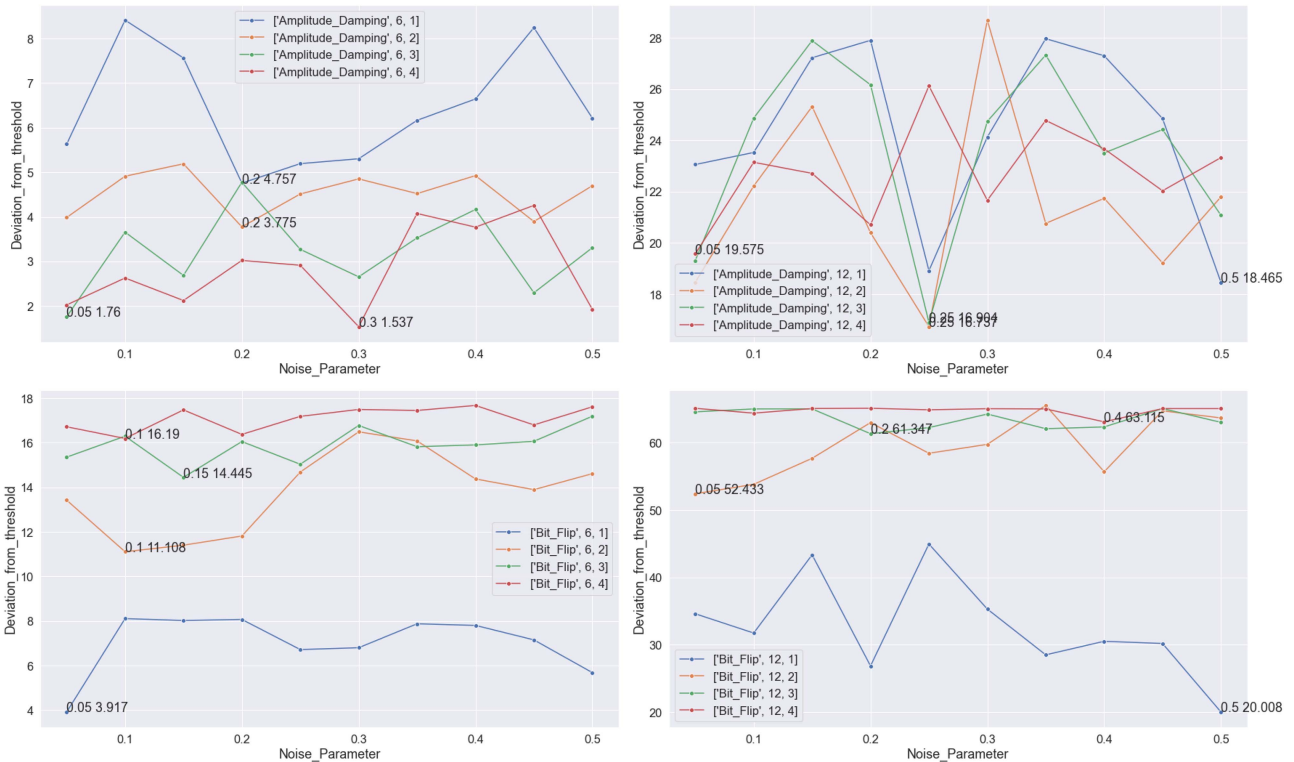


FIGURE 4. Plot illustrating the average deviation of the energy cost of VRP with four layers using various amplitude damping and bit-flip noise models. The plot consists of two charts depicting the simulation output of 6 and 12 qubit circuits, respectively. The average deviation at each layer is represented in pairs with (noise parameter, average deviation of energy cost) format.

can infer that increasing layers does not necessarily improve the results for the amplitude-damping channel. Table 6 in the Appendix summarizes the amplitude damping average energy values. Fig. 4 refers to the average deviation of energy cost from the classical minimum at each noise parameter across layers. This again confirms that deviation from classical minimum energy cost due to amplitude damping noise remains within 25%–50%, which is recorded in the Table 3.

B. BIT-FLIP NOISE

For the bit-flip noise channel, we note that the VQE values are 50% or above the classical minimum for the first layer,

but it degrades to around 25% for the second layer, falling further on third layer and finally close to zero on fourth layer. Since this trend is seen in both 6 and 12 qubit circuits, we can infer that increasing the number of layers degrades the VQE values for the bit-flip noise channel. We have summarized the bit-flip noise channel average energy values in Table 8 in the Appendix. Fig. 4 refers to the average deviation of energy cost from the classical minimum at each noise parameter across layers. This again confirms that deviation from classical minimum energy cost due to bit-flip noise remains within the range of 50%–75% for the first two layers before deteriorating further, which is recorded in Table 3.

