Optimal PMU Placement via Quantum Optimization

Yuqi Jiang, *Graduate Student Member, IEEE*, Zhiding Liang, *Member, IEEE*, Yan Li, *Senior Member, IEEE*, and Thomas Morstyn, *Senior Member, IEEE*

***Abstract*— Phasor Measurement Units (PMUs) are essential for real-time monitoring and improving grid observability. Determining the optimal PMU installation is critical, especially as the penetration of renewable energy resources increases, as the Optimal PMU Placement (OPMUP) can guarantee full system observability and reduce the installation cost. However, identifying the optimal PMU installation is a prototypical combinatorial optimization problem, requiring substantial classical computational resources. Furthermore, the significant expansion of the grid in terms of renewable energy integration makes the determination of optimal PMU placement increasingly computationally intensive. In this work, a hybrid quantum-classical approach, Quantum Approximate Optimization Algorithm (QAOA), is developed to effectively solve the optimal PMU installation problem under normal and channel limitation scenarios. A tailored objective function is proposed for quantum optimization, which takes into account both PMU placement cost and system observability constraints. To analyze the observability of QAOA-generated solution distributions, recursion-based Depth-First Search and Breadth-First Search algorithms are proposed for normal and channel-limited scenarios. These methods determine solution feasibility on classical computers with $\mathcal{O}(N+M)$ complexity, outperforming the $\mathcal{O}(N^2)$ complexity of inequality-based approaches, where N and M denote the number of buses and branches, respectively. In addition, the proposed quantum optimization framework can significantly reduce the computational complexity from the polynomial or exponential levels required by previous classical methods, i.e., $\mathcal{O}(RN)$ on quantum circuits and $\mathcal{O}(R(M+N))$ on classical resources, where R is the repetition times of quantum circuits executions. The proposed method is tested on IEEE 9-, 14-, 24-, and 30-bus systems, where better installation results are achieved compared to the state-of-the-art results, providing a new baseline for further quantum studies in OPMUP. Furthermore, a landscape optimization strategy is introduced to improve QAOA solution quality. This approach also reduces the time cost of quantum computing resources, making it more efficient for current quantum applications. Additionally, parameter studies are conducted to identify key factors influencing QAOA performance. This work is expected to lay the foundations for addressing challenging power system problems through quantum technology in the Noisy Intermediate Scale Quantum (NISQ) era.**

***Index Terms*— Quantum Approximation Optimization Algorithm (QAOA), Phasor Measurement Unit (PMU), Optimal Placement, System Observability**

NOMENCLATURE

Sets

$\mathcal{B}$	Bus set with load injection
$\mathcal{C}$	The appearing times of a specific solution string
$\mathcal{I}_j$	PMU-installed bus set, $j = 1, ..., H$
$\mathcal{P}$	Partially observable set
$\mathcal{R}$	Fully observable bus set
$\mathcal{T}$, $\mathcal{F}_n$	Solution set yield by QAOA and feasible solution set, where $\mathcal{F}_n \in \mathcal{T}$
$\mathcal{V}$	Bus set
E	Branch set

Variables

$\mathbf{I}$	Current between 2 buses
$\mathbf{U}$	Voltage phasor
$\mathbf{X}$, x_i	PMU placement binary sequence, i-th binary variable $x_i \in \mathbf{X}$
H	Number of installed PMUs
H_B, H_C	Initilization Hamiltonian, problem Hamiltonian
O	Number of observable buses given $\mathbf{X}$
Y, y_i	Converted binary variables for quantum input

Constants

λ	Penalty coefficients of the objective function
$\mathbf{A}$	Adjacent matrix of G
$\mathbf{F}$	Observability variable of the inequality-based observability modeling method
$\mathbf{Z}$	Impedance matrix
$\mathcal{Y}$	Branch visit indicator matrix
ω	Channel Limitation
σ^x, σ^z	Pauli-X operator, Pauli-Z operator
$\vec{\gamma}$, $\vec{\beta}$	QAOA trainable parameters
D	Number of feasible solutions
G	Graph representation of power grids
M	Number of branches
N	Number of buses
P_1, P_2	Penalty for installing PMU at less important bus, and excessive PMU installing penalty
R	Number of quantum circuits execution repetition times
s	Number of QAOA layers
T	Excessive PMU usage penalty threshold

Functions

$\mathcal{G}$	Observability modeling functions of previous work
$\mathcal{L}(\cdot)$	Output the set of neighbors of the input bus
$\mathcal{M}$	Proposed observability modeling method
$\mathcal{N}(\cdot)$	Output the number of neighbors of the input bus
ψ	Variational wavefunction of QAOA
$C(\cdot)$	Cost function of the binary optimization problem
F	Expectation function of the problem Hamiltonian in the variational state

I. INTRODUCTION

The increasing integration of renewable energy resources into power systems has facilitated decarbonization. However,

This work is supported by the Office of Naval Research under the award N00014-22-1-2504 and the National Science Foundation under the award OAC-2417773.

Y. Jiang and Y. Li are with the Department of Electrical Engineering, Pennsylvania State University, University Park, PA 16802, USA (e-mail: yql5925@psu.edu).

Z. Liang is with the Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, 12180, USA.

T. Morstyn is with the Department of Engineering Science, University of Oxford, Wellington Square, Oxford OX1 2JD, UK.

the intermittent nature of renewable energy such as solar and wind energy introduces significant fluctuations, underscoring the importance of enhanced monitoring and control mechanisms to maintain system stability and efficiency. Managing these fluctuations requires extensive online computational resources, optimal control strategies, and continuous state monitoring to accurately model the rapidly changing electricity generation. Phasor Measurement Units (PMUs) are vital in this regard, offering high-resolution, real-time measurements of current and voltage amplitude and phase in a synchronized manner, providing a comprehensive view of the grid's state.

However, considering the cost of installing PMUs and the vast amount of data they generate and transmit to the communication network, it is neither necessary nor recommended to install PMUs at every bus in the grid, since current and voltage phasors can also be derived using Ohm's Law and Kirchhoff's Current Law (KCL) [1]. Therefore, it is essential to determine the minimum number of PMUs needed to ensure complete observability of the grid.

Generally, the Optimal PMU Placement (OPMUP) problem is a prototypical combinatorial problem: a binary optimization problem. This binary optimization problem is NP-hard, which makes it very complicated to give out exact solutions based on classical computers [2]. To address the issue of OPMUP, quantum computing offers a promising solution. Leveraging the unique properties of quantum mechanics, quantum computing can tackle this complex problem that are insurmountable for classical computers. Recent advancements in quantum hardware and algorithms have inspired various applications in real-world engineering challenges. For instance, the introduction of IBM's 127-qubit quantum computer and their roadmap for 1000-qubit quantum processors in 2021 paved the way for the rapid advancement of quantum computing. In 2023, IBM further advanced this trajectory by releasing the Condor quantum processor with 1,121 qubits, surpassing their previous 1,000-qubit goal and setting a new target of 10,000 qubits [3].

Previous work on OPMUP can be categorized into two main types. The first type includes meta-heuristic methods, which utilize intelligent search techniques to handle non-continuous cost functions. This kind of searching technique involves Simulated Annealing (SA), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Tree Search (TS), etc. In [4], a multistage SA method is proposed, considering the observability constraints during iterations. In [5], an Exponential Binary PSO (EBPSO) is performed, which can generate multiple solutions. Ref [6] develops a Binary PSO (BPSO) integrating a mutation strategy under PMU channel limitations. Ref [7] proposes a Nondominated Sorting GA (NSGA), allowing a trade-off between minimizing the number of PMUs and maximizing observability. In [8], the vanilla GA is enhanced by integrating immune system-inspired mechanisms, avoiding premature stagnation in local optima. Refs [9], [10] utilize the minimal spanning tree search to compute the observability, and Ref [11] utilizes the reinforcement learning based tree search method. Though constructing a single spanning tree can be fast with linear complexity, it only guarantees a locally optimal solution for the chosen spanning tree [12]. Although meta-heuristic methods can adapt to large-scale problems, they can be computationally intensive, especially for very large grids, as they require numerous iterations to converge to a good solution [13]. Moreover, the solutions provided by meta-heuristics are often approximate and may not always be optimal [14].

The second type is deterministic techniques, which convert OPMUP into multiple Integer Linear Programming (ILP) formulations or employ combinatorial searching methods [15]–[17]. Ref [17] conducts an exhaustive combinatorial search to determine the OPMUP solutions with the highest redundancy. Ref [18] utilizes ILP, considering both conventional and power flow measurements. Ref [19] proposes a multiobjective ILP method with heuristic searching techniques. Ref [20] introduces an ILP method considering the realistic costs and the upcoming constraints for modern power systems. Ref [21] introduces the prioritization of the crucial buses in OPMUP using the ILP method. Ref [22] applies ILP to study the OPMUP with channel limitations. Ref [23] proposes a multistage cost-effective placement strategy based on ILP. Ref [24] studies OPMUP based on an equivalent linear formulation of exhaustive search, which can guarantee a global optimum at the cost of exponential complexity. Ref [25] proposes an enumeration-based ILP to incrementally place PMUs to ensure full system observability. Although ILP and combinatorial search methods can effectively handle redundancy constraints, they are also often computationally intensive, typically exhibiting complexities of $\mathcal{O}(N^2)$ or higher [17]–[25].

In OPMUP, determining the observability of a specific placement can be divided into numerical observability and topological observability. Numerical observability is based on the mathematical solvability of the state estimation problem, where continuous input variables are involved in determining whether the Jacobian matrix is full-rank using the measured data from PMU [26]. Topological observability considers the connectivity of the graph and placement of the PMU, ensuring that every bus and branch is sufficiently observable, which is the crucial prerequisite for numerical observability [27]. In most cases, the topological observability and the numerical observability are equivalent [28]. Only under a few scenarios, the topological observability may not guarantee numerical observability [29]. Though topologically observable, if the Jacobian matrix is rank-deficient due to the numerical coincidences of the system parameters, the gain matrix of the state estimation problem is singular, which leads to numerical unobservable [27], [30]. Despite its limitations, topological observability remains essential for power system analysis, as it establishes the basic framework upon which numerical observability must build [27].

To determine the topological observability, Refs [31], [32] utilize the simple fact that the bus placed with a PMU or its neighbors placed with a PMU can be observed. This method can be fast, but it may not model the observability correctly, which will be further discussed in Section II-A. Refs [18], [29], [33]–[36] utilize inequality constraints to determine the observability of buses. Though this method can measure the observability correctly, the number of inequality constraints is $\mathcal{O}(N)$, and it typically requires $\mathcal{O}(N^2)$ complexity. In addition, handling the inequality constraints is a challenge for the current quantum computing algorithms. Currently, to

encode the inequality constraints, at least $\mathcal{O}(N)$ extra qubits are required to encode the inequality constraints [37], [38]. At the current Noisy Intermediate Scale Quantum (NISQ) stage, given that the feasible qubit resources on both the quantum simulators and real quantum computers are relatively few and expensive, it is not economical to embed such inequality constraints in terms of budget and time.

