Quantum Approximate Optimization Algorithm Based Maximum Likelihood Detection

Jingjing Cui, *Member, IEEE,* Yifeng Xiong, Soon Xin Ng, *Senior Member, IEEE* and Lajos Hanzo, *Fellow, IEEE*

***Abstract*—Recent advances in quantum technologies pave the way for noisy intermediate-scale quantum (NISQ) devices, where the quantum approximation optimization algorithm (QAOA) constitutes a promising candidate for demonstrating tangible quantum advantages based on NISQ devices. In this paper, we consider the maximum likelihood (ML) detection problem of binary symbols transmitted over a multiple-input and multiple-output (MIMO) channel, where finding the optimal solution is exponentially hard using classical computers. Here, we apply the QAOA for the ML detection by encoding the problem of interest into a level-p QAOA circuit having $2p$ variational parameters, which can be optimized by classical optimizers. This level-p QAOA circuit is constructed by applying the prepared Hamiltonian to our problem and the initial Hamiltonian alternately in p consecutive rounds. More explicitly, we first encode the optimal solution of the ML detection problem into the ground state of a problem Hamiltonian. Using the quantum adiabatic evolution technique, we provide both analytical and numerical results for characterizing the evolution of the eigenvalues of the quantum system used for ML detection. Then, for level-1 QAOA circuits, we derive the analytical expressions of the expectation values of the QAOA and discuss the complexity of the QAOA based ML detector. Explicitly, we evaluate the computational complexity of the classical optimizer used and the storage requirement of simulating the QAOA. Finally, we evaluate the bit error rate (BER) of the QAOA based ML detector and compare it both to the classical ML detector and to the classical minimum mean squared error (MMSE) detector, demonstrating that the QAOA based ML detector is capable of approaching the performance of the classical ML detector.**

***Index Terms*—Quantum technology, maximum likelihood (ML) detection, quantum approximation optimization algorithm (QAOA), bit error rate (BER).**

I. Introduction

The evolution of social networking and the ubiquitous wireless connectivity, coupled with the availability of low-cost yet powerful computing devices jointly shape the next generation of wireless systems. Integrated ground-air-space (IGAS) networks [1] together with new enabling technologies such as large-scale antenna arrays [2], reconfigurable intelligent surfaces [3] and Terahertz communications [4] constitute compelling solutions for the emerging services and applications. Powerful detection schemes play a pivotal role in supporting these novel techniques for achieving their potential gains. Maximum-likelihood (ML) detection is capable of providing the optimal solution by minimizing the probability of error, but it is NP-hard [5], because its complexity of finding the exact ML solution grows exponentially with the size of the constellations and the number of the symbols transmitted, which constitutes a challenge for classical computers. The properties of quantum mechanics such as superposition, entanglement and coherence have been beneficially exploited in the wireless communication field in terms of quantum information science [6] and quantum teleportation [7]. Given the intrinsic parallelism of quantum mechanics, it is promising to investigate the potential of quantum algorithms for solving ML detection problems.

Specifically, quantum computation exhibits advantages in solving some problems that require searching through a large space [8], benefiting from the nature of quantum mechanics. The most canonical examples are Schor's algorithm [9] designed for discrete Logarithms and factoring at an exponential speedup over classical methods as well as Grover's algorithm [10] conceived for unstructured search, which shows the potential of quadratic speed-up over the classical search. Hence, some applications of Grover's search inspired quantum algorithms to multi-carrier interleave-division multiple-access (MC-IDMA) systems and to Pareto optimal routing for wireless multihop networks can be found in [11], [12]. The benefits of these algorithms arise from the assumption of a universal fault-tolerant quantum computer, which is capable of providing error-free operation. However, a powerful system supporting millions of physical qubits and high-fidelity long sequences of gate operations is not available at the time of writing, but fortunately, noisy intermediate-scale quantum (NISQ) devices are likely to become available in the near future [13]. The state-of-the-art quantum device size ranges from 50 to 100 qubits. For instance, in 2020 IBM has built a quantum device processing 65 qubits [14]. Therefore, the near-term quantum computers will contain a limited number of quantum gates due to gate errors and decoherence [15], [16]. A family of hybrid quantum-classical algorithms, namely variational quantum eigensolvers (VQE), was developed in [17], which was implemented by combining a reconfigurable quantum device with a classical computer. The goal of hybrid quantum-classical algorithms is to take advantage of the potential of NISQ devices by incorporating partial computational resources of classical computers, hence it becomes one of leading candidate algorithms for establishing quantum advantages in near-term quantum computers [13], [18], [19]. The quantum approximation optimization algorithm (QAOA) [20], [21] constitute a hybrid quantum-classical algorithm designed based on the variational principle for solving combinatorial optimization problems on gate-based quantum computers. Indeed, this has become an active field of research due to its promising potential of being implemented by near-term quantum computers [22]–[24] as well as owing to its universality in quantum computation [25], [26].

J. Cui, Y. Xiong, S. Ng and L. Hanzo are with the School of Electronics and Computer Science, University of Southampton, SO17 1BJ, Southampton (UK) (e-mail: {jingj.cui, Yifeng.Xiong}@soton.ac.uk and {sxn,lh}@ecs.soton.ac.uk).

This work would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council projects EP/P034284/1 and EP/P003990/1 (COALESCE) as well as of the European Research Council's Advanced Fellow Grant QuantCom (Grant No. 789028)

The basic idea of the QAOA is to alternately apply the problem Hamiltonian, whose ground state encodes the solution of the problem considered, and the initial Hamiltonian. The QAOA relies on the combination of preparing the parameterized quantum circuit on the NISQ devices, and a classical optimizer that is used for finding the optimal parameters. The QAOA was first proposed in [20] for solving combinatorial optimization problems, where the paradigm and analysis on the QAOA were treated in terms of the max-cut problem. As a further advance, the capability of the QAOA to solve constrained combinatorial optimization problems was studied in [27], where the goal is to maximize the sum of a series of bounded linear equations. Then, the performance of the QAOA was analysed in the context of the MAX-kXOR and MAX-kSAT problems in [28]. The application of the QAOA to the channel decoding of binary codes was presented in [29] by means of the Ising Hamiltonian to Boolean constraint satisfaction problems. Furthermore, inspired by the adiabatic evolution principle, a learning method was proposed in [30] for optimizing parameters aiming at achieving a high overlap to the ground state of the problem Hamiltonian. Given the high flexibility of the QAOA, it was extended in [31] to solve Grover's unstructured search problem by replacing Grover's diffusion operator with the transverse field, which only requires single-qubit gates. In addition, the authors in [32] proposed a framework for designing the QAOA circuits for solving a couple of combinatorial problems subject to both hard and soft constraints, such as the graph coloring optimization problem and the traveling salesman problem (TSP). As a futher development, the QAOA was generalized as one of the standalone ansatz approaches in [22], termed as the quantum alternating operator ansatz. Recall that the quality of the solution produced by the QAOA for a specific problem of interest depends on the quality of the variational parameters found by the classical optimizer. Developing efficient QAOA parameter optimization approaches is therefore of pivotal importance for achieving quantum advantage. Diverse techniques have been conceived for optimizing the QAOA parameters such as gradient-based methods [33]–[35] and gradient-free methods [30], [36]–[38]. As for the max-cut problem in a general graph, the analytical expression for a level-1 QAOA was provided for guiding the associated parameter selections in [33]. Furthermore, analytical expressions were also derived for the QAOA haivng an arbitrary number of levels for a class of special max-cut graph instances in [33]. Moreover, a number of methods based on neural networks and reinforcement learning have been applied for optimizing the QAOA parameters [30], [35]–[38], concerning different max-cut graph instances.

Given the high flexibility of the QAOA in terms of handling various coherence times and gate requirements etc, there has been a growing interest in exploring the advantages of the QAOA in solving numerous practical problems of diverse fields. In this paper, we discuss how to apply the QAOA for solving ML detection problems that are NP-hard in classical computers. The basic procedure of the QAOA is to alternately apply the problem Hamiltonian and the initial Hamiltonian – also known as mixing Hamiltonian [22] or driver Hamiltonian [39] – to the initial state of the quantum system. The fundamental principle of the QAOA relys on gradually driving the quantum system to its ground state based on the quantum adiabatic evolution technique of [40]. Explicitly, the evolution of a quantum system is governed by the Schrödinger equation, and the adiabatic theorem tells us how to track this evolution, when the system changes sufficiently slowly. To elaborate a little further, the quantum adiabatic algorithm starts from an initial Hamiltonian whose ground state is easy to prepare, and evolves to a final Hamiltonian whose ground state encodes the solution of the problem considered. For implementing the QAOA, the evolution is encoded into a series of unitary quantum logic gates. Therefore, the goal of this paper is to illustrate the applicability of the QAOA to the family of ML detection problems and hence to open up new avenues of optimization in wireless communications. We commence with a brief overview of the applications of the QAOA in solving different problem instances, as seen in **Table I**. In this paper we present the first results on the QAOA applied to a real-world ML detection problem in wireless communications. Our main contributions are summarized as follows.

1) We encode the ML detection problem of binary symbols transmitted over a multiple-input and multiple-output (MIMO) channel into a Hamiltonian operator in the Ising transverse field [41]. Specifically, we use a qubit to encode a single binary symbol. Hence the total number of qubits to be processed by the QAOA is equal to the number of parallel data symbols transmitted by multiple antennas. Correspondingly, any legitimate solution of the ML detection problem can be represented by multiple-qubit tuple and the optimal solution is encoded into the ground state of the problem Hamiltonian.
2) For illustrating the fundamental principles of the QAOA, we provide both analytical and numerical results concerning the evolution of the eigenvalues in terms of the ML detection problem using a time-dependent Hamiltonian, which is a linear interpolation between the initial Hamiltonian and the problem Hamiltonian.
3) We transform the ML detection problem into a p-level QAOA circuit, where in each level the problem Hamiltonian that encodes the objective function of the ML detection problem is applied to the initial state of the quantum system followed by the initial Hamiltonian. The $2p$ times Hamiltonian operators involve the variational parameters, which are optimized classically for best performance. In particular, we derive the analytical expression of the ML detection problem for the level-1 QAOA, which simplifies the numerical optimization used for finding the optimal values of the parameters.
4) We provide numerical results for characterizing the expectation values of the QAOA solution of the ML detection problem, which illustrates that the energy landscape of the QAOA is nonconvex and has locally optimal points. Furthermore, the performance of the QAOA based ML detector is compared to that of both the classical ML detector as well as to the classical minimum mean squared error (MMSE) detector.

Table I: Problem instances solved by the QAOA

	[20] -2014	[27]-2015	[28]-2016	[31]-2017	[32]-2017	[33]-2018	[35]-2018	[36]-2019	[22]-2019	[42]-2020	[24]-2021	This work
Max-Cut	√	√	√		√	√	√	√	√		√	
MIS									√			
Graph Coloring					√				√	√		
TSP									√			
Unstructured search				√								
ML detection												√

Organization: The rest of our paper is organized as follows. Section II presents the problem model of the ML detection, followed by modelling its quantum Hamiltonians in Section III. Section IV introduces the quantum adiabatic evolution for characterizing the ML detection based quantum system. Then, Section V discusses the procedure of the QAOA harnessed for solving the ML detection problem. The computational complexity analysis of the QAOA is provided in Section VI. Finally, Section VII presents our simulation results and discussions, followed by our conclusions in Section VIII.

II. System Model

Consider a MIMO system and transmitting binary symbols over an $M_r \times M_t$ channel matrix, having M_t transmit antennas (TA) as well as M_r receive antennas (RA). Let $\mathbf{H} \in \mathcal{R}^{M_r \times M_t}$ be the channel matrix from the transmitter to the receiver, which is assumed to be known to the receiver. Therefore, the received signal can be expressed as

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \tag{1}$$

where $\mathbf{y} \in \mathcal{R}^{M_r}$ and $\mathbf{s} \in \mathcal{R}^{M_t}$ are the vectors of the received and the transmitted signals, respectively, while $\mathbf{n} \sim \mathcal{N}(0, \mathbf{I}_{M_r})$ denotes the noise vector.