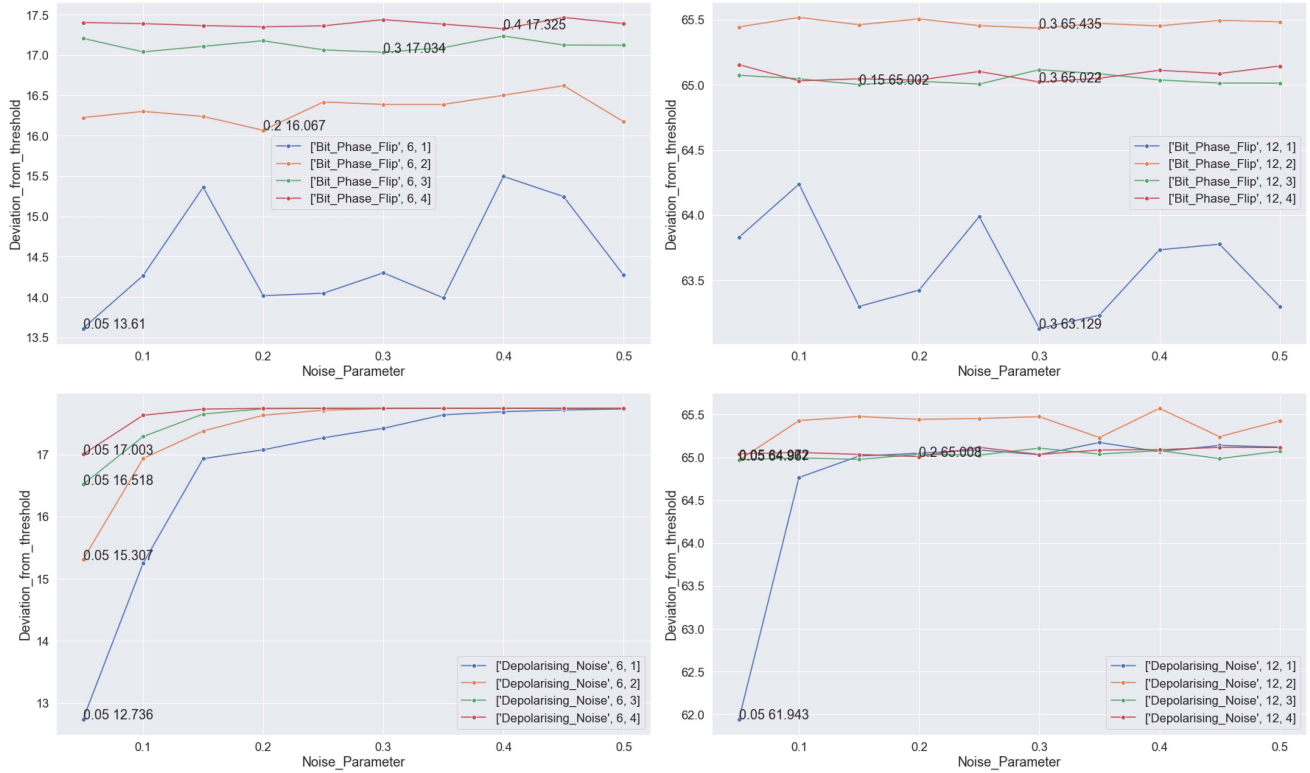


FIGURE 5. Plot illustrating the average deviation of the energy cost of VRP with four layers using bit-phase-flip and depolarizing noise. The plot consists of two charts depicting the simulation output of 6 and 12 qubit circuits, respectively. The average deviation at each layer is represented in pairs with (noise parameter, average deviation of energy cost) format.

C. BIT-PHASE-FLIP NOISE

There is a similar observation for the bit-phase-flip channel. The VQE values are 25% (or above) of the classical minimum for the first layer, but they degrade as the layers increase for 6 qubit. For 12 qubit circuits, the VQE values are consistently poor. We have summarized bit-phase-flip noise channel average energy values in Table 8 in the Appendix. Fig. 5 refers to the average deviation of energy cost from the classical minimum at each noise parameter across layers. This again confirms that deviation from classical minimum energy cost due to bit-phase-flip noise remains close to 100%; this is recorded in Table 3.

D. PHASE-FLIP AND DEPOLARIZING NOISE CHANNEL

Finally, for both depolarizing and phase-flip channels, the VQE values remain close to zero for both 6 and 12 qubit circuits. It appears that phase-flip and depolarizing noise channels are the most detrimental in VQE circuits (see Tables 10 and 14 in the Appendix). Figs. 5 and 6 refer to the average deviation of energy cost from the classical minimum at each noise parameter across layers. This again confirms that deviation from classical minimum energy cost due to depolarizing and phase-flip noise remains close to 100%, recorded in Table 3.

E. DATA GATHERING AND STATISTICS COLLECTION

In all the simulations, we have used a quantum instance object and a fixed random seed in the Qiskit framework to avoid VQE terminating early and mitigate statistical fluctuations. Hence, all the noise models used here are applied to the quantum instance object, which in turn applies noise to the circuit whose parameters are varied by VQE to arrive at a result. We have executed ten iterations of VQE using various noise channels described previously. From the results of the ten simulations, we have taken the average energy value of each noise parameter at each layer. Our figure of merit is the difference between the layer’s classical minimum and average energy cost. We remind the reader that gate-based simulations are extremely expensive. The results reported here amounted to 219 h of CPU time on a standard laptop computer using Qiskit’s built-in simulators [37]. While more iterations would improve the variability of the average energy calculations, some clear trends have already been observed.