Quantum computing and algorithms have demonstrated impressive advantages in solving real-world problems, especially in optimization. Quantum algorithms take advantage of quantum mechanics to give faster and more optimal solutions in various complicated optimization problems compared with the classical computing method. Such applications can be found in the area of financial portfolio optimization [39], traffic flow optimization [40], drug discovery [41], and power system optimization [42], etc. Ref [43] processes the financial data via quantum RAM and applies the quantum algorithm to speed up the portfolio optimization. Ref [44] develops a Quantum Annealing (QA) strategy to optimize the traffic flow and reduce the number of congested roads compared with the previous baselines. Ref [45] proposes a quantum machine learning method for protein structure discovery, suggesting the potential of quantum hardware as the size of the problem expands. In power systems, recent applications include the hybrid quantum algorithms in power system fault diagnosis [46], unit commitment [47], and quantum neural networks in power flow analysis [48].

As for OPMUP, Ref [31] is the first to apply QA to determine the optimal placement, assuming all buses are zero-injection. In [49], a Quantum Particle Swarm Optimization (QPSO) algorithm is developed and tested on both zero-injection and non-zero injection scenarios. The proposed QA optimization method in [31] demonstrates both superior solution quality and reduced execution time compared to the CPLEX solver. However, the observability modeling method used in this approach cannot reliably ensure accurate observability, which may result in suboptimal solutions for larger problem sizes. Additionally, the tests are only compared against the CPLEX solver, without evaluation against other benchmark methods. In [49], the QPSO method achieves better or equivalent solutions compared to previous benchmarks. However, the high computational complexity of their observability modeling algorithm represents a significant bottleneck for the overall algorithm.

In this work, we develop an end-to-end optimization method based on the Quantum Approximate Optimization Algorithm (QAOA) to solve the OPMUP problem on both normal and channel limitation scenarios. The tailored objective function is designed to minimize the number of installed PMUs while encoding observability constraints by adding penalties to solutions that result in unobservable outcomes. To determine observability, we employ efficient modeling algorithms with $\mathcal{O}(M+N)$ complexity for both normal and channel limitation scenarios, significantly improving upon the $\mathcal{O}(N^2)$ complexity of traditional methods while eliminating the need for extra qubits compared with the ILP method. This enables the proposed QAOA optimization approach to rapidly assess the feasibility of generated solutions. In this study, only PMU measurements are considered, excluding conventional and power flow measurements. The proposed approach uses both quantum and classical resources of $\mathcal{O}(RN)$ and $\mathcal{O}(R(N+M))$, demonstrating theoretical speedup from exponential or polynomial. In terms of outcomes, it achieves better results compared to benchmark models on IEEE 9-, 14-, 24-, and 30-bus systems when using quantum optimization on both normal and channel limitation scenarios. Generally, the main contributions and novelties can be summarized as:

1) This paper presents the first end-to-end QAOA optimization approach for OPMUP, integrating a tailored objective function that accounts for both PMU installation and observability constraints across normal and channel limitation scenarios. To ensure precise observability, recursion-based Depth-First Search (DFS) and Breadth-First Search (BFS) algorithms are proposed for normal and channel limitation situations, respectively, achieving $\mathcal{O}(N+M)$ complexity—outperforming conventional $\mathcal{O}(N^2)$ inequality methods. Notably, the proposed methods require no extra qubits compared with ILP, making them more practical for NISQ-era applications. As for the outcome, the first QAOA optimization baseline for OPMUP across both normal and channel limitation scenarios is validated on real quantum computers and quantum simulators. The results exceed benchmark methods on both normal and channel limitation situations, demonstrating the potential of quantum optimization to surpass classical approaches, especially with the ongoing advancements in quantum hardware.
2) The developed QAOA optimization method for OPMUP demonstrates the theoretical speedup compared with the previous baselines from exponential or polynomial complexity to $\mathcal{O}(RN)$ on quantum resources and $\mathcal{O}(R(N+M))$ on classical computers, thereby enhancing scalability and efficiency for large-scale optimization problems in the future.
3) This paper introduces a landscape optimization strategy for QAOA, incorporating penalty terms as classical post-processing adjustments to the quantum energy function without adding computational complexity. It enhances the quality of the solution set while also reducing implementation time on quantum computing resources, providing a more practical and cost-effective approach for NISQ-era applications.
4) The parameter studies analyze key factors affecting QAOA performance, providing insights into optimizing the number of QAOA layers based on the problem scale. This guidance enhances the adaptability and efficiency of QAOA in solving large-scale optimization problems.

The remainder of this paper is organized as follows: Section II formulates the problems and the corresponding objective function, Section III introduces the detailed QAOA optimization solutions, Section IV shows the numerical case studies and conclusions are drawn in Section V.

II. Problem Formulation

Assume a grid system as $G=(\mathcal{V},E)$ where $\mathcal{V}$ denotes the bus set (node) and E is the edge set (branch). The number

of buses is N and the number of edges is M. The adjacency matrix for this graph G is $\mathbf{A}$. For further analysis, an adjacency list function $\mathcal{L}$ is also defined, where $\mathcal{L}(\cdot)$ will output the set of neighbors of the input node. Therefore, a function $\mathcal{N}(\cdot)$ can also be adapted to count the total number of the adjacent neighbors of a specific node. We assume that the line impedances of the branches are known and denote the impedance matrix as $\mathbf{Z}$.

In this paper, we consider both the zero-injection and non-zero injection scenarios. In addition, the PMU channel limitation scenario is also studied. Assume that all the injections are in the form of current and need to be measured by a PMU or calculated via KCL. In addition, only the PMU measurement is considered, while conventional measurement and power flow measurement are not involved. In this paper, it is also assumed that the reliability of PMUs is guaranteed and all PMUs can function well. The overall process of the developed method is illustrated in Fig. 1. In the next section, the mathematical modeling of the power grid will be demonstrated.

A. Objective Function for OPMUP

For the simplicity of formulation, we use the zero-injection scenarios to illustrate the proposed method, and the non-zero injection scenario can be trivially extended. The main task of the OPMUP is to determine the minimum number of PMUs that must be placed on certain buses to ensure full observability of the system. In this paper, the full observability is defined as follows:

Definition 1 (Fully Observable)**.** *A bus is called fully observable if its voltage phasor and all the currents on its connected branches are known.*

To determine the observability of a bus, the following rules can be utilized:

(i) A PMU placed at a specific bus, which can measure the voltage phasor of this bus and all the currents on the connected branches.
(ii) The branch current can be calculated by Ohm's Law if the voltage phasor at both ends of that branch is known.
(iii) If the voltage phasor at one end of the branch and the current on the branch are known, the voltage phasor at the other end can be calculated via Ohm's Law.
(iv) If a bus i is connected to $\mathcal{N}(i)$ branches and $(\mathcal{N}(i)-1)$ branch currents are known, the last unknown current can be calculated via KCL.

Therefore, assume $\mathbf{X} \in \mathbb{R}^{N\times 1}$ is the binary sequence that determines the placement of the PMU installation, $x_i \in \mathbf{X}$ represents the i-th binary variable for the placement of the PMU on the bus i, where $i \in N$. If $x_i = 1$, then a PMU will be installed on bus i, and the opposite for $x_i = 0$. O is the number of observable buses in a specific placement $\mathbf{X}$. As a result, the objective function can be expressed as:

$$\begin{aligned} &\min \sum_{i=1}^{N} x_i \\ &\text{s.t. } O = N. \end{aligned} \tag{1}$$

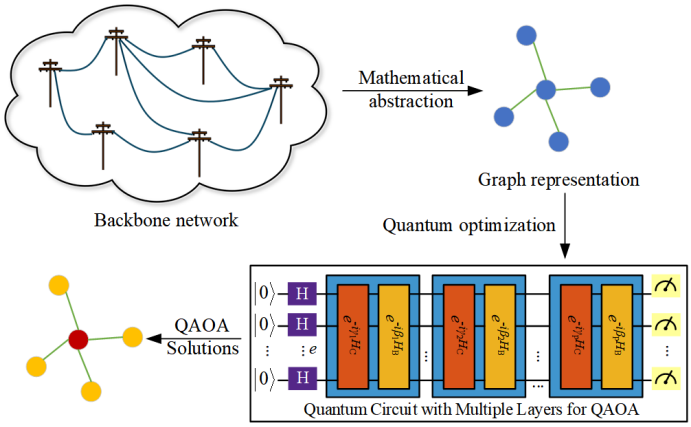

Fig. 1. The overall quantum optimization process for OPMUP.

We can further write this function in the form:

$$\min \sum_{i=1}^{N} x_i + \lambda(N - O), \tag{2}$$

where λ is the penalty coefficient. In the numerical examples, for IEEE 9-, 14-, 24-, and 30-bus systems, hyperparameter λ is set to 20, 50, 100, and 300, respectively. If the system is not fully observable under a specific placement, the penalty λ will be incurred for this placement sequence. Obviously, less observability will lead to a larger penalty. In most PMU placement studies, partially observable buses often appear, which also appear in this study. In this paper, the rigorous definition of partially observable can be expressed as:

Definition 2 (Partially Observable)**.** *A bus is called partially observable if its voltage phasor is known.*

In this paper, only fully observable buses will be considered as observable ones. Finding the optimal solution to this objective function requires substantial computational resources. Classical combinatorial search requires exponential computational power to identify global optimal solutions, while most ILP methods, with a complexity of $\mathcal{O}(poly(N))$, are typically only capable of finding suboptimal solutions. Due to limited computational resources, most studies have adopted ILP-based methods with similar objective functions. In the following Section II-B.1, we will discuss their mathematical formulations and inherent limitations. Afterward, the proposed observability modeling algorithm and the quantum solutions will be illustrated in Section II-B.2 and Section III-A.

B. Observability Modeling Algorithm

1) Previous Observability Modeling Method $\mathcal{G}$: To determine observability under a specific PMU installation, based on the review of the previous paper, the observability modeling method can be categorized into 2 kinds (denoted as $\mathcal{G}_t, t = 1, 2$). $\mathcal{G}_1$ uses the trivial fact that the bus neighboring the PMU-installed bus is fully observable [31], [32]. However, $\mathcal{G}_1$ is not accurate in identifying all observable buses. The IEEE 9-bus system is used as an illustrative example in Appendix I, demonstrating that under a specific placement, $\mathcal{G}_1$ incorrectly

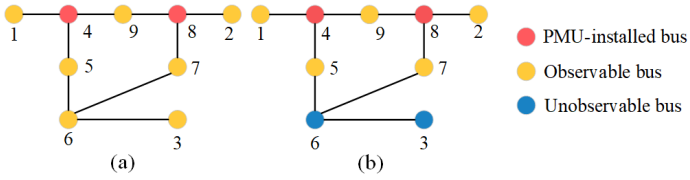

Fig. 2. Observability visualization for IEEE 9-bus system. In (a), the PMUs are placed in buses 4 and 8. Based on rule (i)-(iv) ($\mathcal{M}_1$), all the buses can be observable. In (b), according to the observability method $\mathcal{G}_1$, only buses 1, 2, 4, 5, 7, 8, and 9 can be observed.

suggests that the system is not fully observable, whereas it is, in fact, fully observable. The outcomes of $\mathcal{G}_1$ and the proposed method are shown in Fig. 2 (a). Thus, it can be inferred that $\mathcal{G}_1$ tends to include fewer observable buses and does not guarantee the precise computation of the observability of the power grid.