The maximum likelihood (ML) detector can be formulated as

$$\min_{\mathbf{s} \in \mathcal{X}^{M_t}} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2, \tag{2}$$

where $\mathcal{X}$ represents the signal constellation of the binary symbols. Explicitly, problem (2) can be physically interpreted as follows: The ML detector is to find a vector $\mathbf{s}$ producing a vector $\mathbf{H}\mathbf{s}$, which is closest to the received signal $\mathbf{y}$. Correspondingly, problem (2) can be reformulated as the following optimization form:

$$\min_{\mathbf{s} \in \mathcal{X}^{M_t}} \mathbf{s}^T\mathbf{H}^H\mathbf{H}\mathbf{s} - 2\mathbf{y}^T\mathbf{H}\mathbf{s} + \mathbf{y}^T\mathbf{y}, \tag{3}$$

where the superscript T denotes the transpose of a matrix. Note that the ML detection problem is constituted by the combinatorial optimization problem of selecting the most likely transmit symbol from a discrete set, and which was shown to be NP-hard in [5]. The computational complexity of the ML detection problem for BPSK is an exponential function of the number of the transmit antennas M_t, formulated as in 2^{M_t}.

Therefore, the objective function of problem (3) can be expressed for BPSK having $\mathcal{X} = \{-1, +1\}$ as

$$\begin{aligned} f(\mathbf{s}) &= \mathbf{s}^T\mathbf{H}^T\mathbf{H}\mathbf{s} - 2\mathbf{y}^T\mathbf{H}\mathbf{s} + \mathbf{y}^T\mathbf{y} \\ &= \sum_{k,l=1}^{N} A_{k,l}s_k s_l - \sum_{k=1}^{N} 2b_k s_k + c \\ &= \sum_{l>k}^{N} 2A_{k,l}s_k s_l - \sum_{k=1}^{N} 2b_k s_k + c + \sum_{k=1}^{N} A_{k,k}, \end{aligned} \tag{4}$$

where $N = M_t$, $\mathbf{A} = \mathbf{H}^T\mathbf{H}$, $\mathbf{b} = \mathbf{y}^T\mathbf{H}$ and $c = \mathbf{y}^T\mathbf{y}$. Consequently, $A_{k,l}$ and b_k are the (k,l)-th element and the k-th element in $\mathbf{A}$ and $\mathbf{b}$, respectively. The last step in (4) comes from the fact that A is Hermitian, i.e., $A_{k,l} = A_{l,k}$, $\forall k, l$.

III. Quantum Hamiltonians and the Adiabatic Theorem

A. Hamiltonian H_f of the ML Problem

Since binary symbols are considered, we can use a single qubit to encode the values of a single symbol transmitted. Hence the total number of qubits required for quantum computation is equal to the number of parallel data symbols transmitted. More explicitly, an example is provided in Fig. 1 for demonstrating the connections between the different symbols received, where ↑ and ↓ represent the two legitimate states of BPSK systems, respectively. However, in quantum computing one qubit can be represented by a superposition of the two states. Hence, given an unknown binary symbol s_k, $k = 1, \cdots, N$, the associated quantum state $|z_k\rangle$ can be expressed as $|z_k\rangle = \alpha_k|\uparrow\rangle + \beta_k|\downarrow\rangle, \alpha_k, \beta_k \in \mathcal{C}$.

In order to transform the ML detection problem from a classical computation to quantum computation, we map (4) to its spin Hamiltonian. Explicitly, to arrive at the Hamiltonian of (4), we define $z_k \in \{0, 1\}$ as a spin-$\frac{1}{2}$ qubit associated with

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \text{ and } |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{5}$$

Then, we have

$$\sigma_z^{(k)}|z_k\rangle = \pm 1|z_k\rangle, \text{ and } \sigma_z^{(k)} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{6}$$

Note that $\sigma_z^{(k)}$ denotes the Pauli-Z operator acting on the k-th qubit. We can see that the binary symbol s_k is mapped onto

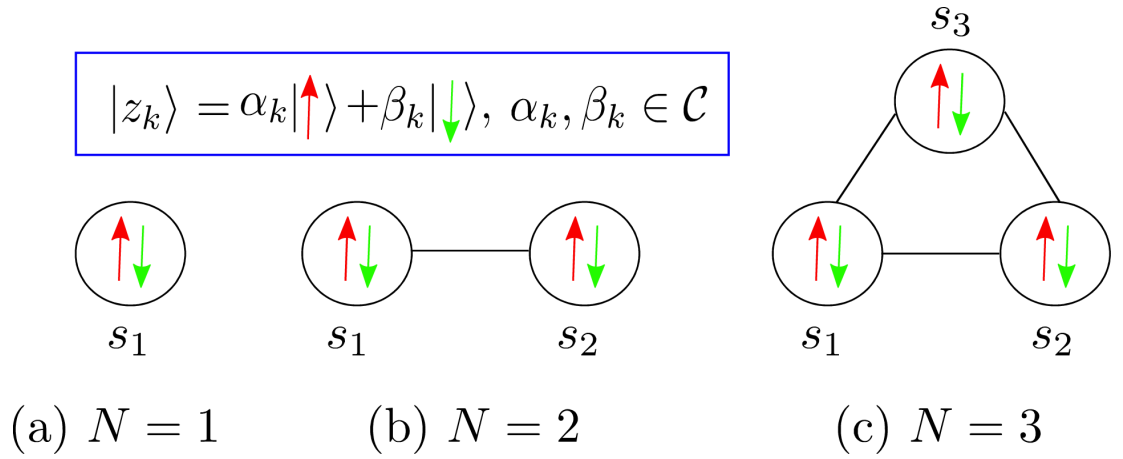


Figure 1: An example of the connections between different symbols received, where $\uparrow$ and $\downarrow$ represent the pair of legitimate possibilities for BPSK systems, respectively. Due to the quantum superposition, an unknown symbol s_k can be represented as a single qubit $|z_k\rangle = \alpha_k|\uparrow\rangle + \beta_k|\downarrow\rangle, \alpha_k, \beta_k \in \mathcal{C}$.

the eigenvalues of the Pauli-Z operator. Then, we have the Ising Hamiltonian associated with (4) as follows.

$$H_f = \sum_{l>k}^{N} 2A_{k,l}\sigma_z^{(k)}\sigma_z^{(l)} - \sum_{k=1}^{N} 2b_k\sigma_z^{(k)} + c + \sum_{k=1}^{N} A_{k,k}. \quad (7)$$

Note that the problem Hamiltonian H_f was also referred to as phase Hamiltonian in [22]. Now we verify that the problem Hamiltonian H_f such that the eigenvectors of H_f forms the solution space of the original problem (2).

Proposition 1. *Let $|z\rangle = |z_1, \cdots, z_N\rangle$ be the eigenvector of the problem Hamiltonian H_f, where $\{|z_1, z_2, \cdots, z_N\rangle, z_k \in \{0, 1\}\}$ is a basis for the 2^N-dimensional Hilbert space of the quantum computer. The solutions $\{|s_1, \cdots, s_N\rangle, s_k \in \{-1, +1\}\}$ of the ML detection problem (2) can be mapped to the eigenvectors of H_f by a bijective function such that $g : (\{|z_1, z_2, \cdots, z_N\rangle\} \rightarrow (\{|s_1, s_2, \cdots, s_N\rangle\}$.*

Proof. See Appendix A. □

Having encoded the MIMO-ML detection problem into its Hamiltonian, we can now expand it to a more general multi-user systems.

Proposition 2. *The encoding process of the problem Hamiltonian to the MIMO-ML detector is suitable for the multi-user single-input and single-output (SISO)/MIMO systems of binary symbols.*

Proof. See Appendix B □

B. The Adiabatic Theorem

Here we revisit the adiabatic theorem that was first proved in 1928 [43] for describing certain properties of particle behaviour in quantum systems, which can be formulated as follows.

Theorem 1 ([44]). *Consider a time-varying Hamiltonian $H(t)$, which starts from $H_I = H(0)$ at $t = 0$ and subsequently becomes $H_{t'}$ at some later time $t = t'$. If a quantum system is initially in the ground state H_I and as long as the change in the Hamiltonian is sufficiently slow, the system state is likely to remain in the ground state throughout the evolution, therefore being in the ground state of $H_{t'}$ at $t = t'$.*

The adiabatic theorem tells us that the quantum system can smoothly evolve from a known ground state of Hamiltonian $H(0)$ to an unknown ground state of Hamiltonian $H(T)$ given a run time T, where the state of the system evolves according to the Schrödinger equation [45]:

$$\frac{i\hbar d|\psi(t)\rangle}{dt} = H(t)|\psi(t)\rangle, \quad (8)$$

where $\hbar$ is the reduced Planck constant [45], [46] and we set $\hbar = 1$ for simplicity throughout the paper [8], [40]. Here the state $|\psi(0)\rangle$ at $t = 0$ is known and easy to construct. The Hamiltonian that governs the evolution is given by

$$H(t) = \tilde{H}(\frac{t}{T}), \quad (9)$$

where $H(t)$ denotes the Hamiltonian, which is slowly varying with t. Here $\tilde{H}(\frac{t}{T})$ is a smooth single-parameter representation of $H(t)$ and T controls the variation rate of $H(t)$. Note that a Hamiltonian H is an operator described by a Hermitian matrix, whose eigenstates and the associated eigenvalues represent the states of the system and the corresponding energy levels, respectively. Based on the quantum adiabatic theorem, the state of the quantum computer $|\psi(t)\rangle$ is close to the ground state of $H(t)$ for $0 \leq t \leq T$, and in particular $|\psi(T)\rangle$ will be close to the ground state of $H_f = H(T)$, i.e. the encoded solution of the problem considered. Consequently, following the quantum adiabatic theorem, the evolution of a system can be treated as a time independent Schrödinger equation [45], i.e., $H|\psi\rangle = \lambda|\psi\rangle$.

C. Initial Hamiltonian H_B

We now consider an N-qubit Hamiltonian H_B whose ground state is easy to find. Define the states $|x_k\rangle$ as the

eigenstates of the x-component of the k-th spin-$\frac{1}{2}$, where

$$|x_k = 0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix}, \quad \text{and} |x_k = 1\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\-1\end{pmatrix}. \tag{10}$$

Consequently, we have

$$\sigma_x^{(k)}|x_k\rangle = x|x_k\rangle, \text{ and } \sigma_x^{(k)} = \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}, \tag{11}$$

where $x = +1, -1$ and $\sigma_x^{(k)}$ is the Pauli-X operator acting on the k-th qubit. Therefore, we can express the initial Hamiltonian of (4) [20], [22] as

$$H_B = \sum_{k=1}^{N} \sigma_x^{(k)}. \tag{12}$$

In the QAOA, the initial state is usually an equiprobable superposition of the Z-basis states, which can be obtained by applying a Hadamard gate to each of the qubits in the system. Hence, the initial state is given by

$$\begin{aligned}|\psi(0)\rangle =& |x_1 = 0\rangle|x_2 = 0\rangle \cdots |x_N = 0\rangle \\ =& \frac{1}{\sqrt{2^N}} \sum_{z_1} \cdots \sum_{z_N} |z_1\rangle \cdots |z_N\rangle,\end{aligned} \tag{13}$$

where $z_k \in \{0, 1\}$ and $|x_k = 0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

IV. Quantum Adiabatic Evolution

In this Section, we firstly revisit the principles of quantum adiabatic evolution and we then provide some examples of the ML detection problem associated with one-, two- and three-qubit, respectively.