VIII. CONCLUSION

Detailed simulation results, first for a noiseless case (see Figs. 2 and 3) and then for each noise channel, are provided in the Appendix. Each noise channel’s simulation results are divided into two tables: an average and a minimum. These tables make it easy to see that the deviation from the classical

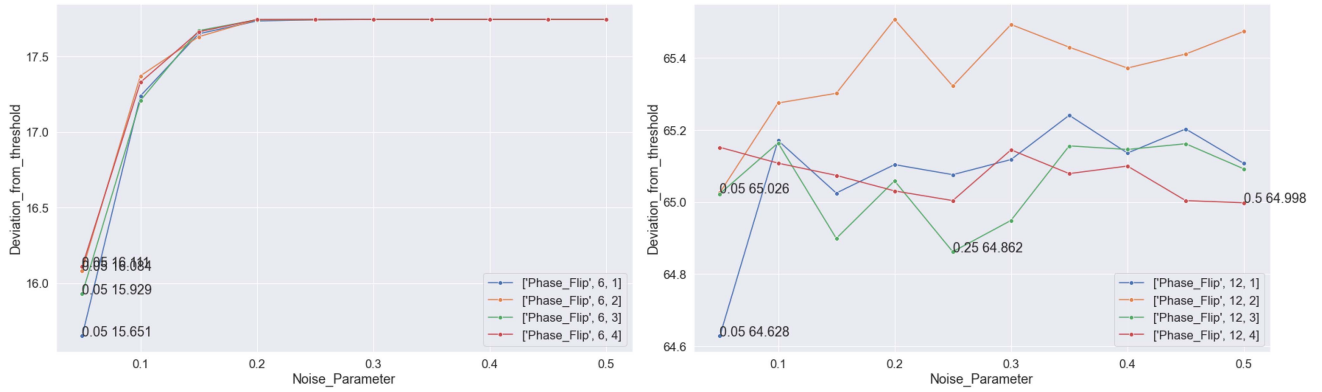


FIGURE 6. Plot illustrating the average energy cost of VRP with four layers using the phase-flip noise model. The plot consists of two charts depicting the simulation output of 6 and 12 qubit circuits, respectively. The average deviation at each layer is represented in pairs with (noise parameter, average deviation of energy cost) format.

minimum does not change linearly when the noise parameter is changed from 0.05 to 0.5 for each given noise channel. It is also clear that the deviation from the classical minimum does not grow linearly with the number of circuit layers used for optimization. Consider the instance of a bit-flip noise channel with a noise parameter of 0.25; for layers 1–3, the minimum energy cost is almost the classical minimum at -16.8 , but at fourth layer, it abruptly lowers to -1.9 . In the same scenario, the average energy continues a downward trend, going from 10.9 to 0.5. As we go over the tables for all of the noise channels, we observe several similar patterns. As a corollary, this validates making all noise channels use the same 0.05 to 0.5 broad noise parameter range. When we choose a lesser range or a range of noise parameters that do not ruin the results (different for each noise channel), we will see similar tendencies as we have found here, despite the fact that separate noise channels impact the circuit and the findings differently. We can see similar behaviors and range of noise parameters (though the case and experiment is different) for the application of noise channels in quantum teleportation [39].

The work we have presented in this article provides an interesting avenue for evaluating the effect of noise on detailed gate-based simulations of hybrid quantum algorithms for real-world applications. Noise is considered the most problematic aspect of today’s intermediate-scale devices, and hence, understanding the details of the effects of noise is critical in understanding how to make the most effective use of them.

In most cases, the effects of noise are minimal at the first layer of the VRP circuits. While additional layers improve upon the results in the noiseless case, the opposite is valid with the induction of noise. Since some noise will always be present in quantum circuits, an empirical finding of our results is that the COBYLA optimizer performs better for VQE circuits compared with the other available optimizers. Yet there is room to study other optimizers, such as SPSA. As we had come across prior work on noise simulations in quantum circuits, one clear trend is that the results are heavily

TABLE 4 Table Summarizing the Average Energy Cost (Out of 15 Runs) of VQE Simulation Using 5 Optimizers

Qubits	Layers	Average of energy_cost			
		COBYLA	L_BFGS_B	SLSQP	SPSA
6	1	-9.919	-3.367	-2.381	-8.753
	2	-10.985	-4.257	-3.692	-11.854
	3	-11.467	-3.792	-3.037	-12.850
	4	-13.711	-3.931	-5.420	-13.057
12	1	-30.920	-9.480	-5.941	-32.290
	2	-38.107	-12.802	-16.892	-37.090
	3	-38.461	-17.027	-18.203	-32.380
	4	-37.779	-18.312	-18.916	-32.460

TABLE 5 Table Containing the Minimum Energy Value (Out of 15 Runs) Across Each Layer of VQE Simulation Using 5 Optimizers

Qubits	Layers	Minimum of energy_cost			
		COBYLA	L_BFGS_B	SLSQP	SPSA
6	1	-16.776	-8.109	-6.240	-16.793
	2	-17.684	-8.033	-11.730	-17.589
	3	-17.438	-9.526	-7.461	-17.658
	4	-17.674	-7.655	-9.882	-17.651
12	1	-58.481	-39.316	-32.918	-62.181
	2	-65.227	-32.425	-35.286	-57.846
	3	-60.576	-27.507	-37.533	-55.635
	4	-60.704	-50.267	-51.096	-61.020

influenced by the optimizer used for the simulations, while some optimizers perform well at first for lower values of noise, as the noise probabilities increase the performance degrades; yet, among them, COBYLA performs well as seen by multiple experiments [22].