$\mathcal{G}_2$ applies the inequality constraints to determine the observability conditions [18], [29], [33]–[36], and typically $\mathcal{O}(N)$ inequality constraints are set up to model the precise observability. To ensure the system is fully observable and minimize the number of PMU installations, the objective function can be formulated as:

$$\begin{aligned} &\min \sum_{i=1}^{N} x_i \\ &\text{s.t. } [\mathbf{F}]_{N\times 1} = [\mathbf{A}]_{N\times N} * [\mathbf{X}]_{N\times 1} \geq [\mathbb{1}]_{N\times 1}, \end{aligned} \tag{3}$$

where $\mathbf{F} \in \mathbb{R}^{N\times 1}$ is the observability variable, and the overall number of inequality constraints is $\mathcal{O}(N)$. Although several of the inequality constraints can be canceled during the processing, the number of constraints can still be approximated as $\mathcal{O}(N)$. In this optimization constraint, $\mathbf{F}$ indicates that each bus requires a PMU placed on that bus or buses connected to it through transmission lines, and $\mathbb{1}$ is a vector with all elements 1. Thus, only if $\mathbf{F} \geq \mathbb{1}$, the full observability of the system can be satisfied. Regarding the complexity of the computation, for each placement $\mathbf{X}$, the check of the observability of the inequality is quadratic with the number of buses N ($\mathcal{O}(N^2)$). In addition, to handle the $\mathcal{O}(N)$ inequality constraints, Ref [38] illustrates that to process one inequality constraint in the form of: $\sum_{i=1}^{N} x_i \leq \theta$, $\mathcal{O}(\theta N)$ extra qubits are needed. Thus, given the constraints $\mathcal{O}(N)$, the number of additional qubits required can be approximated as $\mathcal{O}(N^2)$. Given the limitations of hardware resources, it is not economically feasible to adopt $\mathcal{G}_2$ for quantum optimization methods.

Empirically, to handle the inequality constraints, even if Ref [31] utilizes $\mathcal{G}_1$ to largely simplify the observability constraints, for the IEEE 9-bus system, another 9 qubits are required. This is even worse for the IEEE 14-bus system, where 21 extra qubits are demanded. Thus, it is not recommended to adopt the $\mathcal{G}_2$ to the quantum algorithm for OPUMP.

2) Proposed method: Given the above limitations of $\mathcal{G}_t$, we propose a graph-based recursion searching method $\mathcal{M}_1$. In terms of KCL, we take advantage of rule (iv) to design the recursion scheme. The recursive base is the initial PMU placement sequence, where the node installed with a PMU can be fully observed. The recursive target is to find all fully observable nodes starting with the initial placement recursively through rules (i) to (iv).

Suppose that the H PMUs are installed in terms of the initialization set $\mathcal{I}_j, j = 1, 2, ..., H$, and these buses placed with PMUs will be fully observed. A fully observable bus set can be denoted as $\mathcal{R}$, where the initial placement from $\mathcal{I}$ is used to initialize $\mathcal{R}$. For each node $j, j \in \mathcal{I}$ in the initialization set, its neighbors $p \in \mathcal{L}(j)$ will be at least partially observed based on the rule (iii) where we can calculate the voltage phasor of p as $\mathbf{U}_p = \mathbf{U}_j - \mathbf{Z}_{jp}\mathbf{I}_{jp}$, where $\mathbf{Z}_{jp}$ and $\mathbf{I}_{jp}$ can be obtained from the line impedance matrix $\mathbf{Z}$ and the output of PMU placed at j. Since $\mathbf{U}_p$ and the current of one of its connected branches is known ($\mathbf{I}_{jp}$), p can be called a partially observed node. During the recursive process, any fully observable bus will have at least one neighbor that is at least partially observable. In this paper, the partially observed nodes are treated as not fully observable ones in our quantum algorithm iterations. One can also prove that the partially observable node must satisfy the following:

Lemma 1 (Partially Observable Lemma)**.** *For any partially observable bus, at least one current in its connected branch is known.*

The proof of Lemma 1 is given in Appendix II. Based on this observation, the recursion can be designed as: beginning with an observable node $j \in \mathcal{I}$, since all nodes in the neighbor set $\mathcal{L}(j)$ are partially observable (the voltage phasor and one current connecting j), we can use rule (ii)-(iv) to determine whether any of the nodes $p \in \mathcal{L}(j)$ can be fully observable. Since p has $\mathcal{N}(p)$ neighbors and according to rule (iv), it is required to obtain the current of its $(\mathcal{N}(p) - 1)$ connected branches to make p fully observable.

A trivial case is that p has only one neighbor, which is j. Since j is placed with a PMU, then j is observed. Another trivial case is that if p has two neighbors, then p can be directly observed and added to $\mathcal{R}$ ($\mathbf{I}_{jp}$ is known and applies rule (iv) to compute the other current). Afterward, p will be processed in this recursive manner in a similar way. According to Lemma 1, since the voltage phasor of the partially observable bus is known, we only need to check the branch current to determine whether the partially observable bus can be observed during the recursion process. During recursion, the processed buses that are determined to be fully observable will also go into the recursion process immediately, while the partially observable nodes will be stored until they can be determined as observable.

The other case is that p has multiple neighbors, and since only one neighbor of p is known (j), p is partially observed. Thus, the recursion process starting from j is terminated, and the algorithm will try to search in $\mathcal{I}$ to see if there are other observable nodes that have not been into the recursion yet. As previously mentioned, a partially observable set denoted as $\mathcal{P}(\cdot)$ will be utilized to memorize these partially observable nodes and their corresponding known current during the recursion, where only if the number of known current exceeds $(\mathcal{N}(p) - 1)$, p can be fully observable. During the recursion, if $\mathcal{P}(p) \geq (\mathcal{N}(p) - 1)$ occurs, p can be observed and added to $\mathcal{R}$. As a result, p will enter the recursion process and will be removed from the set $\mathcal{P}$.

In general, starting with the initial placement $\mathcal{I}$, it can be recursively computed to find the potential observable neighbors of $j \in \mathcal{I}$ and the recursion will stop if all nodes that can be observed are traversed. Given that the adjacency matrix $\mathbf{A}$ and the bus set $\mathcal{V}$ are constant when the grid system is fixed, only the placement $\mathbf{X}$ is the input variable. Thus, in the zero-injection scenario, we can use $\mathcal{M}_1$ to represent this Algorithm 1 and the pseudocode can be summarized in Algorithm 1:

Algorithm 1: Proposed observability modeling algorithm: zero-injection

Input : Adjacency matrix $\mathbf{A}$, placement sequence $\mathbf{X}$, bus set $\mathcal{V}$

Output: Total number of observed buses O

```
CalObservability(A, X, V):
// Initialize placement and observation sets
I ← Initial placement based on X;
R ← I;
P ← {};
// Calculate neighbors counter for each node in V
Q ← N(V) ;        // Q ∈ R^{N×1}, counter of neighbors for all the nodes in V
for j ∈ I do
    TryObservedNode(j);
end for
O ← length of R;
return O;

TryObservedNode(j):
for p ∈ L(j) do
    if p ∉ R then
        if p ∉ P then
            // Add p to the potential observation set P if not already included
            P.add(p);
        end if
        // Increase the counter for potential observation
        P[p] ← P[p] + 1;
        if P[p] ≥ Q[p] − 1 then
            R.add(p);
            // Remove p from potential observation list
            delete P[p];
            TryObservedNode(p);
        end if
    end if
end for
```

$\mathcal{M}_1$ will take the adjacency matrix $\mathbf{A}$, given placement sequence $\mathbf{X}$, and the bus set $\mathcal{V}$ as input, and output will yield the total number of observed buses O.

As illustrated in Fig. 2, assume that two PMUs are installed on the bus 4 and 8 initially. Thus, $\mathcal{I}$ and $\mathcal{R}$ can be initialized as $\{4, 8\}$. Starting with buses 4, buses 1, 5, and 9 can be partially observed and added to: $\mathcal{P}[1] = 1, \mathcal{P}[5] = 1$, and $\mathcal{P}[9] = 2$. Apparently, it can be calculated that:

$$\mathcal{P}[1] = 1 \geq \mathcal{N}(1) - 1 = 0, \tag{4}$$

$$\mathcal{P}[5] = 1 \geq \mathcal{N}(5) - 1 = 1, \tag{5}$$

$$\mathcal{P}[9] = 2 \geq \mathcal{N}(9) - 1 = 1. \tag{6}$$

Therefore, bus 1, 5, and 9 will be added to $\mathcal{R} = \{1, 4, 5, 8, 9\}$ and eliminated from $\mathcal{P}$. Since buses 1, 5, and 9 are observed, they will be implemented with recursion, respectively. Buses 1 and 9 have no neighbors to do the recursion since buses 4 and 8 are installed with PMUs already. The recursion on bus 5 will make bus 6 partially observable and $\mathcal{P}[6] = 1$. Since bus 6 has 3 neighbors, the recursion starting from 4 is end. Then bus 8 will be implemented with recursion, resulting in bus 7 being observable since it only has 2 neighbors. Thus, it can be updated that $\mathcal{P}[6] = 2 \geq \mathcal{N}(6) - 1 = 2$ and bus 6 is observable. Lastly, bus 3 can be observable following the recursion, similarly.

3) Observability Modeling for Non-Zero Injection: Similar to the zero-injection scenario, the observability modeling of the non-zero injection can be trivially extended as $\mathcal{M}_1^*$. Assume the bus set with the load as $\mathcal{B}$, if each bus $q \in \mathcal{B}$ is fully observable, then $(\mathcal{N}(q) + 1)$ should be either calculated or measured since the injection current exists. In this case, if a bus with load cannot be directly measured by an installed PMU, then all of its connected branches (apart from the load injection) should be calculated in order to apply rule (iv) to compute the injection current. The observability modeling under the non-zero injection scenario can be described as $O = \mathcal{M}_1^*(\mathbf{A}, \mathbf{X}, \mathcal{V}, \mathcal{B})$.

4) Observability Modeling Considering PMU Channel Limitations: The cost of PMU can heavily rely on the technical configurations, especially the number of channels [50]. Given the high cost of unlimited PMU measurement capabilities, the OPMUP problem is examined under the more realistic constraint of limited available measurement channels [51].

In previous work, assuming a bus i is placed with a PMU and the channel limitation is ω, another binary decision variable is involved in determining whether the PMU channel is installed at a branch $ip, p \in \mathcal{L}(i)$. However, for quantum optimization, this will increase $\mathcal{O}(\omega H)$ qubits involved in the QAOA optimization (H is the number of installed PMUs). In the NISQ era, due to the limited quantum resources, especially the available qubits in both the quantum simulator and the quantum computers, it is not economical to increase more binary variables for QAOA. Therefore, we first introduce the following two methods as a proof of concept and for quantum demonstration, providing a foundational approach to explore the feasibility and effectiveness of QAOA in solving the problem of OPMUP within current quantum hardware constraints.