A. The Adiabatic Evolution

Following the adiabatic evolution [40], [45], we assume that the quantum system starts from a known ground state H_B and evolves to an unknown ground state H_f. Consider a linear interpolation between H_B and H_f, formulated as

$$H(t) = (1 - \frac{t}{T})H_B + H(\frac{t}{T})H_f. \tag{14}$$

Let $\tau = \frac{t}{T}$, $0 \leq \tau \leq 1$. We then have the single parameter Hamiltonian evolution of

$$\tilde{H}(\tau) = (1 - \tau)H_B + \tau H_f. \tag{15}$$

Therefore, we prepare a system by ensuring that it evolves at $t = 0$ from the ground state of $H(0) = H_B$. Let g_{min} denote the minimum difference between the two lowest eigenvalues of $\tilde{H}(\tau)$. According to the adiabatic theorem, if g_{min} is not zero and the system evolves following (8), then for a sufficiently long time T, $|\psi(T)\rangle$ will be close to the ground state of H_f [40], [47], [48], which is the encoded solution of the problem considered.

Upon defining the instantaneous eigenstates and the associated eigenvalues of $\tilde{H}(\tau)$ by

$$\tilde{H}(\tau)|\psi_l(\tau)\rangle = \lambda_l(\tau)|\psi_l(\tau)\rangle, \tag{16}$$

where $|\psi_l(\tau)\rangle$ represents the l-th eigenstates of $\tilde{H}(\tau)$ and $\lambda_l(\tau)$ is the associated eigenvalue. Furthermore, $\lambda_l(\tau)$ represents the energy levels of the quantum system, which should be sufficiently well separated, satisfying

$$s\lambda_0(\tau) \leq \lambda_1(\tau) \leq \cdots \leq \lambda_{N-1}(\tau). \tag{17}$$

According to the adiabatic theorem, if $\lambda_1(\tau) - \lambda_0(\tau) > 0$, then the expectation value obeys

$$\lim_{T\to\infty} \langle\psi_0(1)|\psi(T)\rangle = 1, \tag{18}$$

which means that there exists a $\psi(t)$ obeying (8) that is very close to the instantaneous ground state of $H(t)$ with a non-zero gap, if T is long enough.

Upon considering a particular search problem, the quantum algorithm is considered to be successful if the run time required only at most increases polynomially with the number of bits. As discussed in [40], the run time required is related to the spectrum of $\tilde{H}(\tau)$, which has to satisfy that

$$T \gg \frac{\xi}{g^2}, \tag{19}$$

where g is the minimum gap between the two lowest eigenvalues $\lambda_1(\tau)$ and $\lambda_0(\tau)$, which is formulated as

$$g_{min} = \min_{0\leq\tau\leq1} \lambda_1(\tau) - \lambda_0(\tau). \tag{20}$$

Furthermore, ξ in (19) is no higher than the largest eigenvalue of $H_f - \tilde{H}(0)$, which is usually a polynomially increasing function of n, and thus T is dominated by g_{min}^{-2} [40], [46].

B. Single-qubit Example

Consider a single-qubit problem associated with $M_t = M_r = N = 1$. Then, (4) can be rewritten as

$$f(s) = as^2 - 2bs + c, \tag{21}$$

where $a = |h|^2, b = yh$ and $c = |y|^2$. As $s \in \{-1, +1\}$, $s^2 = 1$. Thus, (21) can be cast as

$$f(s) = -2bs + a + c. \tag{22}$$

Consequently, the Hamiltonian H_f of problem (22) can be expressed as

$$H_f = -2b\sigma_z + a + c = \text{diag}([a - 2b + c, a + 2b + c]). \tag{23}$$

We can see that H_f has two different eigenvalues, namely $\lambda = a - 2b + c$ associated with the eigenstate $|0\rangle$ and $\lambda' = a - 2b + c$ associated with the eigenstate $|1\rangle$, respectively. This means that the ground state $\psi(\tau = 1)$ of H_f can be expressed as

$$\psi_0(\tau = 1) = \begin{cases}|0\rangle & \text{if } \lambda \leq \lambda'\\ |1\rangle & \text{if } \lambda > \lambda'\end{cases}. \tag{24}$$

Furthermore, from (12), we have $H_B = \sigma_x$. Thus, the smooth interpolating Hamiltonian of

$$\tilde{H}(\tau) = \begin{pmatrix}(a - 2b + c)\tau & 1 - \tau\\ 1 - \tau & (a + 2b + c)\tau\end{pmatrix} \tag{25}$$

has two eigenvalues $(a+c)\tau \pm \sqrt{1 - 2\tau + (1 + 4b^2)\tau^2}$, which will be further discussed in Section VII.

C. Two- and Three-qubit Example

Let us first consider a two-qubit example associated with a MIMO system of $N = 2$, which allows the signal values $\{-1,-1\}$, $\{-1,+1\}$, $\{+1,-1\}$ and $\{+1,+1\}$. Correspondingly, from (7), we have H_f for $N = 2$ as follows

$$\begin{aligned} H_f = {} & 2A_{1,2}\sigma_z^{(1)}\sigma_z^{(2)} - 2(b_1\sigma_z^{(1)} + b_2\sigma_z^{(2)}) \\ & + c + A_{1,1} + A_{2,2}. \end{aligned} \tag{26}$$

Furthermore, we take H_B from (12) in conjunction with $N = 2$, yielding

$$H_B = \sigma_x^{(1)} + \sigma_x^{(2)}, \tag{27}$$

which has the minimum eigenvalue of -1 associated with the eigenstate $|x_1 = 0\rangle|x_2 = 0\rangle$. The full matrix forms of H_f, H_B and $\tilde{H}(\tau)$ are given in *Appendix C*. Therefore, the corresponding interpolating Hamiltonian, namely $\tilde{H}(\tau) = (1-\tau)H_B + H_f$, can be written as the sum of terms in (7) and (12), where each term acts on two qubits.

Finally, we consider a three-qubit system, i.e. $N = 3$. The corresponding interpolated Hamiltonian, $\tilde{H}(\tau)$ can be written as the sum of

$$\begin{aligned} H_f &= 2\sum_{(k,l)\in E}\sigma_z^{(k)}\sigma_z^{(l)} - 2\sum_k \sigma_z^{(k)} + \tilde{c} \\ H_B &= \sum_{k=1}^{3}\sigma_x^{(k)}, \end{aligned} \tag{28}$$

where $E = \{(1,2),(1,3),(2,3)\}$ and $\tilde{c} = c + \sum_{k=1}^{3} A_{k,k}$. The eigenvalues of $\tilde{H}(\tau)$ for two-qubit and three-qubit ML problems will be further discussed in Section VII. The goal of the three concrete examples is to provide a general introduction to the quantum-domain optimization of the ML detection problem. Note that the run time of the quantum adiabatic algorithm depends crucially on the minimum gap $g_m in$ [47], [48]. More general results concerning the gap would be considered in our future work.

V. QAOA for Solving the ML Detection Problem

In this section, we first present the basic principles of quantum adiabatic approximation using Trotterization[1] in quantum computers [51], which usually involves a long sequence of gates. For avoiding this issue, we discuss how the QAOA may be adapted for solving the ML detection problem.

A. The Quantum Adiabatic Approximation

Recall that the evolution of a quantum system is governed by the Schrödinger's equation (8). If the system starts from some initial state $|\psi(0)\rangle$, the solution to (8) is the unitary evolution of the state [45], which is given by

$$|\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle = e^{-i(H_f+H_B)t}|\psi(0)\rangle, \tag{29}$$

describing what state the quantum system will be in after H has been applied to it over a certain time period of t. As a result, from a classical computing perspective, the adiabatic algorithm is the process where the state $|\psi(T)\rangle$ is obtained by applying a series of unitary operators to the initial state. In the adiabatic evolution algorithm, the unitary operator $U(H,T)$ is approximated by a product of unitary operators relying on a discrete-time basis by discretizing the interval $[0,T]$ into p slices denoted by $\mathcal{T} = \{t_1,\cdots,t_p\}$. Then, Trotterization is applied to approximate each discrete time slice. Specifically, the basic idea of the Trotterization technique is to decompose the system Hamiltonian into a sequence of short-time operators that are easy to simulate, and then approximate the total evolution by consecutively simulating each simpler operator. Next we first introduce the method of implementing $U(H,t) = e^{-iHt}$ at the discrete time instant t, $t \in \mathcal{T}$.

Remark 1. *The operators H_f and H_B do not commute, i.e. $[H_f, H_B] = H_f H_B - H_B H_f \neq 0$.*

The commutator of σ_z and σ_x can be calculated as $[\sigma_z,\sigma_x] = -[\sigma_x,\sigma_z] = 2i\sigma_y$, hence we see that σ_x and σ_z do not commute. Since H_f and H_B are functions of σ_z and σ_x, respectively, we have $H_f H_B - H_B H_f \neq 0$, i.e. H_B and H_f do not commute. As a consequence, the following matrix exponentials have to obey $e^{-i(H_f+H_B)t} \neq e^{-iH_f t}e^{-iH_B t}$. Based on Trotter product formula [52], we have

$$e^{-iHt} = e^{-i(H_f+H_B)t} = \lim_{r\to\infty}\left(e^{-iH_f t/r}e^{-iH_B t/r}\right)^r, r \in \mathcal{Z}. \tag{30}$$

Therefore, the unitary operator $U(H,t) = e^{-iHt}$ can be approximated by a sequence of small slices. Since the bound of (30) is only approached at $r \approx \infty$, we have to truncate the series at a finite order, such as a finite number r, for simulations on quantum computers. For each slice $e^{-iH_f t/r}$, there is an approximation based on the Trotter-Suzuki formula [49], [50], which is formulated as

$$e^{-i(H_f+H_B)t/r} = e^{-iH_f t/r}e^{-iH_B t/r} + \epsilon, \tag{31}$$

where ϵ is the approximation error. From the Baker–Campbell–Hausdorff formula of [6], the norm of the error obeys $\|\epsilon\|_2 \leq O(\|iH_f t/r\|\cdot\|iH_B t/r\|) = O\big(\frac{t^2}{r^2}\|H_f\|\|H_B\|\big)$.

Therefore, the unitary operator $U(H,T) = e^{-iHT}$ can be written as a product of p unitary operators that are easy to simulate,

$$\begin{aligned} U(H,T) &= e^{-iHT} \\ &= U(H,t_p = T-\triangle)U(H,t_{p-1} = T-2\triangle) \\ &\quad \cdots U(H,t_1 = 0) \\ &= \prod_{k=1}^{p}\left(e^{-iH_f t_k/r}e^{-iH_B t_k/r}\right)^r, \end{aligned} \tag{32}$$

where we have $U(H,t_k) = e^{-iH_f t_k/r}e^{-iH_B t_k/r}$ and $\triangle = \frac{T}{p}$. We can see that implementing these unitary operators requires a quantum circuit of depth $2pr$, which indicates that the length of the circuits depends both on the evolution time and on the Trotter steps. As a result, the depth of the associated quantum circuits grows with the product of p and r, which

[1]Trotterzation is a very useful tool for simulating non-commuting operators in quantum computers, by using the Trotter-Suzuki formula [49], [50] to approximately decompose the system operator into a sum of easy to implement operators.

usually requires a very long circuit. Gate errors and decoherence restrict the number of sequential gate operations in the quantum devices, therefore a class of hybrid classical quantum algorithms are developed for near-term quantum computing [15], [16]. Indeed, QAOA belongs to the family of hybrid classical quantum algorithms, which creates a parameterized quantum state by alternately applying the Hamiltonian H_f and H_B p times for a given p.

B. Implementation of QAOA

Instead of using Trotterization methods [51], [53], QAOA prepares a pair of unitary operators in terms of H_f and H_B. For ease of computation, we introduce the following remark to simplify H_f.