We also aim to test and compare these results on more significant VRP instances in physical devices, which is beyond the ability to simulate classically. Future work is also needed to analyze more detailed noise models guided by the measured characteristics from the real NISQ devices proposed to solve problems, such as VRP.

APPENDIX

Tables 4–15 show noiseless VQE simulation statistics.

TABLE 6 Table Containing the Average Energy Values (Out of 10 Runs) for Amplitude Damping Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Average Energy Cost (Amplitude Damping)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-12.053	-9.273	-10.121	-12.928	-12.491	-12.381	-11.519	-11.040	-9.440	-11.470
	2	-13.698	-12.773	-12.495	-13.909	-13.171	-12.834	-13.162	-12.760	-13.789	-12.986
	3	-15.924	-14.026	-14.999	-12.909	-14.413	-15.029	-14.246	-13.519	-15.388	-14.377
	4	-15.659	-15.055	-15.560	-14.661	-14.771	-16.147	-13.608	-13.916	-13.427	-15.757
12	1	-42.626	-42.157	-38.465	-37.787	-46.766	-41.555	-37.722	-38.388	-40.835	-47.219
	2	-47.215	-43.464	-40.368	-45.283	-48.947	-36.422	-44.922	-43.944	-46.463	-43.890
	3	-46.368	-40.817	-37.797	-39.519	-48.781	-40.945	-38.358	-42.179	-41.264	-44.605
	4	-46.110	-42.539	-42.971	-44.977	-39.561	-44.023	-40.907	-42.012	-43.650	-42.357

TABLE 7 Table Containing the Minimum Energy Values (Out of 10 Runs) for Amplitude Damping Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Minimum Energy Cost (Amplitude Damping)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-16.796	-16.796	-16.799	-16.799	-16.806	-16.790	-16.799	-16.799	-16.799	-16.796
	2	-17.684	-17.684	-17.684	-17.684	-17.684	-17.638	-17.684	-17.388	-17.648	-17.684
	3	-17.467	-17.349	-17.655	-17.497	-17.043	-17.642	-17.520	-16.780	-17.684	-17.642
	4	-17.546	-17.609	-17.684	-17.480	-17.178	-17.530	-17.138	-16.609	-16.076	-17.467
12	1	-62.027	-61.905	-61.971	-61.889	-61.964	-61.961	-61.882	-61.866	-61.961	-61.879
	2	-61.908	-61.375	-55.915	-64.155	-64.033	-53.234	-62.422	-61.047	-64.905	-55.145
	3	-65.491	-55.402	-48.300	-51.375	-65.320	-53.079	-48.096	-51.954	-65.586	-65.303
	4	-63.899	-50.359	-64.895	-53.408	-48.994	-53.405	-62.089	-51.583	-56.198	-51.556

TABLE 8 Table Containing the Average Energy Values (Out of 10 Runs) for BitFlip Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Average Energy Cost (Bit Flip)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-13.768	-9.577	-9.661	-9.614	-10.965	-10.883	-9.809	-9.881	-10.531	-11.999
	2	-4.262	-6.576	-6.282	-5.872	-3.002	-1.198	-1.603	-3.024	-3.790	-3.075
	3	-2.332	-1.392	-3.239	-1.634	-2.651	-0.910	-1.864	-1.784	-1.621	-0.498
	4	-0.969	-1.494	-0.213	-1.312	-0.502	-0.193	-0.238	-0.012	-0.886	-0.074
12	1	-31.086	-33.947	-22.328	-38.835	-20.722	-30.384	-37.187	-35.172	-35.500	-45.676
	2	-13.251	-11.857	-7.993	-2.685	-7.242	-5.921	-0.137	-9.986	-0.931	-1.993
	3	-1.086	-0.656	-0.649	-4.338	-3.428	-1.416	-3.589	-3.328	-0.603	-2.647
	4	-0.565	-1.276	-0.596	-0.560	-0.781	-0.624	-0.657	-2.570	-0.604	-0.608