For this task, 2 BFS-based greedy search algorithms $\mathcal{M}_2$ and $\mathcal{M}_3$ are proposed, both utilizing a recursive approach inspired by the recursion strategy from $\mathcal{M}_1$, where they use greedy and random channel selection strategies respectively. Specifically, for $\mathcal{M}_2$, when a bus is placed with a PMU, the measurement channel will be installed on the branches, where the bus on the other end has more neighbors. Since QAOA can generate a more abundant solution set than the classical method, it is very likely for QAOA to pinpoint the global optimum. In $\mathcal{M}_3$, the measurement channel will be randomly selected to install on the connecting branches of a PMU-installed bus. Since quantum optimization can leverage the randomness in generating a large solution space and process-

ing it in parallel [52], QAOA may have the capability to handle the randomness of the measurement channel selection as well. Empirical results from numerical examples demonstrate that the proposed $\mathcal{M}_2$ and $\mathcal{M}_3$ can guarantee better or equivalent results compared with the previous.

For both $\mathcal{M}_2$ and $\mathcal{M}_3$, assume H PMUs are installed, and a queue $\mathbf{Q}$ is initialized with each PMU-installed bus $j, j = 1, ..., H$. If a PMU is placed at bus j and the number of its neighbors satisfies $\mathcal{N}(j) \leq \omega + 1$, then bus j becomes fully observable, and each of its neighboring buses $p \in \mathcal{N}(j)$ will be marked as visited with their observability count updated as $\mathcal{P}(p) = \mathcal{P}(p)+1$. If $\mathcal{P}(p) \geq \mathcal{N}(p)-1$, p will enter the queue. After all $p \in \mathcal{N}(j)$ is processed, j will be out of the queue, the branch jq will not be visited in the following recursion (A symmetric matrix $\mathcal{Y}$ is utilized, where the entry $\mathcal{Y}[jq] = \mathcal{Y}[qj] = 1$ if jq is visited, otherwise $\mathcal{Y}[jq] = \mathcal{Y}[qj] = 0$), and such j will not enter the queue again.

The other case is that $\mathcal{N}(j) > \omega + 1$, and the observability of bus j is not determined. The measurement channel will be selected based on different strategies. In $\mathcal{M}_2$, the measurement channel will be installed on the branches $jp, p \in \mathcal{N}(j)$ where: (1), jp has not been visited ($\mathcal{Y}[jp] = 0$), (2), $p \in \mathcal{N}(j)$ has more neighbors. If the number of unvisited branches of j is less than ω, there will be measurement channel redundancy at j, and these redundant channels will still be placed at p with more neighbors, where $p \in \mathcal{N}(j)$ and $\mathcal{Y}[jp] = 1$. Thus, after the updating of the channel placement for j, the total known branches of j can be counted as $\mathcal{P} = \sum_j \mathcal{Y}[jp], p \in \mathcal{N}(j)$. If j is determined to be fully observable, j will enter $\mathbf{Q}$ again. Otherwise, j is partially observable. For $\mathcal{M}_3$, the measurement channel and the redundant channel will be randomly selected, while the other recursion procedures are the same as $\mathcal{M}_2$. In both methods, if any PMU-installed bus j is partially observable during its first iteration, but is further determined to be fully observable in the following iterations, j will enter $\mathbf{Q}$ again. During the iterations of the PMU-installed buses, though they may be determined to be fully observable, some of their neighbors may not be visited. Therefore, if the PMU-installed buses are determined to be fully observable during the recursion, they will always enter $\mathbf{Q}$ twice to ensure there are no unvisited neighbors.

Apart from the PMU-installed buses, for both $\mathcal{M}_2$ and $\mathcal{M}_3$, those who enter $\mathbf{Q}$ are all fully observable, and all of their neighbors will be processed. Thus, they will not reenter $\mathbf{Q}$ and will only be iterated once. The overall algorithms of $\mathcal{M}_2$ and $\mathcal{M}_3$ are illustrated in Appendix IV.

5) Complexity analysis of the proposed Observability modeling algorithm: As for the computation complexity analysis of a placement $\mathbf{X}$, for $\mathcal{M}_1$, in the recursion part, if the graph G is fully observable, every bus i will only enter the recursion process once, resulting in $\mathcal{O}(N)$ complexity. For every branch in the fully observable situation, it will be checked at most twice, yielding a $\mathcal{O}(2M)$ complexity. Thus, the total complexity is $\mathcal{O}(M + N)$. As for $\mathcal{M}_2$ and $\mathcal{M}_3$, if all the buses are fully observable (which is the worst case for the algorithm), all the branches will be only visited once, the PMU-installed buses will be iterated twice, and the remaining buses will be iterated once, resulting in $\mathcal{O}(M+N)$ complexity. Generally, the proposed observability modeling method can cost fewer computational resources than $\mathcal{G}_2$ of $\mathcal{O}(N^2)$. In the next section, we will introduce the quantum optimization method to solve the objective function proposed.

III. Quantum optimization method

In this section, the QAOA solution strategy will be developed to solve the OPMUP objective function, as shown in Fig. 1. For simplicity of demonstration, $\mathcal{M}$ is applied to denote the observability modeling regardless of the zero, non-zero injection, or channel limitation scenarios. Empowered by the quantum principle of superposition, QAOA can solve the combinatorial optimization problem more efficiently by exploring multiple solutions in the large searching space simultaneously, which is typically suitable for OPMUP.

A. QAOA for OPMUP

In quantum circuits, qubits are used as the basic data structure, and each binary variable $x_i \in \mathbf{X}$ is encoded into the qubit form. However, the binary data input required for QAOA is $\{-1, +1\}$. Thus, an input transformation is necessary to convert the $\{0, 1\}$ binary string to $\{-1, +1\}$ string Y:

$$y_i = -2x_i + 1, \tag{7}$$

where $Y = [y_1, y_2, ..., y_i, ..., y_N], y_i \in \{-1, +1\}$. Thus, the original objective function can be rewritten as:

$$\min C(Y) = -\sum_{i=1}^{N}(y_i - 1)/2 + \lambda(N - \mathcal{M}(-\frac{Y-1}{2})). \tag{8}$$

On quantum computers, an N qubits string $|Y\rangle = |y_1...y_i...y_N\rangle$ will denote the state of placement. Each qubit $|y_i\rangle$ is the superposition of the quantum states $|0\rangle$ and $|1\rangle$, where $|y_i\rangle = a_i|0\rangle + b_i|1\rangle$, a_i and b_i are the complex probability amplitudes associated with $|0\rangle$ and $|1\rangle$, and $|a_i|^2 + |b_i|^2 = 1$. $|0\rangle$ and $|1\rangle$ are the eigenstates of the Pauli-Z operators σ^z with the eigenvalues of 1 and -1, respectively. As a result, we could further measure $|0\rangle$ to represent 1 and measure $|1\rangle$ to represent -1, which can be executed on quantum circuits, validating the motivation of the input transformation in Eq. (7).

The main task of QAOA is to find the minimum expectation value of the energy state described by the problem Hamiltonian H_C. The first step of QAOA is to convert the classical objective function into a problem Hamiltonian $H_C = C(\sigma_1^z, \sigma_2^z, ..., \sigma_i^z, ..., \sigma_N^z)$. To encode the objective function into the QAOA form explicitly, a problem Hamiltonian H_C will be constructed as:

$$H_A = -\sum_{i=1}^{N}\frac{\sigma_i^z - 1}{2}, \tag{9}$$

$$H_{A'} = \lambda(N - \mathcal{M}(cat(-\frac{\sigma_i^z - 1}{2}))), \tag{10}$$

$$H_C = H_A + H_{A'}, \tag{11}$$

where $cat(\cdot)$ is the function that concatenates the input variables into a string (concatenate $\{\sigma^z\}_1^N$ into $[\sigma_1^z, ..., \sigma_i^z, ..., \sigma_N^z]$). The input for $\mathcal{M}(\cdot)$ will take the concatenated and converted strings ($\{0, 1\}$ binary) to calculate

the observability modeling. Thus, we can have $|Y\rangle$ as the mapping of Y to quantum circuits. For quantum computing, the expectation value of the energy associated with the state $|Y\rangle$ reflects the quality of the solution, and the objective is to minimize this expectation to achieve optimal solutions. This expectation value, described by the problem Hamiltonian H_C, for a given $|Y\rangle$ is expressed as $C(|Y\rangle) = \langle Y| H_C |Y\rangle$.

Furthermore, we assume the deterministic string Y_k out of the 2^N N-qubit candidate strings on the computational basis, where each $|y_{k,i}\rangle = |0\rangle$ or $|1\rangle$, $y_{k,i} = \langle y_{k,i}| \sigma_i^z |y_{k,i}\rangle$, and $Y_k = [y_{k,1}, ..., y_{k,i}, ..., y_{k,N}]$. To map $C(Y_k)$ to $C(|Y_k\rangle)$:

$$C(Y_k) = -\sum_{i=1}^{N} \frac{y_{k,i} - 1}{2} + \lambda(N - \mathcal{M}(-\frac{Y_k - 1}{2})) \tag{12}$$

$$= -\sum_{i=1}^{N} \frac{\langle y_{k,i}| \sigma_i^z |y_{k,i}\rangle - 1}{2} + \lambda(N - \mathcal{M}(-\frac{Y_k - 1}{2})) \tag{13}$$

$$= -\sum_{i=1}^{N} \frac{\langle y_{k,i}| \sigma_i^z |y_{k,i}\rangle - 1}{2}$$

$$+ \lambda(N - \mathcal{M}(cat(-\frac{\langle y_{k,i}| \sigma_i^z |y_{k,i}\rangle - 1}{2}))) \tag{14}$$

$$= \langle Y_k| H_c |Y_k\rangle \equiv C(|Y_k\rangle) \tag{15}$$

As a result, we have mapped the 2^N classical cost function $C(Y_k)$ to 2^N quantum cost function $C(|Y_k\rangle)$. Then 2^N $|Y_k\rangle$ forms the complete basis of the N-bit Hilbert space for the N-bit quantum states. In that case, $|Y\rangle$ can be expressed as the linear combination of $|Y_k\rangle$: $|Y\rangle = \sum_{k=1}^{2^N} \alpha_k |Y_k\rangle$, where $\sum_{k=1}^{2^N} |\alpha_k|^2 = 1$. We can further have:

$$C(|Y\rangle) = \langle Y| H_C |Y\rangle = (\sum_{k=1}^{2^N} \alpha_k \langle Y_k|) H_C (\sum_{k=1}^{2^N} \alpha_k |Y_k\rangle) \tag{16}$$

$$= \sum_{k=1}^{2^N} \alpha_k^2 \langle Y_k| H_C |Y_k\rangle \tag{17}$$

$$= \sum_{k=1}^{2^N} \alpha_k^2 C(|Y_k\rangle) \tag{18}$$

And now the OPMUP problem is converted into the problem of finding the expectation value of the minimum energy state described by the obtained Hamiltonian problem H_C.

Thus, we have built the overall forward process. To initialize the quantum circuit, the initialization Hamiltonian H_B creates a superposition of all N quantum states. The most common approach is to construct H_B as a sum of Pauli-X operators acting on each of the N qubits as $H_B = \sum_{i=1}^{N} \sigma_i^x$ [53].