Remark 2. *The objective of ML detection is to minimize the objective function* (4)*, which is equivalent to minimizing the following function*

$$f(|s\rangle) = \sum_{l>k}^{N} A_{k,l} s_k s_l - \sum_{k=1}^{N} b_k s_k, \tag{33}$$

by omitting the constant values in (4)*. Consequently, the problem Hamiltonian H_f of the ML detection is equivalently transformed into*

$$H_f = \sum_{l>k}^{N} A_{k,l} \sigma_z^{(k)} \sigma_z^{(l)} - \sum_{k=1}^{N} b_k \sigma_z^{(k)}. \tag{34}$$

Given H_f of (34), the QAOA prepares a parameterized unitary operator in terms of H_f depending on an angle γ as follows:

$$U(H_f, \gamma) = e^{-i\gamma H_f} \tag{35a}$$

$$= \prod_{k=1}^{N} \prod_{l>k}^{N} e^{-i\gamma A_{k,l} \sigma_z^{(k)} \sigma_z^{(l)}} e^{i\gamma b_k \sigma_z^{(k)}} \tag{35b}$$

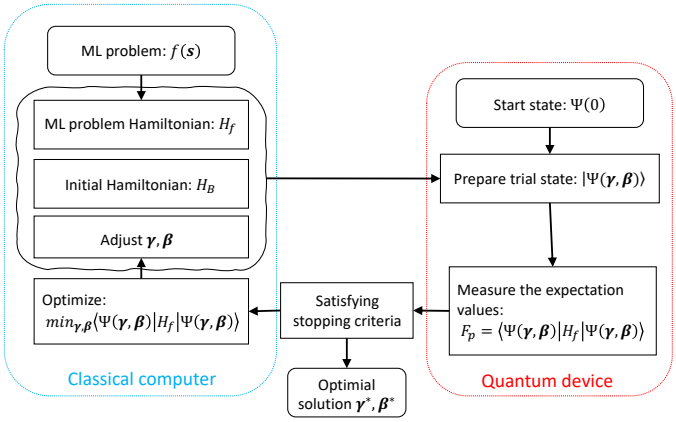
$$= \prod_{k=1}^{N} \prod_{l>k}^{N} U(A_{k,l}, \gamma) U(b_k, \gamma), \tag{35c}$$

with $\gamma \in [0, 2\pi]$. Note that since all terms in H_f of (7) are diagonal in the computational basis, they commute with each other. As a result, $U(H_f, \gamma)$ can be written as a sequence of unitary operators formulated in (35b) and (35c), where the unitary operators are $U(A_{k,l}, \gamma) = e^{-i\gamma A_{k,l} \sigma_z^{(k)} \sigma_z^{(l)}}$ and $U(b_k, \gamma) = e^{i\gamma b_k \sigma_z^{(k)}}$, respectively. Based on the initial Hamiltonian H_B of (12), let us define the parameterized unitary operator $U(H_B, \beta)$ depending on the angle β as

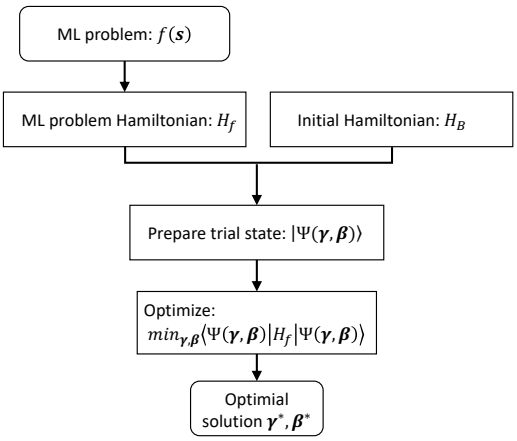
$$U(H_B, \beta) = e^{-i\beta H_B} = \prod_{k=1}^{N} e^{-i\beta \sigma_x^{(k)}}, \tag{36}$$

where we have $\beta \in [0, \pi]$. Here the parameters γ and β in (35) and (36) represent the evolution times of the quantum system, also referred to as angles [20], which are used to create the parameterized QAOA circuit.

Following the VQE principle, the QAOA creates a parameterized state in terms of γ and β on the quantum computer using single-qubit and entangling gates as follows. Given an



(a) Optimizations of F_p based on VQE principles.



(b) Optimizations of F_p using classical solver only.

Figure 2: The QAOA diagram for optimization F_p.

integer p, there are $2p$ angles $\boldsymbol{\gamma} = [\gamma_1, \cdots, \gamma_p]$ and $\boldsymbol{\beta} = [\beta_1, \cdots, \beta_p]$. Specifically, in the QAOA, an angle-dependent quantum state is created by alternately applying Hamiltonians H_f and H_B in p consecutive rounds, which can be expressed as

$$\begin{aligned} |\psi_p(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle = U(H_B, \beta_p) U(H_f, \gamma_p) \cdots \\ U(H_B, \beta_1) U(H_f, \gamma_1) |\psi(0)\rangle. \end{aligned} \tag{37}$$

Another important component of the QAOA is the computation of the expectation value of H_f in the state of $|\psi_p(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle$, which can be expressed as

$$\begin{aligned} F_p &= \langle \psi_p^\dagger(\boldsymbol{\gamma}, \boldsymbol{\beta}) | H_f | \psi_p(\boldsymbol{\gamma}, \boldsymbol{\beta}) \rangle \\ &= \langle \psi(0) | U^\dagger(H_f, \gamma_1) \cdots U^\dagger(H_B, \beta_p) H_f U(H_B, \beta_p) \cdots \\ &\quad U(H_f, \gamma_1) | \psi(0) \rangle, \end{aligned} \tag{38}$$

where $\dagger$ denotes the conjugate transpose. Since the expectation value of F_p relies on the parameterized state $\psi_p^\dagger(\boldsymbol{\gamma}, \boldsymbol{\beta})$, the goal of the QAOA is to approximate the optimal $\boldsymbol{\gamma}^*$ and $\boldsymbol{\beta}^*$ that satisfies

$$\boldsymbol{\gamma}^*, \boldsymbol{\beta}^* = \arg\min F_p(\boldsymbol{\gamma}, \boldsymbol{\beta}). \tag{39}$$

Specifically, in QAOA, the gate parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are designed on the classical computer by optimizing the expectation values of H_f, which is obtained by measuring the state of the

quantum system. Fig. 2 illustrates two schemes of realizing QAOA. Fig. 2(a) illustrates the main steps of realizing QAOA using hybrid quantum-classical approaches based on VQE principles, which is suitable for the case, where the depth of the quantum circuits is excessive for NISQ devices or the number of qubits is too high for computations to be carried out by a classical computer. Explicitly, the parameterized trial states $\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})$ are created on the quantum device, starting from the initial state $\psi(0)$. The output from the quantum devices is the measurement of the expectation values of the state $\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})$, which are fed into the classical computer for updating $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ by the classical optimizer. The new parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are then fed back to the quantum device to adjust the system state. The algorithm terminates, when the minimized value of F_p is reached. By contrast, when the circuit is shallow and when the number of qubits in terms of the problem Hamiltonian H_f is not too high, the expectation value F_p can be calculated classically, as illustrated in Fig. 2(b). To this end, the analytical expression of F_1 can be derived, which is given in **Proposition 3**.

Proposition 3. *For the QAOA associated with $p = 1$, the expectation value F_1, depending on a pair of parameters (γ, β) omitting the subscript for notational simplicity, can be calculated analytically, which is given as follows:*

1) *When $N = 1$, we arrive at:*

$$F_1 = b \sin(2\beta) \sin(2b\gamma). \tag{40}$$

2) *When $N = 2$, we have*

$$F_1 = A_{1,2} f_{1,2} + \sum_{k=1}^{2} g_k, \tag{41}$$

where

$$\begin{aligned} f_{1,2} =& \langle ++ | U^\dagger(A_{1,2}, \gamma) U^\dagger(b_1, \gamma) U^\dagger(b_2, \gamma) U^\dagger(\beta^{(1)}) \\ & U^\dagger(\beta^{(2)}) \sigma_z^1 \sigma_z^2 U(\beta^{(2)}) U(\beta^{(1)}) U(b_2, \gamma) U(b_1, \gamma) \\ & U(A_{1,2}, \gamma) | ++ \rangle, \\ g_k =& \langle ++ | U^\dagger(A_{1,2}, \gamma) U^\dagger(b_1, \gamma) U^\dagger(b_2, \gamma) U^\dagger(\beta^{(1)}) \\ & U^\dagger(\beta^{(2)}) \sigma_z^k U(\beta^{(2)}) U(\beta^{(1)}) U(b_2, \gamma) U(b_1, \gamma) \\ & U(A_{1,2}, \gamma) | ++ \rangle, \end{aligned}$$

with $U(\beta^{(k)}) = e^{-i\beta\sigma_x^{(k)}}$ representing the unitary operator acting on the k-th qubit.

3) *When $N \geq 3$, we have*

$$F_1(\gamma, \beta) = \sum_{l>k} A_{k,l} f_{k,l} - \sum_k b_k g_k, \tag{43}$$

where

$$\begin{aligned} f_{k,l} =& \langle +^N | U_N^\dagger(\gamma, \beta) \cdots U_1^\dagger(\gamma, \beta) \sigma_z^{(k)} \sigma_z^{(l)} U_1(\gamma, \beta) \cdots \\ & U_N(\gamma, \beta) | +^N \rangle, \quad \textit{and} \\ g_k =& \langle +^N | U^\dagger(A_{k',k}, \gamma) U^\dagger(A_{k,k''}, \gamma) U^\dagger(b_k, \gamma) U^\dagger(1, \beta) \\ & \sigma_z^k U(1, \beta) U(b_k, \gamma) U(A_{k,k''}, \gamma) U^\dagger(A_{k',k}, \gamma) \\ & \sigma_z^{(k)} | +^3 \rangle, \quad k' < k < k'', \end{aligned}$$

with $U_k(\gamma, \beta) = \prod_{l>k} U(A_{k,l}, \gamma) U(b_k, \gamma) U(k, \beta)$ with $k = 1, \cdots, N$.

Proof. See Appendix D. □

VI. Complexity Analysis for Fixed p

In this section, we provide the computational complexity analysis of the QAOA for solving the ML detection problem of interest, when the analytical expression of F_p is given. In this context, the maximization of F_p can be solved by classical solvers such as COBYLA [54] via linear approximation techniques. Then, the quantum circuits of QAOA are constructed based on the resultant $\boldsymbol{\gamma}^*$ and $\boldsymbol{\beta}^*$. The complexity of QAOA in terms of its quantum implementation directly depends on the number of gates required. We see from (35c) that $U(H_f, \gamma)$ is represented as a product of $(N-1)N/2$ unitary operators $U(A_{k,l}, \gamma)$ and N unitary operators $U(b_k, \gamma)$, which indicates that implementing $U(H_f, \gamma)$ requires $(N+1)N/2$ unitary gates [55]. Similarly, the implementation of $U(H_B, \beta)$ requires N unitary gates, since $U(H_B, \beta)$ consists of N unitary operators in terms of σ_x, as given in (36). Observe from (35c) and (36) that implementing $U(H_f, \gamma)$ and $U(H_B, \beta)$ requires $(N+3)N/2$ unitary gates in total. Furthermore, the preparation of the initial state ψ_0 requires N Hadamard gates, while the circuits of QAOA associated with the depth of p involves $(N+5)Np/2$ quantum gates altogether. In the following, we discuss the computational complexity of the classical solver of (39) as well as the memory requirement of simulating the QAOA on a classical computer. It should be noted that the complexity analysis of this section serves the readers of the communications society, which demonstrates the potential of QAOA in terms of solving an NP-hard ML detection problem, while reflecting the challenges of implementing QAOA to a degree. Quantifying the advantage and limitations of the QAOA in terms of different problem instances are investigated in [21], [42], [56], [57], which may be considered for the ML detection problem in our future work.