TABLE 9 Table Containing the Minimum Energy Values (Out of 10 Runs) for BitFlip Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Minimum Energy Cost (Bit Flip)									
		Noise Parameter Energy									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-16.806	-16.783	-16.799	-16.783	-16.799	-16.799	-16.799	-16.793	-16.790	-16.799
	2	-8.424	-16.602	-11.780	-11.859	-16.793	-3.497	-11.786	-11.790	-10.842	-16.790
	3	-9.951	-5.599	-10.799	-9.681	-16.799	-2.658	-8.158	-15.346	-8.322	-3.109
	4	-4.089	-7.556	-1.013	-9.082	-1.898	-1.891	-2.020	-0.520	-3.635	-0.520
12	1	-62.556	-62.537	-62.547	-62.507	-56.500	-62.550	-62.566	-62.570	-62.533	-62.510
	2	-38.609	-41.921	-25.464	-25.559	-42.451	-50.056	-0.497	-56.892	-8.194	-9.020
	3	-5.632	-0.819	-0.786	-14.905	-26.730	-9.385	-24.846	-28.303	-0.727	-20.372
	4	-0.882	-6.806	-0.819	-0.895	-2.286	-0.809	-0.888	-21.122	-1.059	-0.938

TABLE 10 Table Containing the Average Energy Values (Out of 10 Runs) for PhaseFlip Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Average Energy Cost (Phase Flip)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-2.033	-0.447	-0.036	0.049	0.058	0.059	0.059	0.059	0.059	0.059
	2	-1.601	-0.313	-0.055	0.058	0.058	0.059	0.059	0.059	0.059	0.059
	3	-1.755	-0.476	-0.014	0.058	0.059	0.059	0.059	0.059	0.059	0.059
	4	-1.574	-0.357	-0.022	0.059	0.059	0.059	0.059	0.059	0.059	0.059
12	1	-1.056	-0.514	-0.660	-0.581	-0.609	-0.567	-0.444	-0.548	-0.482	-0.577
	2	-0.658	-0.410	-0.383	-0.178	-0.363	-0.192	-0.255	-0.313	-0.274	-0.211
	3	-0.662	-0.520	-0.786	-0.626	-0.823	-0.736	-0.529	-0.539	-0.523	-0.593
	4	-0.533	-0.577	-0.611	-0.653	-0.680	-0.539	-0.606	-0.585	-0.681	-0.686

TABLE 11 Table Containing the Minimum Energy Values (Out of 10 Runs) for PhaseFlip Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Minimum Energy Cost (Phase Flip)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-2.309	-0.493	-0.092	0.043	0.056	0.059	0.059	0.059	0.059	0.059
	2	-2.243	-0.602	-0.086	0.056	0.056	0.059	0.059	0.059	0.059	
	3	-2.234	-0.599	-0.089	0.056	0.056	0.059	0.059	0.059	0.059	
	4	-2.240	-0.609	-0.086	0.056	0.059	0.059	0.059	0.059	0.059	
12	1	-1.559	-0.539	-0.849	-0.664	-0.822	-0.717	-0.543	-0.589	-0.589	-0.645
	2	-1.135	-0.536	-0.516	-0.451	-0.510	-0.365	-0.385	-0.438	-0.576	-0.362
	3	-0.914	-0.602	-1.013	-0.855	-1.089	-0.921	-0.618	-0.641	-0.615	-0.763
	4	-0.914	-0.914	-0.711	-0.743	-0.849	-0.622	-0.717	-0.796	-0.987	-0.730

TABLE 12 Table Containing the Average Energy Values (Out of 10 Runs) for Bit-Phase-Flip Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Average Energy Cost (Bit-PhaseFlip)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-4.241	-2.571	-3.370	-5.296	-4.159	-3.756	-4.722	-3.519	-2.716	-3.104
	2	-1.668	-1.118	-1.313	-1.411	-1.621	-1.699	-1.468	-1.304	-1.236	-1.237
	3	-0.551	-0.605	-0.550	-0.492	-0.407	-0.600	-0.723	-0.574	-0.620	-0.693
	4	-0.411	-0.446	-0.343	-0.101	-0.219	-0.368	-0.440	-0.439	-0.241	-0.221
12	1	-1.486	-1.713	-2.390	-3.355	-2.124	-0.978	-2.046	-2.405	-3.078	-1.565
	2	-0.417	-0.419	-0.299	-0.417	-0.358	-0.372	-0.431	-0.400	-0.555	-0.433
	3	-0.409	-0.460	-0.493	-0.464	-0.401	-0.349	-0.289	-0.446	-0.584	-0.345
	4	-0.510	-0.630	-0.617	-0.553	-0.508	-0.499	-0.526	-0.596	-0.736	-0.607

TABLE 13 Table Containing the Minimum Energy Values (Out of 10 Runs) for Bit-PhaseFlip Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Minimum Energy Cost (Bit-Phase-Flip)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-5.355	-5.069	-5.378	-5.368	-5.352	-5.378	-5.355	-5.365	-5.378	-5.378
	2	-1.819	-1.845	-1.839	-1.852	-1.799	-1.763	-1.826	-1.770	-1.780	-1.855
	3	-0.776	-0.776	-0.799	-0.766	-0.556	-0.793	-0.796	-0.724	-0.783	-0.793
	4	-0.503	-0.533	-0.447	-0.438	-0.447	-0.530	-0.507	-0.516	-0.431	-0.533
12	1	-4.171	-4.332	-4.197	-4.523	-4.352	-2.632	-4.523	-4.516	-4.516	-4.033
	2	-0.638	-0.510	-0.368	-0.470	-0.447	-0.500	-0.609	-0.477	-0.780	-0.539
	3	-0.497	-0.717	-0.711	-0.714	-0.609	-0.408	-0.477	-0.599	-0.776	-0.582
	4	-0.737	-0.691	-0.829	-0.796	-0.609	-0.727	-0.720	-0.757	-0.885	-0.832