Therefore, all N qubits are initialized to $|+\rangle^{\otimes N}$ states. By applying the problem Hamiltonian H_C and the initialization Hamiltonian H_B, the variational wavefunction of QAOA can be generated as:

$$|\psi_s(\vec{\gamma}, \vec{\beta})\rangle = e^{-i\beta_s H_B} e^{-i\gamma_s H_C} ... e^{-i\beta_1 H_B} e^{-i\gamma_1 H_C} |+\rangle^{\otimes N}, \tag{19}$$

where the variational wavefunction can be parameterized by $2s$ trainable parameters $\vec{\gamma} = (\gamma_1, \gamma_2, ..., \gamma_s)$ and $\vec{\beta} = (\beta_1, \beta_2, ..., \beta_s)$. The number of parameters s also indicates that the circuit has s layers. Finally, we obtain the expectations of the problem Hamiltonian H_C in the variational state as:

$$F_s(\vec{\gamma}, \vec{\beta}) = \langle \psi_s(\vec{\gamma}, \vec{\beta})| H_C |\psi_s(\vec{\gamma}, \vec{\beta})\rangle . \tag{20}$$

To update the trainable parameters $\vec{\gamma}$ and $\vec{\beta}$, multiple approaches can be applied on classical computers to search for the optimal $\langle \vec{\gamma}^*, \vec{\beta}^* \rangle$ to minimize $F_s(\vec{\gamma}, \vec{\beta})$. A common method is COBYLA, which is a derivative-free optimization approach without the requirements for smooth objective functions [54].

As the power grid size grows, with more buses and branches added, the increasingly complex landscape of the solution space may constrain QAOA's ability to obtain the global optimum. Therefore, selecting the appropriate number of QAOA layers s is essential for the success of QAOA. Additionally, increasing the number of QAOA layers s can also boost the possibility of success.

Regarding QAOA implementation, Qiskit is often utilized to simulate quantum circuits [55]. The simulated quantum circuit will be run repeatedly R times and generate a subset of the solution distribution $\mathcal{T}$ from the candidate strings 2^N. Unlike the traditional optimization method, the quantum optimization method will yield the distribution of the solution to the problem. Yet we can still sample the solution strings from the distribution that produce the least PMU installations. Within the subset, solution strings can appear multiple times, with a higher frequency of occurrence indicating a more probable solution. Assume that there are D feasible solution strings (that can make the system fully observable) within the feasible solution set $\mathcal{F} \in \mathcal{T}$. For each feasible solution string $\mathcal{F}_n, n = 1, ..., D$, the corresponding appearing times are denoted as $\mathcal{C}_n$, and the corresponding probability can be formulated as $P(\mathcal{F}_n) = \frac{\mathcal{C}_n}{\sum_{m=1}^{D} \mathcal{C}_m}$.

B. Complexity Analysis

The complexity of the overall process comes from both quantum circuits and classical computers. The cost of quantum computation involves the initialization of the quantum state and the implementation of the QAOA layer. On classical computers, the observability modeling algorithm $\mathcal{M}$ will take the strings generated by quantum circuits to compute the observability. Besides, the parameter updating of QAOA is also conducted on classical computers via COBYLA, and the number of iteration steps is represented as u, where u is approximated as $\mathcal{O}(poly(s))$ via empirical simulations for heuristic parameter optimization method [53]. In this study, simulations of the COBYLA iteration steps k are also conducted, suggesting a $k = \mathcal{O}(poly(s))$ relation. The desired convergence precision ratio of QAOA can be described as: $\frac{C(\mathbf{X})}{C_{SOTA}} \geq \epsilon^*$, where the target is to optimize $\epsilon^* \geq 1$, indicating QAOA can surpass the performance of the SOTA.

For one iteration, given the placement $\mathbf{X}$, the initial state preparation will cost N Hadamard gates to create a superposition of all possible states, which is at an $\mathcal{O}(N)$ cost. As for each QAOA layer, the initialization Hamiltonian H_B requires the number of gates equal to the qubits, thus $\mathcal{O}(N)$ gates are involved. The number of gates to implement the

problem Hamiltonian H_C is divided into two parts (H_A and $H_{A'}$ respectively). The implementation of H_A will use $\mathcal{O}(N)$ single-qubit to do the summation. For $H_{A'}$, the observability modeling algorithm will be carried out on classical computers, and the output will serve as a constant offset to the energy computed by the quantum circuit, which does not require additional quantum gates for its implementation. In other words, $H_{A'}$ uses classical computation results to adjust the quantum measurement or final interpretation of the quantum states. Thus, $H_{A'}$ will cost $\mathcal{O}(M+N)$ on classical computers and the total cost of obtaining H_C is $\mathcal{O}(2N+M)$.

Since there are s QAOA layers and R repeating times, assume the total iteration step is k, the total cost will be:

$$\mathcal{O}(kRN) + \mathcal{O}(ksR(3N+M)) \rightarrow \mathcal{O}(poly(s)R(3N+M)), \tag{21}$$

where $\mathcal{O}(poly(s)RN)$ and $\mathcal{O}(poly(s)R(N+M))$ are the costs of quantum circuits and classical computers, respectively. In this paper, it is empirically discovered that $poly(s) \ll R$, thus, the complexity of the developed hybrid algorithm is $\mathcal{O}(RN)$ for quantum circuits execution and $\mathcal{O}(R(N+M))$ for a classical computer execution.

C. QAOA Training Landscape Optimizing with Strategic Penalties

During the iterative refinement of QAOA, the objective is to adjust the vector of control parameters $\vec{\gamma}$ and $\vec{\beta}$, in order to optimize the expected value of the problem Hamiltonian $F_s(\vec{\gamma}, \vec{\beta})$. In this subsection, we propose a constrained optimization method for the landscape of QAOA during the training phase, which aims to promote the quality of the solution and improve convergence.

The solution set generated by QAOA contains a large number of strings that cannot guarantee full observability or use too many PMUs. In addition, the generated solution set is random, where the quantum circuits have no prior knowledge about the power grid. One of the crucial reasons is that the PMUs are installed at several less important nodes, which is unnecessary. Thus, we can add strategic penalties to optimize the QAOA training landscape. The penalties can be categorized into two kinds: penalties for improper placement P_1, and for redundant placement P_2.

For P_1, based on graph theory, Lemma 2 is proposed as:

Lemma 2 (Degree-Prioritized Placement Lemma)**.** *For a grid that contains at least one bus with a degree of 3 or greater, installing PMUs on buses with a degree of 1 or 2 is less critical compared to installation on buses with a degree of 3 or more.*

The proof of Lemma 2 can be found in the Appendix III. In most power systems, it is common for buses to have 3 or more neighbors. According to Lemma 2, for any placement $\mathbf{X}$, if a node with a degree of 1 or 2 is installed with a PMU, then a penalty can be added to the corresponding problem Hamiltonian. For example, as illustrated in Fig. 2, taking into account the strategic importance of buses 1 and 4 demonstrates that a PMU installed at bus 4 would cover more buses, whereas a PMU at bus 1 would only make itself observable.

Additionally, penalties may be imposed for excessive PMU installations, even when the system achieves full observability. The penalty threshold T can be established using baseline methods from prior research, which provide benchmarks on the number of PMUs typically required. For instance, most previous work has reported 7 PMUs for the IEEE 30-bus system, and $T = 7$ can be chosen as the threshold. If no existing work provides baseline results, T can be established by running the developed QAOA method $\mathcal{H}$ times and taking the ceiling of the average optimal results across $\mathcal{H}$ implementations. Thus, the modified problem Hamiltonian can be formulated as:

$$H_C = \begin{cases} H_C & \text{if } \forall x_i = 0 \text{ when } \mathcal{L}(x_i) \leq 2, \\ & \text{and } \sum_{i=1}^{N} x_i \leq T \\ H_C + P_1 & \text{if } \exists x_i = 1 \text{ when } \mathcal{L}(x_i) \leq 2, \\ & \text{and } \sum_{i=1}^{N} x_i \leq T \\ H_C + P_2 & \text{if } \forall x_i = 0 \text{ when } \mathcal{L}(x_i) \leq 2, \\ & \text{and } \sum_{i=1}^{N} x_i > T \\ H_C + P_1 + P_2 & \text{if } \exists x_i = 1 \text{ when } \mathcal{L}(x_i) \leq 2, \\ & \text{and } \sum_{i=1}^{N} x_i > T \end{cases} \tag{22}$$

where P_1 and P_2 are the constant value penalties. In this work, P_1 and P_2 are both set to 500. The determination process is carried out on classical computers that also do not need extra quantum gates and qubits. Similarly, the penalty will be added to the energy computed by the quantum circuits as an offset. The mapping process is the same as the QAOA implementation illustrated in Section III-A.

IV. Numerical Examples

In this section, the case studies have been carried out on IEEE 9-, 14-, 24-, and 30-bus systems. The experiments are tested on both an IBM noiseless quantum simulator (ibmq-qasm-simulator, with a maximum of 32 qubits) [56], the IBM real quantum computer (IBM-Brisbane, with a maximum of 127 qubits) [57], and on a local classical server (2 Intel Xeon 6542 CPUs). The quantum simulators can also be simulated on classical computers, with a maximum limit of 30 qubits. In each experiment, the involved qubits equal the test system size N. The QAOA layers s for IEEE 9-, 14-, and 24-bus systems are 7, and 9 for the 30-bus system. Due to budget constraints and limited access to IBM quantum resources, the IEEE 9-bus test is conducted on IBM's real quantum computer, while the IEEE 9-, 14-, and 24-bus systems are evaluated using the IBM quantum simulator. The IEEE 30-bus system is tested on local classical servers. In this study, it is assumed that the placement of the PMUs costs the same on different buses. Conventional and power flow measurements are not taken into account for the analysis, and we assume all PMUs work well. The quantum circuits are executed $R = 10,000$ times for the IEEE 9- and 14-bus systems, $R = 300,000$ times for the 24-bus system, and $R = 500,000$ times for the 30-bus system.