1) Computational complexity of the classical optimizer: The optimization of the expectation value F_p of (39) involves $2p$ variables in total. Since we use COBYLA as the classical solver for optimization of (39), it requires $O(m^2)$ function evaluations for each iteration [54]. Note that m is the number of interpolation points, which is given by $m = \frac{1}{2}(2p+1)(2p+2) = (p+1)(2p+1) = 2p^2 + 3p + 1$. The number of function evaluations for each iteration is thereby $O(p^4)$. From (38) and **Proposition 3**, we see that computing F_p relies on matrix dot products and matrix exponential operations. In the worst case, all of the unitary operators are associated with a $2^N \times 2^N$ matrix, hence the computational complexities of matrix exponential evaluations and matrix dot products are on the order of $O(2^{3N})$ and $O(2^{2N})$, respectively. As seen from (38), the computation of F_p requires $4p$ matrix exponential operations and $4p+1$ matrix dot product operations. As a result, the computational complexity of each evaluation F_p is given by

$$\begin{aligned} O(F_p) &= 4pO(8^N) + (4p+1)O(4^N) \\ &= O(p8^N). \end{aligned} \tag{45}$$

Therefore, the total computational complexity of solving problem (39) is on the order of $O(Ip^5 8^N)$, where I is the number of the iterations. This indicates that the computational

complexity of the classical solver grows exponentially with N even if $p = 1$.

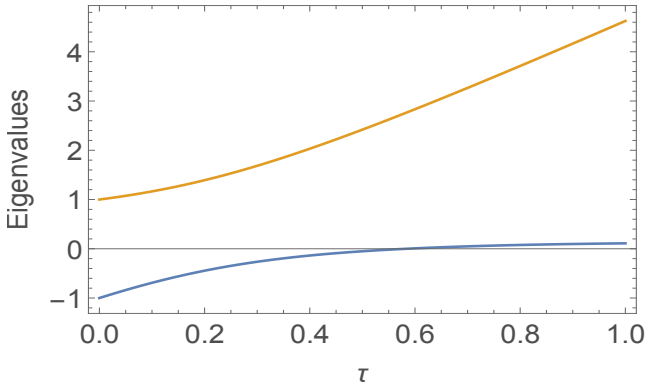
2) Memory requirement of simulating the QAOA classically: When evaluating F_p classically, the classical computer will have to store both the quantum states and the unitary operators in terms of $U(H_f, \gamma)$ and $U(H_B, \beta)$. Specifically, a state of N qubits is characterized by a 2^N-element complex vector, which requires 2^N complex numbers and 2^{N+1} floating-point numbers. In a classical computer, a floating-point number is typically represented by four bytes in single-precision floating-point format. Correspondingly, we have to use 2^{3+N} bytes for storing a quantum state of N qubits. Furthermore, a unitary matrix contains 2^{2N} complex numbers, which thereby occupies 2^{2N+3} bytes in the memory. Therefore, the amount of memory required is dominated by storing the unitary operators. For instance, if we have a computer equipped with 16G RAM, the number of qubits that can be handled is at most $N = 15$. On the other hand, based on the VQE principle, evaluating the expectation value containing the parameterized state and the unitary operators is performed by the quantum devices, while the classical computer only needs to store the current quantum state that is simulating at the quantum device. In this context, the amount of memory required depends on the storage required by a single quantum state. Correspondingly, the maximum number of qubits that can be simulated using 16G RAM is $N = 30$.

VII. Simulation Results

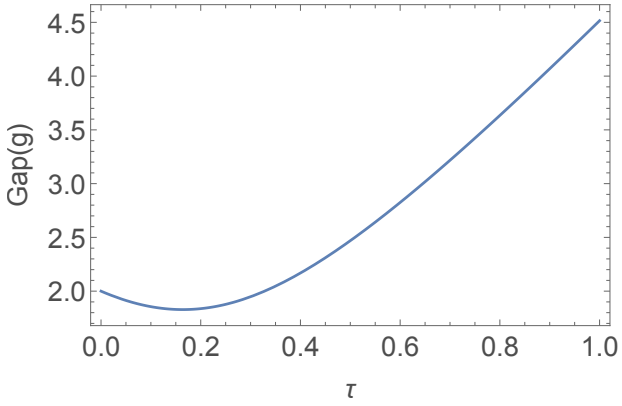
In this section, we first characterize the eigenvalues of the problem Hamiltonian for the ML detection. Then, we visualize the expectation values F_1 in terms of different number of qubits. Finally, we quantify the performance of QAOA in the ML detection of MIMO systems using computer simulations. As for performing quantum computing, Qiskit Aer [58] is used for implementing the noise-free simulations of QAOA circuits.

Fig. 3 depicts the evolution of the eigenvalues for a single-qubit ML detection problem, where we consider a concrete example associated with a random channel coefficient of $h = 1.2416$, $n = 0.3323$ and the transmit signal $s = 1$. The two eigenvalues of $\tilde{H}(s)$ and the corresponding gap (g) of (20) are plotted in Fig. 3(a) and in Fig. 3(b), respectively. We find that the minimum gap $g = 1.94$ is not small and thus it is possible for driving the system to evolve smoothly from $\psi(0) = |x = 0\rangle$ to $|z = 0\rangle$.

Furthermore, the evolutions of eigenvalues for the two-qubit and three-qubit ML detection problems are portrayed in Fig, 4. In Fig. 4(a), we plot the four eigenvalues of $\tilde{H}(\tau)$ for a group of random system parameters, where the values are set as $\mathbf{H} = [[1.2416, -0.1741], [0.3323, -0.0804]]$, $n = [-1.5130, 0.3212]$ and $\mathbf{s} = [-1, +1]$. Then, the minimum gap of (20) can be found as $g = 1.93$, which indicates that the optimal solution may indeed be obtained by smoothly evolving the system from the initial ground state to the final ground state, when $T \gg 1/g^2$. Fig. 4(b) illustrates the eigenvalues of $\tilde{H}(\tau)$ for a three-qubit ML problem, where we have the channel matrix of $\mathbf{H} = [[1.24155, -0.174105, 0.332349], [-0.080418, -1.51301, 0.321184], [-1.7771, 1.55398, 0.23342]]$, the noise of $\mathbf{n} = [-1.703, -1.77439, 1.34985]$ and the transmit signal sequence is $\mathbf{x} = [-1, 1, 1]$. We can see that the minimum gap is non-zero, which indicates convergence to the optimal solution. Furthermore, we see that there are some overlapping eigenvalues in Fig. 4, which arises from the symmetry of $\tilde{H}(\tau)$ [40].



(a) Different eigenvalues

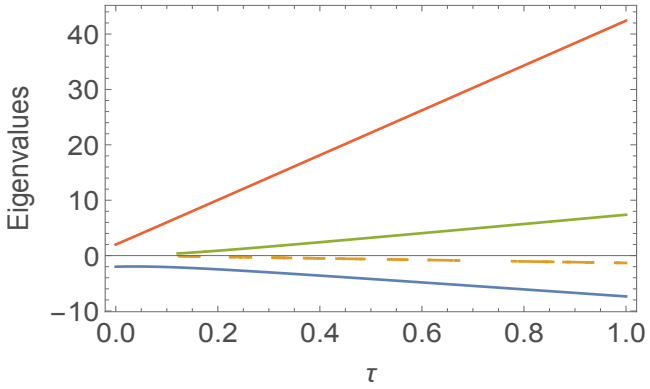


(b) Gap

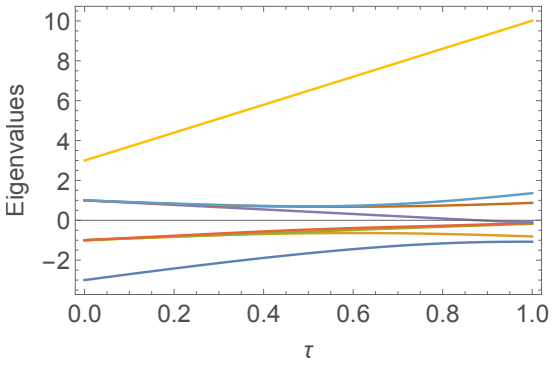
Figure 3: The two eigenvalues of $\tilde{H}(\tau)$ for a single-qubit example.

Fig. 7 shows the landscape of expectation values for a depth of $p = 1$ circuits with two parameters γ_1 and β_1, where the expectation values of $F_1(\gamma_1, \beta_1)$ for $N = 1, 2$ and 3 are plotted in (40), (41) and (43), respectively. Since the analytical expressions of $F_1(\gamma_1, \beta_1)$ for $N = 1, 2$ and 3 are given in **Proposition 3**, we plot their figures using Mathematica. Furthermore, the values of the channel coefficients and the noise are the same as in the corresponding numerical examples of Fig. 3 and Fig. 4. As the goal of QAOA is to minimize the expectation values F_1, the axes of F_1 are plotted bottom-to-top. We see that in Fig. 7 F_1 is non-convex in terms of γ_1 and β_1, since it has multiple local minima. Furthermore, the number of locally optimal points grows upon increasing both N and the depth p of the circuits, especially when the expectation values have to be evaluated by sampling from the quantum circuit measurements. This poses challenges in classical simulations.

As discussed in Section VI, simulating QAOA in classical computers requires excessive amount of memory and computational power. Therefore, we confine our simulations to the

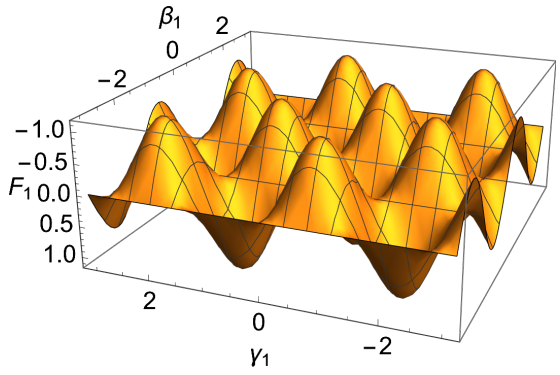


(a) Two qubits

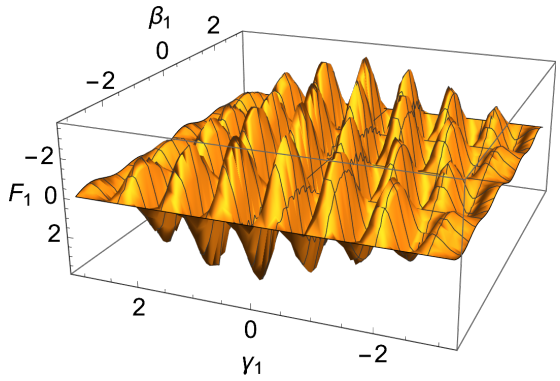


(b) Three qubits

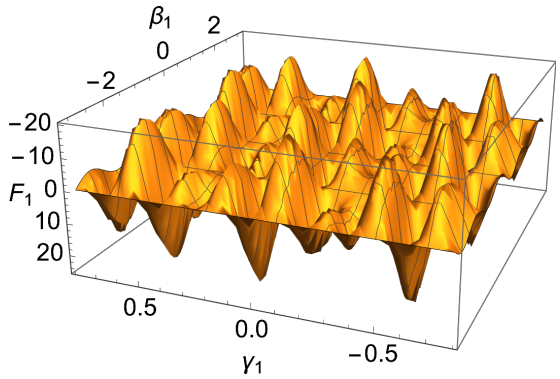
Figure 4: Different eigenvalues of $\tilde{H}(\tau)$ for two and three-qubit examples.



(a) $N = 1$



(b) $N = 2$



(c) $N = 3$

Figure 5: The expectation value function $F_1(\gamma_1, \beta_1)$.