TABLE 14 Table Containing the Average Energy Values (Out of 10 Runs) for Depolarising Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Average Energy Cost (Depolarising)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-4.949	-2.437	-0.750	-0.608	-0.416	-0.261	-0.045	0.006	0.034	0.050
	2	-2.377	-0.748	-0.305	-0.052	0.029	0.055	0.058	0.059	0.059	0.059
	3	-1.167	-0.396	-0.031	0.053	0.058	0.059	0.059	0.059	0.059	0.059
	4	-0.681	-0.052	0.047	0.059	0.059	0.059	0.059	0.059	0.059	0.059
12	1	-3.741	-0.919	-0.669	-0.637	-0.599	-0.653	-0.512	-0.618	-0.545	-0.565
	2	-0.718	-0.255	-0.208	-0.242	-0.231	-0.209	-0.454	-0.113	-0.444	-0.259
	3	-0.713	-0.690	-0.709	-0.649	-0.658	-0.576	-0.646	-0.605	-0.700	-0.613
	4	-0.643	-0.628	-0.650	-0.676	-0.567	-0.651	-0.599	-0.594	-0.569	-0.571

TABLE 15 Table Containing the Minimum Energy Values (Out of 10 Runs) for Depolarising Noise Channel Across Each Layer of VQE Simulation Using COBYLA

Qubits	Layers	Minimum Energy Cost (Depolarising)									
		Noise Parameter									
		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
6	1	-7.632	-3.464	-1.651	-0.701	-0.530	-0.336	-0.102	-0.059	0.013	0.039
	2	-3.507	-1.020	-0.516	-0.089	-0.010	0.046	0.056	0.059	0.059	0.059
	3	-1.822	-0.533	-0.086	0.039	0.056	0.059	0.059	0.059	0.059	0.059
	4	-1.105	-0.115	0.030	0.056	0.059	0.059	0.059	0.059	0.059	0.059
12	1	-7.888	-1.447	-0.783	-1.000	-0.750	-0.763	-0.579	-0.793	-0.655	-0.826
	2	-1.849	-0.421	-0.461	-0.582	-0.391	-0.477	-0.855	-0.359	-0.628	-0.589
	3	-1.102	-1.039	-0.819	-0.803	-0.987	-0.743	-0.934	-0.862	-0.931	-0.793
	4	-0.862	-0.977	-0.829	-0.964	-0.750	-0.954	-0.813	-0.760	-0.697	-0.796

ACKNOWLEDGMENT

The authors would like to thank the IBM Quantum Experience platform and their team for developing the Qiskit platform and providing open access to their simulators for running quantum circuits and performing the experiments reported here [37].

REFERENCES

- [1] A. Montanaro, "Quantum algorithms: An overview," *npj Quantum Inf.*, vol. 2, no. 1, Jan. 2016, Art. no. 15023, doi: 10.1038/npjqi.2015.23.
- [2] S. Jordan, "Optimization, numerics, and machine learning." Accessed: Aug. 24, 2023. [Online]. Available: <https://quantumalgorithmzoo.org/#ONML>