A. Benchmark Tests for PMU Placement: Zero-Injection

In the zero-injection scenarios, the results of the 9-bus system tested on the proposed method and the benchmarks

are listed in the first column of Table I (denoted as Z). The 9-bus case is tested both on the quantum simulator and quantum computer (labeled as QC), where the proposed quantum optimization method performs the same best results on both platforms. The results of the 14-, 24-, and 30-bus systems are listed in the 1-3 columns of Table II. The first column of Table IV shows the installations for IEEE 9-, 14-, 24-, and 30-bus systems under zero-injection scenarios with full observability. The following observations can be made:

- In the 9-bus test case, both the quantum simulator and quantum computers can provide the global optimum results, with 2 PMUs required. Though with the inherent noise and gate errors, the real quantum computer can still illustrate the potential in finding the optimal solutions. Notably, noise and gate errors are the two primary factors contributing to the prolonged execution duration of quantum optimization processes on real quantum computers, compared to noiseless quantum simulators. As shown in Table X, the total execution duration increases significantly from 1,062.90 seconds on quantum simulators to 15,571.54 seconds on real quantum hardware. As quantum technology advances, particularly in noise reduction and error mitigation, the execution duration is expected to decrease, enabling quantum optimization to be applied to significantly larger problem instances more efficiently.
- In the 14-, 24-, and 30-bus systems, the experiments are not tested on quantum computers due to budget and time constraints. On quantum simulators, the proposed method demonstrates better or equivalent performance to the previously best-reported results, demonstrating the installation of 3, 5, and 5 PMUs in respective scenarios.
- It should be noted that with the growing problem size, the number of repetition times of the QAOA circuit execution R should be increased accordingly, which is another key factor to guarantee a good performance of QAOA. The quantum circuits will be repeatedly implemented, on the one hand, to refine the parameters to converge on the optimal or near-optimal results [58]. On the other, sufficient repetitions are demanded to sample all possible states within the superposition in order to locate the global optimal solutions.
- Ref. [31] listed in Table I and Table II utilizes the QA method to determine the minimum number of PMUs. The proposed QAOA method utilizes real quantum computers equipped with superconducting qubits, which are designed to execute quantum gates and circuits in line with the gate model of quantum computing. In contrast, the D-Wave system for QA operates on the adiabatic quantum computation model without employing quantum gates. Conducting the sole QA process on D-Wave systems can be efficient for certain optimization tasks of QUBO format, with a relatively fast complexity of $\mathcal{O}(Re^{-\frac{1}{N}})$ on D-Wave QPU [59]. For [31], the QUBO form converting and the quantum system embedding requires $\mathcal{O}(N^2)$ complexity on a classical computer and D-Wave QPU respectively, thus the overall complexity is $\mathcal{O}(N^2 + Re^{-\frac{1}{N}})$. However, the utility of QA is also confined to problems that can be reformulated into the QUBO format, limiting their broader applicability. In Ref. [31], the observability modeling method is simplified to $\mathcal{G}_1$ in order to convert the objective function into the QUBO form. The use of the $\mathcal{G}_1$ observability modeling approach results in an increased number of PMUs for full system observability, requiring 3, 4, 7, and 10 PMUs in respective scenarios. Additionally, this methodology tends to produce suboptimal results in larger systems. For example, when applied to the IEEE 57 and 118-bus systems, Ref. [31] needs 17 and 32 PMUs respectively, compared to the 12 and 28 PMUs typically required by most benchmark methods.

TABLE I

MINIMUM NUMBER OF PMUS FOR FULL OBSERVABILITY: IEEE 9-BUS SYSTEM BENCHMARK COMPARISON

	IEEE 9-bus-Z	IEEE 9-bus-I
Proposed method	**2**	**3**
Proposed method on QC	**2**	**3**
Ref. [31]	3	4
Ref. [17]	NA	3
Ref. [18]	NA	3
Ref. [34]	NA	3
Ref. [60]	NA	4
Ref. [61]	NA	4
Ref. [62]	2	NA
Ref. [63]	3	3

TABLE II

MINIMUM NUMBER OF PMUS FOR FULL OBSERVABILITY: IEEE 14-, 24-, 30-BUS SYSTEM BENCHMARK COMPARISON

	IEEE 14-bus-Z	IEEE 24-bus-Z	IEEE 30-bus-Z	IEEE 14-bus-I	IEEE 24-bus-I	IEEE 30-bus-I
Proposed method	**3**	**5**	**5**	**7**	**8**	**10**
Ref. [31]	4	7	10	NA	NA	NA
Ref. [9]	3	NA	5	NA	NA	NA
Ref. [17]	3	6	7	7	8	10
Ref. [18]	3	NA	7	NA	NA	NA
Ref. [22]	3	NA	7	NA	NA	NA
Ref. [29]	3	NA	7	NA	NA	NA
Ref. [34]	3	6	7	NA	NA	NA
Ref. [60]	4	7	10	NA	NA	NA
Ref. [61]	4	NA	10	NA	NA	NA
Ref. [62]	NA	7	NA	NA	8	NA
Ref. [63]	3	NA	NA	7	NA	NA
Ref. [64]	NA	NA	NA	7	10	11
Ref. [65]	NA	NA	NA	7	9	NA
Ref. [66]	NA	NA	7	NA	NA	10
Ref. [67]	NA	NA	7	NA	NA	10
Ref. [68]	NA	7	10	NA	NA	NA

B. Benchmark Test For PMU Placement: Non-Zero Injection

As for the non-zero injection scenarios, the results of the 9-bus system are shown in the second column of Table I (denoted as I). The results of the 14-, 24-, and 30-bus systems are shown in the 4-6 columns of Table II. The second column of Table IV illustrates the PMU placement solutions for IEEE 9-, 14-, 24-, and 30-bus systems under the non-zero injections. From Table II, we can see that:

- In the 9-bus case, similar to the zero-injection scenarios, the performance of the proposed quantum optimization method is still the best on both the quantum simulators and the quantum computers, with 3 PMUs required to make the system observable.

TABLE III

BASELINE METHODS COMPUTATION COMPLEXITY

	Complexity
Proposed method	$\mathcal{O}(RN)$ on QC, $\mathcal{O}(R(N+M))$ on classical
Ref. [31]	$\mathcal{O}(N^2)$ on classical, $\mathcal{O}(N^2+e^{-\frac{1}{N}})$ on QPU
Ref. [9]	$\mathcal{O}(N^2 \log N)$
Ref. [17]	$\mathcal{O}(2^N)$
Ref. [18]	$\mathcal{O}(poly(N))$
Ref. [22]	$\mathcal{O}(poly(N))$
Ref. [29]	$\mathcal{O}(poly(N))$
Ref. [34]	$\mathcal{O}(poly(N))$
Ref. [60]	$\mathcal{O}(poly(N))$
Ref. [61]	$\mathcal{O}(poly(N))$
Ref. [62]	$\mathcal{O}(poly(N))$
Ref. [63]	$\mathcal{O}(poly(N))$
Ref. [64]	$\mathcal{O}(2^N)$
Ref. [65]	$\mathcal{O}(poly(N))$
Ref. [66]	$\mathcal{O}(poly(N))$
Ref. [67]	$\mathcal{O}(poly(N))$
Ref. [68]	$\mathcal{O}(poly(N))$

- For the 14-, 24-, and 30-bus systems, on quantum simulators and classical servers, the proposed quantum optimization method can outperform the other classical methods, where 7, 8, and 10 PMUs are installed to guarantee full observability.

C. Benchmark Tests for PMU Placement: Channel Limitations

As for the channel limitation scenarios, among all 4 test systems, the maximum degree of the node is 7, and if the channel limitation is 6 and this node is placed with a PMU, it can still be fully observable since the last 1 branch current can be calculated via KCL. Thus, the number of PMU channels from 1 to 5 is studied. The number of installed PMUs for 9-, 14-, 24-, and 30-bus cases are demonstrated in Table V, VI, VII, and VIII respectively. The detailed installation results are shown in Table IX. For the 9-bus system, since there are no previous studies on it and the system is trivial, the Brutal-Force (BF) algorithm can be utilized to find the global optimum. The following observations can be concluded from Table V, VI, VII, VIII, and IX:

- For 9- and 14-bus systems, the proposed method with both $\mathcal{M}_2$ and $\mathcal{M}_3$ demonstrates equivalent performances. The global optimum is achieved by both observability modeling strategies on the 9-bus system. In addition, the proposed method with both strategies outperforms the baseline results on the 14-bus system.
- For the 24-bus system, the proposed quantum optimization method with $\mathcal{M}_2$ and $\mathcal{M}_3$ illustrates the outperformance compared with the benchmark results. Notably, applying the random neighbor selection strategy ($\mathcal{M}_3$) illustrates better results than applying the greedy strategy ($\mathcal{M}_2$) when the channel limitation is 2. When the test system is relatively small, adding an amount of randomness can benefit the performance of the quantum optimization method within the capability of the quantum computation resources, thus boosting the optimization performance.
- For the 30-bus system, the performance of the proposed quantum optimization method using both $\mathcal{M}_2$ and $\mathcal{M}_3$ still beats the baseline outcomes. Different from the 24-bus case results, the proposed method with $\mathcal{M}_2$ produces better outcomes than $\mathcal{M}_3$ when the channel limitation is set to 4 and 5. As the problem size grows, the randomness utilizing $\mathcal{M}_3$ also grows, which requires an increasing number of quantum computation resources to handle the randomness. Thus, greedy selection strategy $\mathcal{M}_2$ is much more likely to be recommended as the system size expands significantly.

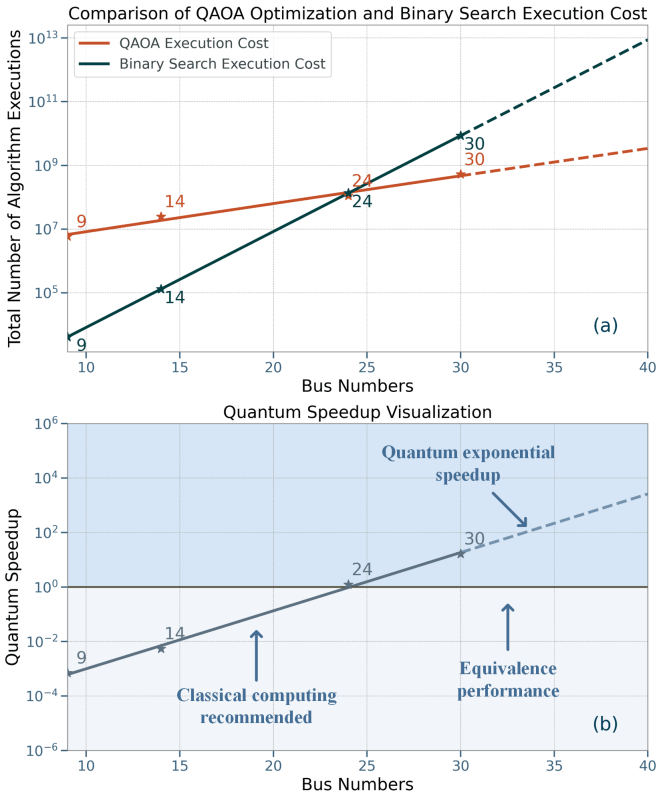

Fig. 3. The theoretical quantum speedup in comparison with the combinatorial search Method.