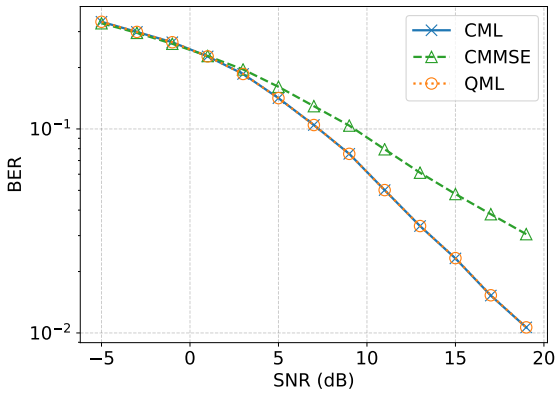
ML detection problem within 3 qubits. In the simulations, the MIMO channel $\mathbf{H}$ and the noise $\mathbf{n}$ were chosen as independent and identically distributed, zero-mean, real-valued normal random variables, i.e. $\mathbf{H} \sim \mathcal{N}(0, 1)$ and $\mathbf{n} \sim \mathcal{N}(0, 1)$. Furthermore, for each SNR, we perform 40 000 Monte Carlo simulations for estimating the average probability of errors in detecting the message vector. For benchmarking the QAOA based ML (QML) detector, we consider a pair of conventional detection methods: Classical ML (CML) and classical MMSE (CMMSE). Fig. 6 illustrates the BER of the three different detection methods for $N = 2$ and 3, where we consider a MIMO system having the same number of transmit and receive antennas, i.e. $M_t = M_r = N$. Here, the best solution for ML detection problem is selected from the distribution of the solutions obtained by the QAOA. Observe from Fig. 6 that the QAOA based ML detector approaches the BER of classical ML detector, and as expected both outperform the MMSE detector. Furthermore, in Fig. 6(a), we can see that the BER curve of QML perfectly matches that of CML. However, in Fig. 6(b) the BER of QML becomes slightly worse than that of CML in the high-SNR region. This is because the QML solution is estimated statistically relying on the quantum circuit measurements.

Below we further study the approximation ratio of the QAOA in solving the ML detection problem, where the expectation values of $F_p(\boldsymbol{\gamma}, \boldsymbol{\beta})$ are attained by measuring the QAOA circuits generated by Qiskit Aer [58] and the parameters $\boldsymbol{\gamma}, \boldsymbol{\beta}$ are updated using COYBLA [54]. For obtaining a quality solution of the QAOA, we set a budget of 2000 runs for each problem instance with respect to each channel realization in the simulations, where the COYBLA starts from different random initial points in each run. Here, the approximation ratio is defined as follows:

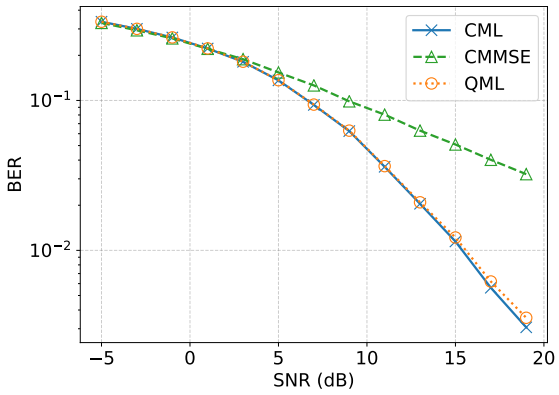
$$\rho = \frac{f_{CML}}{F_p(\boldsymbol{\gamma}^*, \boldsymbol{\beta}^*)}, \tag{46}$$

where f_{CML} is the objective function value obtained by the classical ML detection method. Observe that we have $\rho = 1$ when the optimal solution is attained by the QAOA.

Fig. 7 depicts the approximation ratio of the QAOA in solving the ML detection problem associated with $N = 1$, $N = 2$ and $N = 3$, where the SNR is 10 dB. Specifically, Fig. 7(a) - Fig. 7(c) show the evolution of the approximation



(a) $N = 2$



(b) $N = 3$

Figure 6: The BER performance of MIMO systems with different number of users.

ratio parameterized by different p values versus the number of runs. Observe that the approximation ratio gradually increases and converges to a stable value, when the number of runs increases for each N, since the more runs are performed, the higher the probability of attaining better $(\boldsymbol{\gamma}, \boldsymbol{\beta})$ parameters for the QAOA circuits becomes. Furthermore, the approximation ratio improves with p increases. More explicitly, we compare the approximation ratio versus p in Fig. 7(d), where 2000 runs are performed for each problem instance with respect to each channel realization. A maximum of $p = 10$ is considered for $N = 1$ and $N = 2$, since the approximation ratio saturates within $p = 10$. Moreover, a maximum of $p = 20$ is considered along with $N = 3$ for attaining a good approximation ratio. We observe that the approximation ratio increases monotonically with p for $N = 3$, which indicates that the approximation ratio can still be improved by further increasing p, even if $p > 20$. Therefore, a large p is required by the QAOA for achieving a good approximation ratio, when $N \geq 3$.

Finally, we note that the QAOA is capable of solving the ML detection problem of BPSK even for a large-scale system, since it is a general-purpose algorithm designed for solving combinatorial problems [20], [59]. However, the performance of QAOA depends on the circuit depth p and it improves as p increases [20]. Moreover, the investigations in [20], [27], [33] indicate that the performance of the QAOA highly depends on the structure of a graph. In our paper, the Hamiltonian function H_f formulated for the ML detection of BPSK systems is constituted by a fully connected graph (a complete graph), where each edge is associated with a quadratic term in H_f. Explicitly, for a $N \times N$ MIMO system associated with BPSK that can be mapped to a complete graph of N vertices, the Hamiltonian function consists of $(N-1)N/2$ quadratic terms and N linear terms, which is associated with an N-qubit quantum system. Therefore, for a large-scale MIMO BPSK system, a higher circuit depth (larger p) would be required for obtaining a good approximation ratio. Furthermore, a p-level QAOA circuit involves $2p$ parameters to be optimized in the classical computer, which complicates the classical optimizer due to its heuristics. As a result, we confine our simulations within a small number of qubits. In our future research we would test the QAOA on large-scale MIMO systems, using the results derived in this treatise as a springboard.

VIII. Conclusions

In this paper, we studied the performance of QAOA based ML detection problems, where we considered the ML detection of binary symbols over a MIMO channel. We first encoded the optimal solution of the ML detection problem into the ground state of a problem Hamiltonian and presented the energy evolution of the quantum system of interest. For level-1 QAOA, we derived the analytical expressions of QAOA and provided the energy landscape of QAOA that illustrates its symmetry vs. the parameter values to be optimized. Finally, our simulation results revealed that QAOA based ML detection is capable of approaching the BER of the classical ML detector, while both outperform the classical MMSE detector. The performance of the QAOA depends on the structure of the graph associated with the Hamiltonian function [20], [27], [33]. Hence more comprehensive results associated with more sophisticated system models would be considered in our future research.

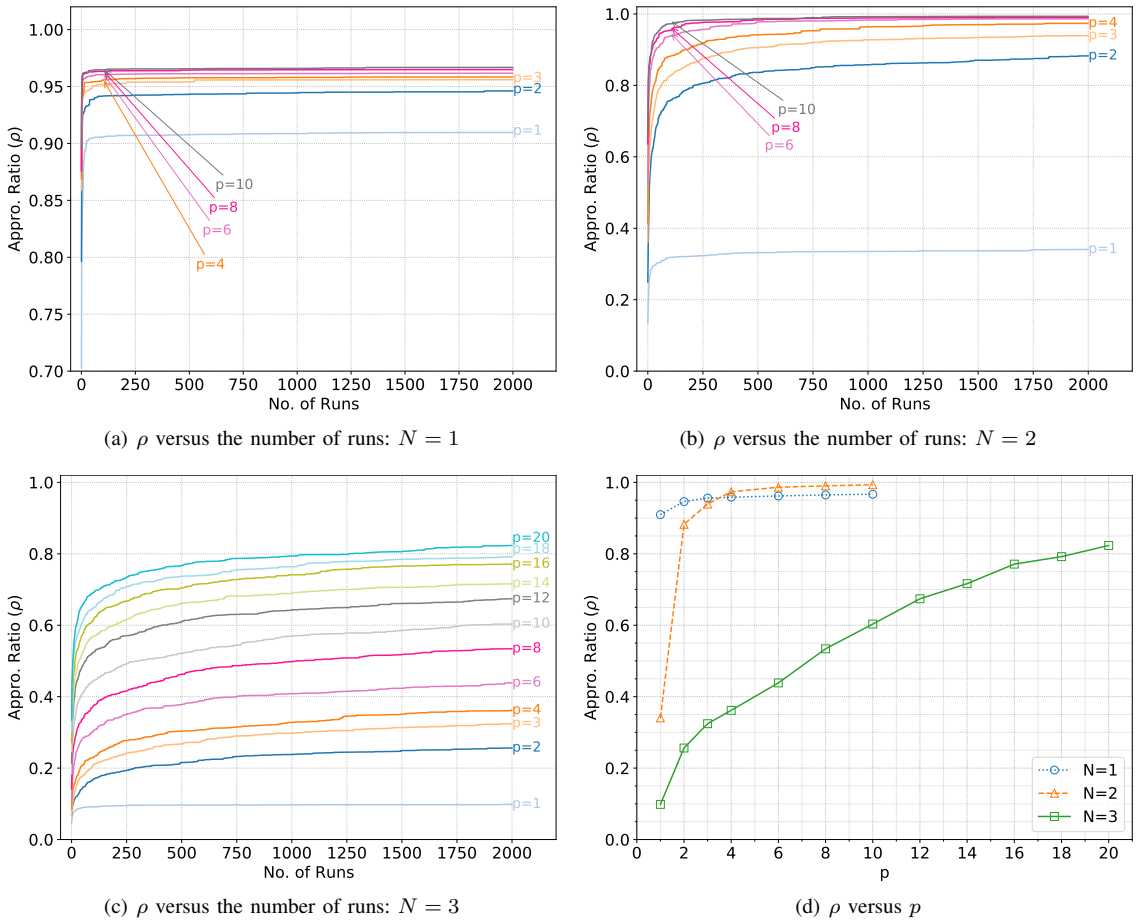
Appendix A: Proof of The Problem Hamiltonian H_f

For mapping the variable $z_k = \{0, 1\}$ into $s_k \in \{-1, +1\}$, we define a bijection function $g(z_k)$: $s_k = g(z_k) = 1 - 2z_k$[2]. Correspondingly, the objective function of (4) can be equivalently reformulated as

$$f(|z\rangle) = \sum_{l>k}^{N} 2A_{k,l}(1-2z_k)(1-2z_l) - \sum_{k=1}^{N} 2b_k(1-2z_k) + c + \sum_{k=1}^{N} A_{k,k}. \tag{A.1}$$

Furthermore, we can rewrite (6) as $\sigma_z^{(k)}|z_k\rangle = (1-2z_k)|z_k\rangle = s_k|z_k\rangle$, which indicates that the binary variables s_k is mapped onto the eigenvalues of the Pauli-Z operator. Now we show that $H_f|z\rangle = f(|z\rangle)|z\rangle$ in (A.2). Therefore, we can see that the objective values of the ML detector is mapped onto the eigenvalues of the problem Hamiltonian H_f, where each bit string $|z\rangle$ associated with a variable vector $|s_1, \cdots, s_N\rangle$.

[2]Note that the mapping function $g(z_k)$ is usually not unique, any bijective function such that $g : \{0, 1\} \rightarrow \{-1, 1\}$ can be chosen as the mapping function such as $g(z_k) = (-1)^{z_k}$.