- [3] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," 2014, *arXiv:1411.4028*, doi: [10.48550/arXiv.1411.4028](https://doi.org/10.48550/arXiv.1411.4028).
- [4] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, "Quantum computation by adiabatic evolution," 2000, *arXiv:quant-ph/0001106*, doi: [10.48550/arXiv.quant-ph/0001106](https://doi.org/10.48550/arXiv.quant-ph/0001106).
- [5] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proc. 28th annual ACM Symp. Theory Comput.*, 1996, pp. 212–219, doi: [10.1145/237814.237866](https://doi.org/10.1145/237814.237866).
- [6] V. Dasari, M. S. Im, and L. Beshaj, "Solving machine learning optimization problems using quantum computers," *Proc. SPIE*, vol. 11419, Apr. 2020, Art. no. 114190F, doi: [10.1117/12.2565038](https://doi.org/10.1117/12.2565038).
- [7] National Academies of Sciences, Engineering, and Medicine, "Quantum algorithms and applications," in *Quantum Computing: Progress and Prospects*. Washington, DC, USA: Nat. Acad. Press, 2019, ch. 3, pp. 57–94, doi: [10.17226/25196](https://doi.org/10.17226/25196).
- [8] S. Harwood, C. Gambella, D. Tenev, A. Simonetto, D. Bernal, and D. Greenberg, "Formulating and solving routing problems on quantum computers," *IEEE Trans. Quantum Eng.*, vol. 2, 2021, Art. no. 3100118, doi: [10.1109/TQE.2021.3049230](https://doi.org/10.1109/TQE.2021.3049230).
- [9] K. Srinivasan, S. Satyajit, B. K. Behera, and P. K. Panigrahi, "Efficient quantum algorithm for solving travelling salesman problem: An IBM quantum experience," May 2018, *arXiv:1805.10928v1*, doi: [10.48550/arXiv.1805.10928](https://doi.org/10.48550/arXiv.1805.10928).
- [10] S. Feld et al., "A hybrid solution method for the capacitated vehicle routing problem using a quantum annealer," *Front. ICT*, vol. 6, 2019, Art. no. 13, doi: [10.3389/fict.2019.00013](https://doi.org/10.3389/fict.2019.00013).
- [11] M. Nazari, A. Orojlooy, L. Snyder, and M. Takac, "Reinforcement learning for solving the vehicle routing problem," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 9861–9871, doi: [10.5555/3327546.3327651](https://doi.org/10.5555/3327546.3327651).
- [12] U. Azad, B. K. Behera, E. A. Ahmed, P. K. Panigrahi, and A. Farouk, "Solving vehicle routing problem using quantum approximate optimization algorithm," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 7, pp. 7564–7573, Jul. 2023, doi: [10.1109/TITS.2022.3172241](https://doi.org/10.1109/TITS.2022.3172241).
- [13] C. Papalitsas, T. Andronikos, K. Giannakis, G. Theocharopoulou, and S. Fanarioti, "A QUBO model for the traveling salesman problem with time windows," *Algorithms*, vol. 12, no. 11, 2019, Art. no. 224, doi: [10.3390/a12110224](https://doi.org/10.3390/a12110224).
- [14] F. Glover, G. Kochenberger, M. Ma, and Y. Du, "Quantum bridge analytics II: QUBO-Plus, network optimization and combinatorial chaining for asset exchange," *4OR*, vol. 18, no. 4, pp. 387–417, 2020, doi: [10.1007/s10288-020-00464-9](https://doi.org/10.1007/s10288-020-00464-9).
- [15] G. Kochenberger et al., "The unconstrained binary quadratic programming problem: A survey," *J. Combinatorial Optim.*, vol. 28, pp. 58–81, Jul. 2014, doi: [10.1007/s10878-014-9734-0](https://doi.org/10.1007/s10878-014-9734-0).
- [16] H. Irie, G. Wongpaisarnsin, M. Terabe, A. Miki, and S. Taguchi, "Quantum annealing of vehicle routing problem with time, state and capacity," in *Quantum Technology and Optimization Problems (Lecture Notes in Computer Science)*, S. Feld and C. Linnhoff-Popien, Eds., Cham, Switzerland: Springer Int. Publishing, 2019, pp. 145–156, doi: [10.1007/978-3-030-14082-3_13](https://doi.org/10.1007/978-3-030-14082-3_13).
- [17] A. Crispin and A. Syrichas, "Quantum annealing algorithm for vehicle scheduling," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2013, pp. 3523–3528, doi: [10.1109/SMC.2013.601](https://doi.org/10.1109/SMC.2013.601).
- [18] M. Sao, H. Watanabe, Y. Musha, and A. Utsunomiya, "Application of digital annealer for faster combinatorial optimization," *FUJITSU Sci. Tech. J.*, vol. 55, no. 2, pp. 45–51, 2019. [Online]. Available: <https://www.fujitsu.com/global/documents/about/resources/publications/fstj/archives/vol55-2/paper12.pdf>
- [19] E. Campos, D. Rabinovich, V. Akshay, and J. Biamonte, "Training saturation in layerwise quantum approximate optimization," *Phys. Rev. A*, vol. 104, no. 3, Sep. 2021, Art. no. L030401, doi: [10.1103/PhysRevA.104.L030401](https://doi.org/10.1103/PhysRevA.104.L030401).
- [20] S. Wang et al., "Noise-induced barren plateaus in variational quantum algorithms," *Nature Commun.*, vol. 12, no. 1, Nov. 2021, Art. no. 6961, doi: [10.1038/s41467-021-27045-6](https://doi.org/10.1038/s41467-021-27045-6).