D. Theoretical and Numerical Complexity Analysis of the Quantum Speedup

In Table III, the theoretical computation complexity of the baseline methods is summarized. As indicated previously, Ref. [31] utilizes QA techniques, which costs $\mathcal{O}(N^2)$ on classical computers to form the adjacency matrix and $\mathcal{O}(N^2+e^{-\frac{1}{N}})$ on QPU for the QA process [59]. Ref [9] utilizes the Minimal Spanning Tree technique, where the complexity to generate a spanning tree is $\mathcal{O}(M \log M)$. In addition, for each spanning tree, $\mathcal{O}(N)$ complexity is needed to determine the observability of the generated spanning tree. Given that power grids are often sparse, the assumption that $M = \mathcal{O}(N)$ can be set, and the overall complexity for [9] is $\mathcal{O}(N^2 \log N)$. For Refs

TABLE IV

PMU INSTALLING SOLUTIONS OF THE PROPOSED METHOD

	Zero-injection	**Non-zero injection**
9-bus	4, 6	4, 6, 8
14-bus	2, 4, 13	2, 4, 5, 6, 9, 10, 13
24-bus	6, 9, 10, 17, 23	2, 3, 8, 9, 10, 13, 16, 18
30-bus	2, 6, 10, 12, 27	2, 3, 5, 6, 9, 10, 12, 15, 19, 27

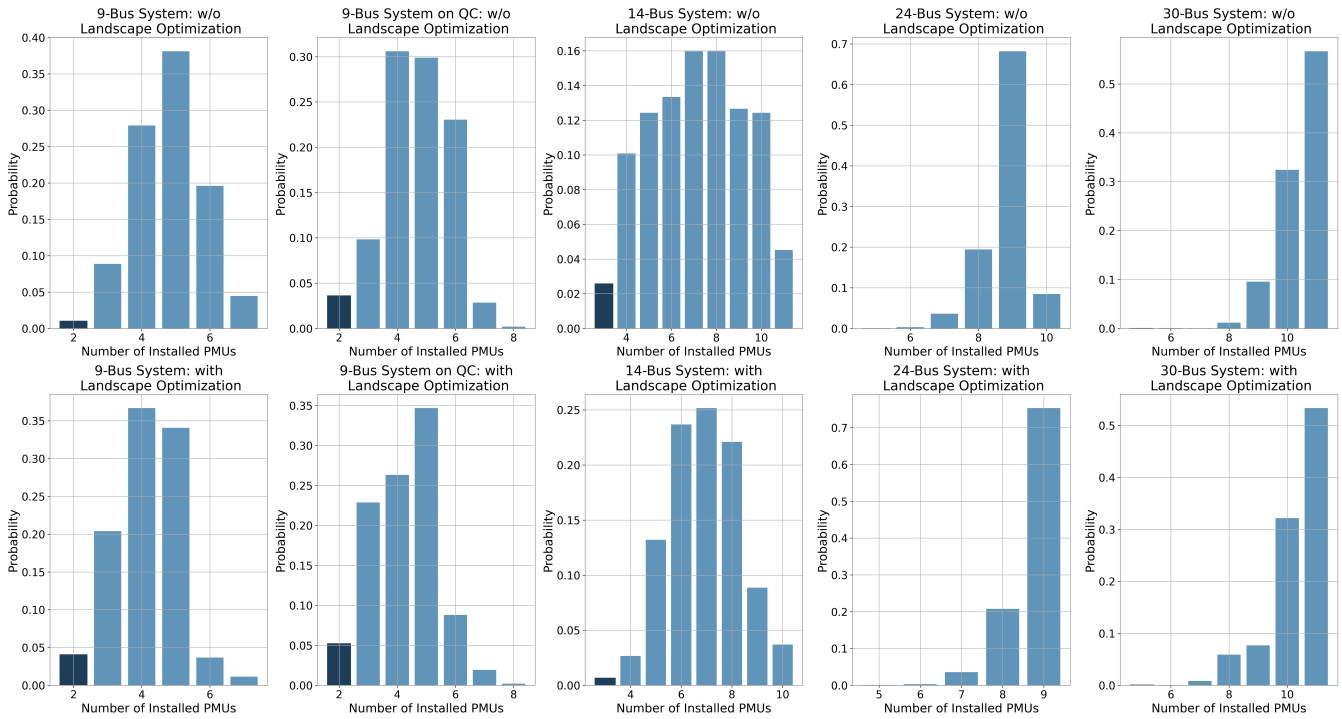

Fig. 4. Distributions of the PMU requirements across IEEE 9-, 14-, 24-, and 30-bus systems with and without landscape optimizing.

TABLE V
MINIMUM NUMBER OF PMUS UNDER CHANNEL LIMITATION: IEEE-9 BUS SYSTEM

Test System	**Channel Limitation ω**				
	1	2	3	4	5
Proposed Method with $\mathcal{M}_2$	**3**	**2**	**2**	**2**	**2**
Proposed Method with $\mathcal{M}_3$	**3**	**2**	**2**	**2**	**2**
BF algorithm	3	2	2	2	2

TABLE VI
MINIMUM NUMBER OF PMUS UNDER CHANNEL LIMITATION: IEEE-14 BUS SYSTEM

Test System	**Channel Limitation ω**				
	1	2	3	4	5
Proposed Method with $\mathcal{M}_2$	**7**	**4**	**3**	**3**	**3**
Proposed Method with $\mathcal{M}_3$	**7**	**4**	**3**	**3**	**3**
Ref. [69]	7	5	4	3	NA
Ref. [51]	7	5	4	3	3
Ref. [70]	NA	NA	4	4	NA
Ref. [29]	NA	NA	4	3	NA

TABLE VII
MINIMUM NUMBER OF PMUS UNDER CHANNEL LIMITATION: IEEE-24 BUS SYSTEM

Test System	**Channel Limitation ω**				
	1	2	3	4	5
Proposed Method with $\mathcal{M}_2$	**10**	**7**	**6**	**5**	**5**
Proposed Method with $\mathcal{M}_3$	**10**	**6**	**6**	**5**	**5**
Ref. [69]	10	7	6	6	NA
Ref. [70]	NA	NA	9	7	NA

TABLE VIII
MINIMUM NUMBER OF PMUS UNDER CHANNEL LIMITATION: IEEE-30 BUS SYSTEM

Test System	**Channel Limitation ω**				
	1	2	3	4	5
Proposed Method with $\mathcal{M}_2$	**13**	**8**	**7**	**6**	**6**
Proposed Method with $\mathcal{M}_3$	**13**	**8**	**7**	**7**	**7**
Ref. [69]	13	8	7	NA	NA
Ref. [51]	14	9	8	7	7
Ref. [70]	NA	NA	9	8	NA
Ref. [29]	NA	NA	7	7	NA

[17] and [64], vanilla binary search is involved, yielding an exponential complexity $\mathcal{O}(2^N)$. Refs [18], [22], [29], [34], [60]–[63], [65]–[68] apply inequality constraints, and the most common solvers (Gurobi, CPLEX) can solve such ILP in $\mathcal{O}(poly(N))$ complexity.

Empirically, based on the numerical results of the IEEE 9-, 14-, 24-, and 30-bus systems and the computation complexity obtained in Section III-B, the approximate computation cost of the QAOA optimization method based on the computation complexity can be calculated. The iteration step k of COBYLA for 9-, 14-, 24-, and 30-bus systems are 173, 176, 184, and 223, respectively, which indicates a fast convergence speed. Assume that the execution cost of quantum circuits is the same as that of classical computers. The theoretical execution counts of quantum optimization combine the execution cost of both quantum circuits and classical computers. Compared with the combinatorial search method in [17], the speedup can be visualized in Fig. 3, where the y-axis is displayed on a logarithmic scale to clearly visualize the computation cost and speedup. From Fig. 3, we can see that:

- When the problem size is small, QAOA underperforms the classical combinatorial search due to the repeated times R. However, as the number of buses increases,

TABLE IX
PMU INSTALLING SOLUTIONS FOR CHANNEL LIMITATION SCENARIO

	Channel Limitation ω				
	1	2	3	4	5
9-bus ($\mathcal{M}_2$)	3, 7, 9	3, 8	1, 8	1, 6	1,6
9-bus ($\mathcal{M}_3$)	1, 2, 9	1, 8	2, 6	6, 8	4, 8
14-bus ($\mathcal{M}_2$)	1, 2, 5, 7, 9, 11, 12	2, 5, 6, 7	2, 9, 13	1, 4, 13	2, 4, 13
14-bus ($\mathcal{M}_3$)	2, 3, 4, 7, 11, 12, 14	2, 4, 10, 13	1, 4, 6	2, 9, 13	2, 4, 6
24-bus ($\mathcal{M}_2$)	2, 5, 8, 9, 10, 11, 12, 21, 22, 23	2, 8, 10, 12, 14, 20, 21	2, 10, 13, 20, 21, 24	1, 9, 12, 20, 21	6, 9, 10, 17, 23
24-bus ($\mathcal{M}_3$)	2, 3, 6, 8, 9, 10, 13, 18, 20, 21	2, 9, 11, 12, 20, 21	3, 8, 10, 13, 20, 21	9, 10, 13, 20, 21	3, 10, 13, 20, 21
30-bus ($\mathcal{M}_2$)	2-4, 7, 10, 13, 15, 16, 19, 22, 24, 26, 28	1, 9, 10, 14, 15, 25, 28, 29	2, 6, 8, 10, 15, 26, 27	2, 6, 10, 11, 12, 27	2, 6, 10, 11, 12, 27
30-bus ($\mathcal{M}_3$)	2, 3, 6, 11, 13, 15, 16, 19, 21, 22, 27, 28, 30	1, 9, 10, 14, 15, 25, 28, 29	2, 6, 8, 10, 15, 26, 27	1, 6, 10, 15, 22, 27, 29	1, 6, 12, 20, 22, 28, 30

TABLE X
COMPARISON OF QAOA PERFORMANCE WITH AND WITHOUT STRATEGIC PENALTY OPTIMIZATION

	$\mathcal{E}_W$	$\mathcal{E}_O$	$\mathcal{V}_W$	$\mathcal{V}_O$	$\mathcal{K}_W$	$\mathcal{K}_O$	$\mathcal{Z}_W$	$\mathcal{Z}_O$
9-bus	3.34	3.26	0.33	0.35	0.05	0.08	1,062.90 s	524.54 s
9-bus-QC	4.18	3.82	0.89	0.38	0.01	0.04	15,571.54 s (374 s)	10,124.87 s (363 s)
14-bus	4.84	4.77	0.21	0.22	0.01	0.03	1,187.03 s	582.09 s
24-bus	7.62	7.74	0.31	0.26	2.44×10^{-3}	2.46×10^{-3}	3,631.72 s	2,866.49 s
30-bus	10.44	10.2439	0.54	0.58	3.00×10^{-4}	5.00×10^{-4}	139,308.93 s	120,724.99 s

the computation cost required by QAOA grows much more slowly compared to combinatorial search. Consequently, quantum optimization becomes more efficient for larger problem sizes. The speedup becomes even more pronounced as the problem size increases to a 40-bus system, where the theoretical speedup is about 10^4.

- However, in the NISQ era, the performance of quantum computing is still restricted by the limited coherence times, gate fidelities, and the number of available qubits [71]. Despite this, the advancements in quantum hardware are expected to enhance the capabilities of quantum computing, making it a more viable tool for solving large-scale optimization problems in the future.

E. Analysis of QAOA Landscape Optimizing

In Fig. 4, according to the obtained $P(\mathcal{F}_n)$ in Section III-A, the distributions of the solution strings for the IEEE 9-, 14-, 24-, and 30-bus systems are presented. The expected number of installed PMUs and their associated variance are quantitatively assessed for two scenarios: with and without landscape optimization application (see Table X for detailed values). These values are denoted as $\mathcal{E}_O$ and $\mathcal{V}_O$ for the optimized scenario and $\mathcal{E}_W$ and $\mathcal{V}_W$ for the non-optimized scenario, respectively (a subscript of "O" indicates the application of optimization, and "W" indicates that no optimization is applied.). We also record the probability of the optimal results in the solution distributions as $\mathcal{K}_W$ and $\mathcal{K}_O$, which is emphasized in darker blue in Fig. 4. In addition, the execution time costs $\mathcal{Z}_W$ and $\mathcal{Z}_O$ for both strategies are also counted.