(a) ρ versus the number of runs: $N = 1$

(b) ρ versus the number of runs: $N = 2$

(c) ρ versus the number of runs: $N = 3$

(d) ρ versus p

Figure 7: Comparison of approximation ratio (ρ) for different N.

$$
\begin{aligned}
H_f|z\rangle &= \left[\sum_{l>k}^{N} 2A_{k,l}\sigma_z^{(k)}\sigma_z^{(l)} - \sum_{k=1}^{N} 2b_k\sigma_z^{(k)} + c + \sum_{k=1}^{N} A_{k,k}\right]|z_1,\cdots,z_n\rangle \\
&= \sum_{l>k}^{N} 2A_{k,l}\sigma_z^{(k)}\sigma_z^{(l)}|z_1,\cdots,z_n\rangle - \sum_{k=1}^{N} 2b_k\sigma_z^{(k)}|z_1,\cdots,z_n\rangle + (c+\sum_{k=1}^{N} A_{k,k})|z_1,\cdots,z_n\rangle \\
&= \sum_{l>k}^{N} 2A_{k,l}(1-2z_k)(1-2z_l)|z_1,\cdots,z_n\rangle - \sum_{k=1}^{N} 2b_k(1-2z_k)|z_1,\cdots,z_n\rangle + (c+\sum_{k=1}^{N} A_{k,k})|z_1,\cdots,z_n\rangle \\
&= \left[\sum_{l>k}^{N} 2A_{k,l}(1-2z_k)(1-2z_l) - \sum_{k=1}^{N} 2b_k(1-2z_k) + c + \sum_{k=1}^{N} A_{k,k}\right]|z_1,\cdots,z_n\rangle \\
&= f(|z\rangle)|z\rangle.
\end{aligned}
\tag{A.2}
$$

Appendix B: Proof of **Proposition 2**

Consider a multi-user SISO system, where K users cooperatively transmit independent symbols to a receiver. The receiver observes the sum of the modulated signals contaminated by the noise as follows

$$
y = \sum_{k=1}^{K} h_k s_k + n = \mathbf{h}^T\mathbf{s} + n, \tag{B.1}
$$

where $\mathbf{h} = [h_1,\cdots,h_K]^T$ and $\mathbf{s} = [s_1,\cdots,s_K]^T$. Therefore, the objective function of our ML detection problem becomes

$$
\begin{aligned}
f(\mathbf{s}) &= \mathbf{s}^T\mathbf{h}\mathbf{h}^T\mathbf{s} - 2y\mathbf{h}^T\mathbf{s} + y^2 \\
&= \mathbf{s}^T\mathbf{A}\mathbf{s} - 2\mathbf{b}\mathbf{s} + c \\
&= \sum_{l>k}^{K} 2A_{k,l}s_k s_l - \sum_{k=1}^{K} 2b_k s_k + c + \sum_{k=1}^{K} A_{k,k},
\end{aligned}
\tag{B.2}
$$

where $\mathbf{A} = \mathbf{h}\mathbf{h}^T$, $\mathbf{b} = y\mathbf{h}^T$, $c = y^2$ and $\mathbf{s} \in \{-1,+1\}^K$. Furthermore, $A^T = A$ is used to in the last step of (B.2).

Following the form of (7), we have

$$H_f = \sum_{l>k} A_{k,l}\sigma_z^{(k)}\sigma_z^{(l)} - b\sum_{k=1}^{K}\sigma_z^{(k)}. \tag{B.3}$$

Note that the constant terms and the coefficient 2 are omitted in (B.3), since they do not affect the optimal solution of the original problem. From (B.3), we can see that the number of qubits required is equal to the number of users for multi-user SISO systems. Now we extend the signal model to the corresponding multi-user MIMO system, which can be expressed as:

$$\mathbf{y} = \sum_{k=1}^{M}\mathbf{H}_k\mathbf{s}_k + \mathbf{n} = \mathbf{H}^T\mathbf{s} + \mathbf{n}, \tag{B.4}$$

where $\mathbf{H} = [\mathbf{H}_1, \cdots, \mathbf{H}_k]^T$ and $\mathbf{s} = [\mathbf{s}_1^T, \cdots, \mathbf{s}_K^T]^T$. Without loss of generality, we assume that $\mathbf{H}_k \in \mathcal{R}^{N\times N}$ and $\mathbf{s}_k \in \{-1, +1\}^N$. As a result, we have $\mathbf{y}$ and $\mathbf{s}$ are $N \times 1$ and $NK \times 1$ vectors, respectively. Based on (B.4), the objective function of our ML detection problem can be expressed as

$$\begin{aligned} f(\mathbf{s}) &= \mathbf{s}^T\mathbf{H}\mathbf{H}^T\mathbf{s} - 2\mathbf{y}^T\mathbf{H}^T\mathbf{s} + \mathbf{y}^T\mathbf{y} \\ &= \sum_{l>k}^{NK} 2A_{k,l}s_k s_l - \sum_{k=1}^{NK} 2b_k s_k + c + \sum_{k=1}^{NK} A_{k,k}, \end{aligned} \tag{B.5}$$

where $\mathbf{A} = \mathbf{H}\mathbf{H}^T$ with $\mathbf{A}^T = \mathbf{A}$, $\mathbf{b} = \mathbf{y}^T\mathbf{H}^T$, $c = \mathbf{y}^T\mathbf{y}$ and $\mathbf{s} \in \{-1, +1\}^{NK}$. Therefore, the Ising Hamiltonian of the ML detection to the corresponding multi-user MIMO system can be expressed as

$$H_f = \sum_{l>k}^{NK} A_{k,l}\sigma_z^{(k)}\sigma_z^{(l)} - \sum_{k=1}^{NK} b_k\sigma_z^{(k)}, \tag{B.6}$$

where the constant terms and the coefficient 2 are also omitted. We can see from (B.6) that the number of qubits required is NK for the ML detection in a quantum computer.

Appendix C: Matrix Form of Two-bit Example

Here, the full matrix forms of H_f, H_B and $\tilde{H}(\tau)$ for a two-qubit ML detection problem are given in (C.1), (C.2) and (C.3), respectively, where $c' = c + A_{1,1} + A_{2,2}$. Here, $\sigma_z^{(1)}\sigma_z^{(2)}$ in (C.1) denotes the tensor product of $\sigma_z^{(1)}$ and $\sigma_z^{(2)}$. Moreover, $\sigma_x^{(1)}$ and $\sigma_x^{(2)}$ in (C.2) represent the Pauli-X operators acting on the 1st and 2nd qubits, respectively. Therefore, $H_B = \sigma_x^{(1)} + \sigma_x^{(2)}$ represents the operation of applying the Pauli-X operators to the qubits of our 2-qubit system, which is obtained from the general form of the initial Hamiltonian of (12) for the QAOA. More explicitly, for the two-qubit ML detection problem, we have $H_B = \sigma_x^{(1)} + \sigma_x^{(2)} = \sigma_x^{(1)} \otimes I + I \otimes \sigma_x^{(2)}$, where $\otimes$ and I denote the tensor product operator and the identity matrix, respectively.

$$H_B = \sigma_x^{(1)} + \sigma_x^{(2)} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}. \tag{C.2}$$

Appendix D: Proof of Proposition 3

For $p = 1$ and $N = 1$: From (34) that omits the constants of (23), we arrive at the problem Hamiltonian in the form of $H_f = -b\sigma_z$. Consequently, we have the unitary operators associated with $N = 1$ as follows:

$$\begin{aligned} U(H_B, \beta_1) &= e^{-i\beta_1\sigma_x}, \text{ and} \\ U(H_f, \gamma_1) &= e^{i\gamma_1 b\sigma_z}. \end{aligned} \tag{D.1}$$

The expectation value is therefore given by

$$\begin{aligned} F_1(\gamma_1, \beta_1) &= -b\langle +|e^{-i\gamma_1 b\sigma_z}e^{i\beta_1\sigma_x}\sigma_z e^{-i\beta_1\sigma_x}e^{i\gamma_1 b\sigma_z}|+\rangle \\ &= -b\langle +|e^{-i\gamma_1 b\sigma_z}\Big(\cos(2\beta_1)\sigma_z + \sin(2\beta_1)\sigma_y\Big)e^{i\gamma_1 b\sigma_z}|+\rangle \\ &= -b\langle +|\cos(2\beta_1) + \sin(2\gamma_1 b) - \sin(2\beta_1)\sin(2\gamma_1 b)\sigma_x|+\rangle \\ &= -b\sin(2\beta_1)\sin(2b\gamma_1), \end{aligned} \tag{D.2}$$

where σ_y represents the Pauli-Y gate. Furthermore, the following relationships are used in (D.2): $\sigma_z\sigma_x = -\sigma_x\sigma_z = i\sigma_y$, $\sigma_x\sigma_y = -\sigma_x\sigma_y = i\sigma_z$ and $\sigma_y\sigma_z = -\sigma_z\sigma_y = i\sigma_x$. Note that the double angle identities of $\cos 2x = \cos^2 x - \sin^2 x$ and $\sin 2x = 2\sin x\cos x$ are employed in (D.2) as well.

Now we consider $p = 1$ and $N = 2$. The problem Hamiltonian then becomes:

$$H_f = A_{1,2}\sigma_z^{(1)}\sigma_z^{(2)} - \sum_{k=1}^{2} b_k\sigma_z^{(k)}, \tag{D.3}$$

From (35c) and (36), we arrive at

$$\begin{aligned} U(H_f) &= U(A_{1,2}, \gamma)U(b_1, \gamma)U(b_2, \gamma), \\ U(H_B) &= U(\beta^{(1)})U(\beta^{(2)}), \end{aligned} \tag{D.4}$$

where $U(A_{1,2}, \gamma) = e^{-iA_{1,2}\gamma\sigma_z^{(1)}\sigma_z^{(2)}}$, $U(b_k, \gamma) = e^{i\gamma b_k\sigma_z^{(k)}}$. Furthermore, $U(\beta^{(k)}) = e^{-i\beta\sigma_x^{(k)}}$ is the unitary operator acting on the k-th qubit. By observing H_f in (D.3), we find that there are two individual components in terms of $A_{1,2}$ and $\{b_1, b_2\}$, respectively. As a result, the expectation values of the two components can be evaluated individually. Let $f_{1,2} = \langle + + |U^\dagger(H_f, \gamma)U^\dagger(H_B, \beta)\sigma_z^{(1)}\sigma_z^{(2)}U(H_B, \beta)U(H_f, \gamma)$ and $g_k = \langle + + |U^\dagger(H_f, \gamma)U^\dagger(H_B, \beta)\sigma_z^{(k)}U(H_B, \beta)U(H_f, \gamma)$. Correspondingly, we have the expectation value F_1 associated with $N = 2$ as follows.

$$F_1 = A_{1,2}f_{1,2} - b_1g_1 - b_2g_2. \tag{D.5}$$

For $N \geq 3$, the expectation value F_1 is the sum of the expectation values associated with the individual components of H_f in (34). Therefore, we have

$$F_1 = \sum_{l>k} A_{k,l}f_{k,l} - \sum_{k}^{N} b_k g_k. \tag{D.6}$$

We first calculate $f_{k,l}$, which is given by

$$\begin{aligned} f_{k,l} =& \langle\psi(0)|U^\dagger(H_f, \gamma)U^\dagger(H_B, \beta)\sigma_z^{(k)}\sigma_z^{(l)}U(H_B, \beta) \\ & U(H_f, \gamma)|\psi(0)\rangle \\ =& \langle +^N|U_N^\dagger(\gamma, \beta)\cdots U_1^\dagger(\gamma, \beta)\sigma_z^{(k)}\sigma_z^{(l)}U_1(\gamma, \beta)\cdots \\ & U_N(\gamma, \beta)|+^N\rangle, \end{aligned} \tag{D.7}$$

where $U_k(\gamma, \beta) = \prod_{l>k} U(A_{k,l}, \gamma)U(b_k, \gamma)U(k, \beta)$. For g_k, all the unitary operators that do not intersect with the Pauli

$$\begin{aligned} H_f &= 2\sigma_z^{(1)}\sigma_z^{(2)} - 2(\sigma_z^{(1)} + \sigma_z^{(2)}) + c' \\ &= \mathrm{diag}([c' + 2A_{1,2} - 2b_1 - 2b_2, c' - 2A_{1,2} - 2b_1 + 2b_2, c' - 2A_{1,2} + 2b_1 - 2b_2, c' + 2A_{1,2} + 2b_1 + 2b_2]). \end{aligned} \tag{C.1}$$

$$\begin{aligned} \tilde{H}(\tau) &= (1-\tau)H_B + \tau H_f \\ &= \begin{pmatrix} \tau(c' + 2A_{1,2} - 2b_1 - 2b_2) & 1-\tau & 1-\tau & 0 \\ 1-\tau & \tau(c' - 2A_{1,2} - 2b_1 + 2b_2) & 0 & 1-\tau \\ 1-\tau & 0 & \tau(c' - 2A_{1,2} + 2b_1 - 2b_2) & 1-\tau \\ 0 & 1-\tau & 1-\tau & \tau(c' + 2A_{1,2} + 2b_1 + 2b_2) \end{pmatrix}. \end{aligned} \tag{C.3}$$

gate σ_z^k commute and do not contribute to g_k, so that the expectation value g_k can be computed as follows.

$$\begin{aligned} g_k =& \langle\psi(0)|U^\dagger(H_f,\gamma)U^\dagger(H_B,\beta)\sigma_z^{(k)}U(H_B,\beta) \\ & U(H_f,\gamma)|\psi(0)\rangle \\ =& \langle +^3|U^\dagger(A_{k',k},\gamma)U^\dagger(A_{k,k''},\gamma)\sigma_z^{(k)}U^\dagger(b_k,\beta)\sigma_x^k \\ & U(b_k,\beta_1)U(A_{k,k''},\gamma)U^\dagger(A_{k',k},\gamma)\sigma_z^{(k)}|+^3\rangle, \end{aligned} \tag{D.8}$$

where $k' < k < k''$.