- [21] J. Marshall, F. Wudarski, S. Hadfield, and T. Hogg, "Characterizing local noise in QAOA circuits," *IOP SciNotes*, vol. 1, no. 2, Aug. 2020, Art. no. 025208, doi: [10.1088/2633-1357/abb0d7](https://doi.org/10.1088/2633-1357/abb0d7).
- [22] W. Lavrijsen, A. Tudor, J. Müller, C. Iancu, and W. de Jong, "Classical optimizers for noisy intermediate-scale quantum devices," in *Proc. IEEE Int. Conf. Quantum Comput. Eng.*, Oct. 2020, pp. 267–277, doi: [10.1109/QCE49297.2020.00041](https://doi.org/10.1109/QCE49297.2020.00041).
- [23] M. Cerezo et al., "Variational quantum algorithms," *Nature Rev. Phys.*, vol. 3, no. 9, pp. 625–644, Sep. 2021, doi: [10.1038/s42254-021-00348-9](https://doi.org/10.1038/s42254-021-00348-9).
- [24] N. Guerrero, "Solving combinatorial optimization problems using the quantum approximation optimization algorithm," Mater's ThesisDept. Eng. Phys., Wright-Patterson AFB, OH, USA, Mar. 2020. [Online]. Available: <https://scholar.afit.edu/etd/3263>
- [25] T. Albash and D. A. Lidar, "Adiabatic quantum computation," *Rev. Mod. Phys.*, vol. 90, no. 1, Jan. 2018, Art. no. 015002, doi: [10.1103/RevModPhys.90.015002](https://doi.org/10.1103/RevModPhys.90.015002).
- [26] E. K. Grant and T. S. Humble, "Adiabatic quantum computing and quantum annealing," in *Oxford Research Encyclopedia of Physics*. New York, NY, USA: Oxford Univ. Press, 2020, doi: [10.1093/acrefore/9780190871994.013.32](https://doi.org/10.1093/acrefore/9780190871994.013.32).
- [27] R. Hoque, "The quantum approximate optimization algorithm," 2019. Accessed: Aug. 24, 2023. [Online]. Available: <https://ryanhoque.github.io/data/QAOA.pdf>
- [28] Y. Sun, J.-Y. Zhang, M. S. Byrd, and L.-A. Wu, "Adiabatic quantum simulation using trotterization," *New J. Phys.*, vol. 22, no. 5, May 2020, Art. no. 053012, doi: [10.1088/1367-2630/ab7a31](https://doi.org/10.1088/1367-2630/ab7a31).
- [29] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices," *Phys. Rev. X*, vol. 10, no. 2, Jun. 2020, Art. no. 021067, doi: [10.1103/PhysRevX.10.021067](https://doi.org/10.1103/PhysRevX.10.021067).
- [30] S. P. Singh, "The ising model: Brief introduction and its application," in *Metastable, Spintronics Materials and Mechanics of Deformable Bodies*. London, U.K.: IntechOpen, Feb. 2020, ch. 8, Art. no. 155201, doi: [10.5772/intechopen.90875](https://doi.org/10.5772/intechopen.90875). [Online]. Available: <https://www.intechopen.com/chapters/71210>
- [31] S. G. Brush, "History of the Lenz-Ising model," *Rev. Mod. Phys.*, vol. 39, pp. 883–893, Oct. 1967, doi: [10.1103/RevModPhys.39.883](https://doi.org/10.1103/RevModPhys.39.883).
- [32] A. Lucas, "Ising formulations of many NP problems," *Front. Phys.*, vol. 2, 2014, Art. no. 5, doi: [10.3389/fphy.2014.00005](https://doi.org/10.3389/fphy.2014.00005).
- [33] A. Peruzzo et al., "A variational eigenvalue solver on a photonic quantum processor," *Nature Commun.*, vol. 5, no. 1, Jul. 2014, Art. no. 4213, doi: [10.1038/ncomms5213](https://doi.org/10.1038/ncomms5213).
- [34] "Qiskit: Vehicle routing." Accessed: Aug. 24, 2023. [Online]. Available: https://qiskit.org/ecosystem/optimization/tutorials/07_examples_vehicle_routing.html
- [35] P. Date, R. Patton, C. Schuman, and T. Potok, "Efficiently embedding QUBO problems on adiabatic quantum computers," *Quantum Inf. Process.*, vol. 18, no. 4, Mar. 2019, Art. no. 117, doi: [10.1007/s11128-019-2236-3](https://doi.org/10.1007/s11128-019-2236-3).
- [36] G. G. Guerreschi, "Solving quadratic unconstrained binary optimization with divide-and-conquer and quantum algorithms," Jan. 2021, *arXiv:2101.07813*, doi: [10.48550/arXiv.2101.07813](https://doi.org/10.48550/arXiv.2101.07813).
- [37] M. S. A. et al., "Qiskit: An open-source framework for quantum computing," 2021. Accessed: Aug. 24, 2023. [Online]. Available: <https://github.com/Qiskit/qiskit/tree/0.25.0>
- [38] S. Ahadpour, F. Mirmasoudi, S. Ahadpour, and F. Mirmasoudi, "The role of noisy channels in quantum teleportation," *Revista Mexicana de Física*, vol. 66, no. 3, pp. 378–387, Jun. 2020, doi: [10.31349/revmexfis.66.378](https://doi.org/10.31349/revmexfis.66.378). [Online]. Available: http://www.scielo.org.mx/scielo.php?script=sci_abstract&pid=S0035-001X2020000300378&lng=es&nrm=iso&tlng=en
- [39] R.-G. Zhou, Y.-N. Zhang, R. Xu, C. Qian, and I. Hou, "Asymmetric bidirectional controlled teleportation by using nine-qubit entangled state in noisy environment," *IEEE Access*, vol. 7, pp. 75247–75264, 2019, doi: [10.1109/ACCESS.2019.2920094](https://doi.org/10.1109/ACCESS.2019.2920094). [Online]. Available: <https://ieeexplore.ieee.org/document/87266306>