REFERENCES

[1] 3GPP, "Study on new radio (NR) to support non-terrestrial networks (Release 15)," *Technical Report 38.811 v15.0.0*, 2019.

[2] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, 2013.

[3] M. Di Renzo, A. Zappone, M. Debbah, M.-S. Alouini, C. Yuen, J. de Rosny, and S. Tretyakov, "Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and the road ahead," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2450–2525, 2020.

[4] T. S. Rappaport, Y. Xing, O. Kanhere, S. Ju, A. Madanayake, S. Mandal, A. Alkhateeb, and G. C. Trichopoulos, "Wireless communications and applications above 100 GHz: Opportunities and challenges for 6G and beyond," *IEEE Access*, vol. 7, pp. 78 729–78 757, 2019.

[5] S. Verdú, "Computational complexity of optimum multiuser detection," *Algorithmica*, vol. 4, no. 1, pp. 303–312, 1989.

[6] M. A. Nielsen and I. Chuang, *Quantum computation and quantum information*. American Association of Physics Teachers, 2002.

[7] A. S. Cacciapuoti, M. Caleffi, R. Van Meter, and L. Hanzo, "When entanglement meets classical communications: Quantum teleportation for the quantum internet," *IEEE Trans. Commun.*, vol. 68, no. 6, pp. 3808–3833, 2020.

[8] A. M. Childs, E. Farhi, J. Goldstone, and S. Gutmann, "Finding cliques by quantum adiabatic evolution," *arXiv preprint arXiv: quant-ph/0012104*, 2000. [Online]. Available: https://arxiv.org/abs/quant-ph/0012104

[9] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994, pp. 124–134.

[10] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, 1996, pp. 212–219.

[11] P. Botsinis, S. X. Ng, and L. Hanzo, "Quantum search algorithms, quantum wireless, and a low-complexity maximum likelihood iterative quantum multi-user detector design," *IEEE Access*, vol. 1, pp. 94–122, 2013.

[12] D. Alanis, P. Botsinis, Z. Babar, H. V. Nguyen, D. Chandra, S. X. Ng, and L. Hanzo, "A quantum-search-aided dynamic programming framework for Pareto optimal routing in wireless multihop networks," *IEEE Trans. Commun. Technol.*, vol. 66, no. 8, pp. 3485–3500, 2018.

[13] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, 2018.

[14] J. Gambetta. (2020) IBM's Roadmap For Scaling Quantum Technology. [Online]. Available: https://www.ibm.com/blogs/research/2020/09/ibm-quantum-roadmap/

[15] N. Moll, P. Barkoutsos *et al.*, "Quantum optimization using variational algorithms on near-term quantum devices," *Quantum Science and Technology*, vol. 3, no. 3, p. 030503, Jun 2018.

[16] K. Bharti, A. Cervera-Lierta *et al.*, "Noisy intermediate-scale quantum (NISQ) algorithms," *arXiv preprint arXiv:2101.08448*, 2021. [Online]. Available: https://arxiv.org/abs/2101.08448

[17] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, "A variational eigenvalue solver on a photonic quantum processor," *Nature Communications*, vol. 5, no. 1, p. 4213, 2014.

[18] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, "Parameterized quantum circuits as machine learning models," *Quantum Science and Technology*, vol. 4, no. 4, p. 043001, nov 2019.

[19] M. Cerezo, A. Arrasmith *et al.*, "Variational quantum algorithms," *arXiv preprint arxiv:2012.09265*, 2020.

[20] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv: 1411.4028*, 2014.

[21] E. Farhi and A. W. Harrow, "Quantum supremacy through the quantum approximate optimization algorithm," *arXiv preprint arXiv:1602.07674*, 2019. [Online]. Available: https://arxiv.org/abs/1411.4028

[22] 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*, vol. 12, no. 2, 2019.

[23] E. Farhi, D. Gamarnik, and S. Gutmann, "The quantum approximate optimization algorithm needs to see the whole graph: A typical case," *arXiv preprint arXiv: 2004.09002*, 2020.

[24] M. P. Harrigan, K. J. Sung *et al.*, "Quantum approximate optimization of non-planar graph problems on a planar superconducting processor," *Nature Physics*, vol. 17, p. 332–336, 2021.

[25] S. Lloyd, "Quantum approximate optimization is computationally universal," *arXiv preprint arXiv: 1812.11075*, 2018.

[26] M. E. Morales, J. Biamonte, and Z. Zimborás, "On the universality of the quantum approximate optimization algorithm," *Quantum Information Processing*, vol. 19, no. 9, pp. 1–26, 2020.

[27] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm applied to a bounded occurrence constraint problem," *arXiv preprint arXiv: 1412.6062*, 2015. [Online]. Available: https://arxiv.org/abs/1411.4028

[28] C. Y.-Y. Lin and Y. Zhu, "Performance of QAOA on typical instances of constraint satisfaction problems with bounded degree," *arXiv preprint arXiv:1601.01744*, 2016. [Online]. Available: http://arxiv.org/abs/1601.01744

[29] T. Matsumine, T. Koike-Akino, and Y. Wang, "Channel decoding with quantum approximate optimization algorithm," in *IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 2574–2578.

[30] D. Wecker, M. B. Hastings, and M. Troyer, "Training a quantum optimizer," *Phys. Rev. A*, vol. 94, p. 022309, Aug 2016.

[31] Z. Jiang, E. G. Rieffel, and Z. Wang, "Near-optimal quantum circuit for Grover's unstructured search using a transverse field," *Phys. Rev. A*, vol. 95, p. 062317, Jun 2017.

[32] S. Hadfield, Z. Wang, E. G. Rieffel, B. O'Gorman, D. Venturelli, and R. Biswas, "Quantum approximate optimization with hard and soft constraints," in *Proceedings of the Second International Workshop on Post Moores Era Supercomputing*, ser. PMES'17. New York, NY, USA: Association for Computing Machinery, 2017, p. 15–21.

[33] Z. Wang, S. Hadfield, Z. Jiang, and E. G. Rieffel, "Quantum approximate optimization algorithm for maxcut: A Fermionic view," *Phys. Rev. A*, vol. 97, p. 022304, Feb 2018.

[34] 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, p. 021067, Jun. 2020.

[35] G. E. Crooks, "Performance of the quantum approximate optimization algorithm on the maximum cut problem," *arXiv preprint arXiv: 1811.08419*, 2018.

[36] R. Shaydulin and Y. Alexeev, "Evaluating quantum approximate optimization algorithm: A case study," in *Tenth International Green and Sustainable Computing Conference (IGSC)*, 2019, pp. 1–6.

[37] M. Streif and M. Leib, "Training the quantum approximate optimization algorithm without access to a quantum processing unit," *Quantum Science and Technology*, vol. 5, no. 3, p. 034008, may 2020.

[38] J. Yao, M. Bukov, and L. Lin, "Policy gradient based quantum approximate optimization algorithm," in *Proceedings of The First Mathematical and Scientific Machine Learning Conference, PMLR*, vol. 107, Jul. 2020, pp. 605–634.

[39] I. Hen and M. S. Sarandy, "Driver Hamiltonians for constrained optimization in quantum annealing," *Phys. Rev. A*, vol. 93, p. 062312, Jun 2016.

[40] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, "Quantum computation by adiabatic evolution," *arXiv preprint arXiv: quant-ph/0001106*, 2000. [Online]. Available: https://arxiv.org/abs/quant-ph/0001106

[41] S. Sachdev, "Quantum phase transitions," *Handbook of Magnetism and Advanced Magnetic Materials*, 2007.

[42] S. Bravyi, A. Kliesch, R. Koenig, and E. Tang, "Hybrid quantum-classical algorithms for approximate graph coloring," *arXiv preprint arxiv: 2011.13420*, 2020.

[43] M. Born and V. Fock, "Beweis des Adiabatensatzes," *Zeitschrift fur Physik*, vol. 51, no. 3-4, pp. 165–180, Mar 1928.

[44] S. Aaronson. (2019) Quantum Computing Lecture. [Online]. Available: https://www.cl.cam.ac.uk/teaching/1920/QuantComp/Quantum_Computing_Lecture_15.pdf

[45] D. J. Griffiths and D. F. Schroeter, *Introduction to quantum mechanics*. Cambridge University Press, 2018.

[46] C. C. McGeoch, "Adiabatic quantum computation and quantum annealing: Theory and practice," *Synthesis Lectures on Quantum Computing*, vol. 5, no. 2, pp. 1–93, 2014.

[47] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, "Adiabatic quantum computation is equivalent to standard quantum computation," *SIAM Review*, vol. 50, no. 4, pp. 755–787, 2008.

[48] T. Albash and D. A. Lidar, "Adiabatic quantum computation," *Rev. Mod. Phys.*, vol. 90, p. 015002, Jan 2018.

[49] H. F. Trotter, "On the product of semi-groups of operators," *Proceedings of the American Mathematical Society*, vol. 10, no. 4, pp. 545–551, 1959.

[50] M. Suzuki, "Relationship between d-dimensional quantal spin systems and (d+ 1)-dimensional Ising systems: Equivalence, critical exponents and systematic approximants of the partition function and spin correlations," *Progress of theoretical physics*, vol. 56, no. 5, pp. 1454–1469, 1976.

[51] L.-A. Wu, M. S. Byrd, and D. A. Lidar, "Polynomial-time simulation of pairing models on a quantum computer," *Phys. Rev. Lett.*, vol. 89, p. 057904, Jul. 2002.

[52] S. A. Chin and C. R. Chen, "Gradient symplectic algorithms for solving the schrödinger equation with time-dependent potentials," *The Journal of Chemical Physics*, vol. 117, no. 4, pp. 1409–1415, 2002.

[53] Y. Sun, J.-Y. Zhang, M. S. Byrd, and L.-A. Wu, "Trotterized adiabatic quantum simulation and its application to a simple all-optical system," *New Journal of Physics*, vol. 22, no. 5, p. 053012, may 2020.

[54] M. J. Powell, "A view of algorithms for optimization without derivatives," *Mathematics Today-Bulletin of the Institute of Mathematics and its Applications*, vol. 43, no. 5, pp. 170–174, 2007.

[55] M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, "Experimental realization of any discrete unitary operator," *Phys. Rev. Lett.*, vol. 73, pp. 58–61, Jul 1994.

[56] E. Farhi, D. Gamarnik, and S. Gutmann, "The quantum approximate optimization algorithm needs to see the whole graph: Worst case examples," *arXiv preprint arXiv:2005.08747*, 2020.

[57] G. G. Guerreschi and A. Y. Matsuura, "QAOA for Max-Cut requires hundreds of qubits for quantum speed-up," *Scientific Reports*, vol. 9, no. 1, p. 6903, 2019.

[58] H. Abraham *et al.*, "Qiskit: An open-source framework for quantum computing," doi: 10.5281/zenodo.2562110, 2019.

[59] E. Farhi, J. Goldstone, S. Gutmann, and L. Zhou, "The quantum approximate optimization algorithm and the Sherrington-Kirkpatrick model at infinite size," *arXiv preprint arXiv:1910.08187*, 2019.