According to Table X, we have the following observations:

- The expectation and variance values after landscape optimization change slightly, indicating that landscape optimization does not have a significant impact on the overall quality of the solution distribution.
- Landscape optimization boosts the performance of QAOA in finding the optimal solutions. Since only the optimal solution strings are utilized as the results in OPMUP, landscape optimization is recommended in order to increase the probability of locating the optimal solutions.
- Notably, the execution time of QAOA after landscape optimization has a significant decrease: on the quantum simulator, the implementation time dropped by about 50% for the IEEE 9- and 14-bus system, and 23% for the 24-bus system when using landscape optimization. When implementing the developed methodology on the real quantum computer for the 9-bus system, the total execution time decreases from 15571.54 seconds to 10124.87 seconds by 35%, while the execution time on the quantum computer itself decreases from 374 seconds to 363 seconds by 3% with landscape optimization strategy (implementing on quantum computers still requires classical computation resources to implement the QAOA parameter optimization). As for the 30-bus system, where the quantum circuits are simulated on classical servers, the total time decreases by 14% utilizing landscape optimization. Generally, landscape optimization has demonstrated significant empirical benefits for boosting the efficiency of QAOA.

Landscape optimization techniques involve imposing additional constraints or penalties to guide the search toward more favorable regions of the solution space. Thus, the performance of QAOA can be enhanced in terms of the execution time and the probability in finding the optimal solutions.

F. Analysis of QAOA parameters

The analysis of the QAOA parameters is also conducted on IBM quantum simulators and classical servers for the OPMUP. The QAOA circuits have $2s$ trainable parameters $\vec{\gamma}$ and $\vec{\beta}$, and the number of these parameters can impact the optimization results. It should be noted that all the parameters are randomly initialized. With fewer parameters, the optimization landscape is less complex, allowing faster convergence. For example, in Fig. 5(a), since the optimization landscape is simple and the initial state is close to the optimal results, QAOA can easily converge. However, given the limited repeating times R, a

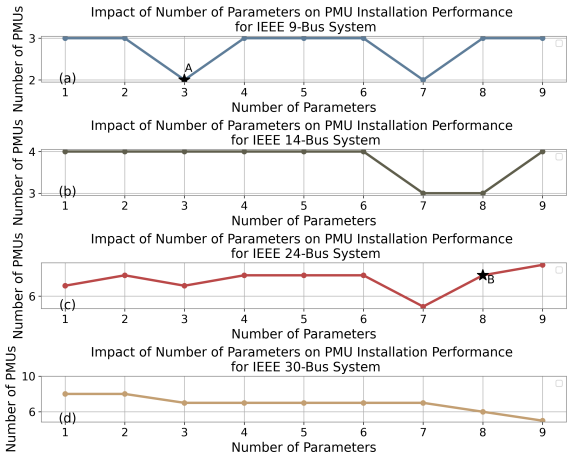

Fig. 5. The quantum optimization performance with different numbers of parameters for IEEE 9, 14, and 24-bus systems.

small number of parameters can limit the expressive power of QAOA, reducing its ability to approximate complex solutions accurately and potentially leading to suboptimal results [72], [73]. In contrast, more parameters generally enhance the ability of the QAOA circuit to represent complex and accurate approximations of the optimal solution. However, as the number of parameters increases, the optimization landscape becomes more complex for the optimization algorithm, which can also impose challenges in finding global optimals [72], [73]. For example, in Fig. 5(c), as we move from point B toward a higher number of parameters, the number of installed PMUs increases, indicating poorer search results.

In Fig. 5, the performance of quantum optimization is analyzed under various parameter settings. For simple test cases such as the 9-bus system, both a few parameters and a reasonable increase in the number of parameters can guarantee optimal results. For more complex systems, such as the 14-bus and 24-bus systems, a moderate increase in the number of parameters generally benefits the QAOA optimization performance. However, applying too many parameters can sometimes lead to worse results, highlighting the need for careful parameter management to avoid suboptimal solutions. For larger test cases such as the 30-bus system, increasing the number of QAOA parameters enhances the representation power of the quantum circuits. This results in a monotonic decrease in the number of required PMUs installed for the 30-bus system as the number of parameters increases. Consequently, as the problem size grows, maintaining a sufficient number of QAOA parameters is essential to preserve the representation power of the quantum circuits, ensuring high solution quality as the complexity of the problem increases. Generally, more parameters do not necessarily guarantee the optimal solution, and the parameter settings need to be carefully balanced to navigate the optimization landscape effectively based on the problem size.

V. Conclusion

In this work, an end-to-end QAOA optimization method is proposed as a pioneering trial to determine the OPMUP for both normal and channel limitation situations. A tailored objective function that incorporates both the number of installed PMUs and observability constraints is introduced. Furthermore, more efficient observability modeling algorithms with a complexity of $\mathcal{O}(M+N)$ are proposed to evaluate the quality of the solutions generated by QAOA, without requiring any extra qubits compared with ILP. Overall, the proposed QAOA optimization costs $\mathcal{O}(RN)$ on quantum circuits and $\mathcal{O}(R(N+M))$ on classical computers, which is more efficient than the binary search, meta-heuristic, and ILP methods.

In the numerical studies, the proposed method is tested on the IEEE 9-, 14-, 24-, and 30-bus systems under both normal and channel limitation scenarios, achieving results that are better than previous benchmarks. The empirical outcomes establish the first benchmarks of QAOA applications in OPUMP. Furthermore, the developed quantum optimization method illustrates the theoretical potential to exponentially surpass binary search in terms of implementation counts for OPMUP, given the advancements in quantum hardware. An experiment of landscape optimization is also conducted, demonstrating that it can boost the performance of QAOA in terms of efficiency and the ability to locate optimal results, providing a more applicable and economical strategy for NISQ-era quantum applications. Lastly, a parameter study is conducted, empirically illustrating that increasing the number of parameters can enhance performance to some extent. However, involving too many parameters can complicate the optimization landscape and potentially compromise performance.

In our future work, the proposed quantum optimization method will be extended to real-time grid management, enabling periodic updates to PMU placement recommendations in response to system changes, such as variations in load injections or modifications in grid topology.

Appendix I
Comparison of Observability Modeling Methods

To demonstrate the insufficiency of $\mathcal{G}_1$ in Section II-B.1, in this section, the illustration of obtaining the observability of the given IEEE 9-bus system example based on $\mathcal{G}_1$ and the proposed $\mathcal{M}_1$ is given. And it is assumed that buses 4 and 8 are placed with PMUs.

As shown in Fig. 4, if PMUs are installed at buses 4 and 8, buses 1, 2, 4, 5, 7, 8, and 9 can be observed. However, based on a rule (iv), since the currents $\mathbf{I}_{45}$ and $\mathbf{I}_{75}$ are known, we can calculate $\mathbf{I}_{56}$ and $\mathbf{I}_{36}$ with KCL: $\mathbf{I}_{56} = -\mathbf{I}_{45}, \mathbf{I}_{76} = -\mathbf{I}_{87}, \mathbf{I}_{36} = -\mathbf{I}_{56} - \mathbf{I}_{76}$.

Then $\mathbf{U}_6$ and $\mathbf{U}_3$ can be computed via rule (iii): $\mathbf{U}_6 = \mathbf{U}_5 - \mathbf{Z}_{56}\mathbf{I}_{56}, \mathbf{U}_3 = \mathbf{Z}_{36}\mathbf{I}_{36} + \mathbf{U}_6$.

Thus, the IEEE-9 bus system can be fully observable.

Appendix II
Proof for Lemma 1

In this section, the proof for the Partially Observable Lemma in Section II-B.2 is given. The Lemma is recapped as:

Lemma 1 (Partially Observable Lemma)**.** *For any partially observable bus, at least one current in a connected branch is known.*

Proof:

- Assume a bus $i \in E$ and its connected buses $j, j \in \mathcal{L}(i)$. For any partially observable buses, it is not possible that they are directly connected to a PMU since a PMU can make it fully observable if there are no channel limitations.
- We want to prove that if only $\mathbf{U}_i$ is known, then at least one current in the connected branch $\mathbf{I}_{ij}$ is required to be known. This can be proved by contradiction. If no currents connected to i are known and $\mathbf{U}_i$ is known, and i is not placed with a PMU, then $\mathbf{U}_i$ can only be obtained via rule (iii). However, the assumption that no currents are known leads to the impossibility of applying Ohm's law to determine $\mathbf{U}_i$, hence the assumption is false. □

APPENDIX III
PROOF FOR LEMMA 2

In this section, the proof for the Degree-Prioritized Placement Lemma in Section III-C is demonstrated. The Lemma is recapped as:

Lemma 2 (Degree-Prioritized Placement Lemma)**.** *For grid that containing at least one bus with a degree of 3 or greater, installing PMUs on buses with a degree of 1 or 2 is less critical compared to installation on buses with a degree of 3 or more.*

Proof:

- In any connected (excluding circular and fully connected) graph where there exists at least one node A with a degree greater than or equal to 3, there must exist at least one node B with a degree of 1 or 2 that is connected to A.
- If a PMU is installed at A, then B can be fully observable. However, if B is placed with a PMU, it is possible that A is not fully observable. Thus, it is more optimal to install a PMU at the buses with 3 more degrees. □

APPENDIX IV
PROPOSED ALGORITHM UNDER CHANNEL LIMITATION

In this section, the details of the proposed observability modeling function for channel limitation scenarios discussed in Section II-B.4 are shown.

The detailed algorithm of $\mathcal{M}_2$ is illustrated in Algorithm 2. The differences of $\mathcal{M}_2$ and $\mathcal{M}_3$ are in the second block of the Algorithm 2, where the neighbors of j during the iterations are randomly selected.

Algorithm 2: Proposed observability modeling algorithm: channel limitation

Input : Adjacency matrix $\mathbf{A}$, placement sequence $\mathbf{X}$, bus set $\mathcal{V}$, channel limitation ω

Output: Total number of observed buses O

```
CalObservabilityd(A, X, V, ω):
I ← Initial placement based on X;
Enqueue(Q, I) ;                    // All the PMU-installed buses enqueue
Θ ← Q.size
Y ← 0_{N×N}
O ← {}
R ← {} ;                 // Fully observable set
P ← {} ;            // Partially observable set
Q ← N(V) ;               // Q ∈ R^{N×1}, counter of neighbors for all nodes in V
Continue the algorithm...;
```

REFERENCES

[1] V. Kekatos, G. B. Giannakis, and B. Wollenberg, "Optimal placement of phasor measurement units via convex relaxation," *IEEE Transactions on power systems*, vol. 27, no. 3, pp. 1521–1530, 2012.

[2] K. M. Schumacher, "Optimization algorithms for power grid planning and operational problems." Ph.D. dissertation, 2014.

[3] "Ibm debuts next-generation quantum processor ibm quantum system two, extends roadmap to advance era of quantum utility," https://newsroom.ibm.com/2023-12-04-IBM-Debuts-Next-Generation-Quantum-Processor-IBM-Quantum-System-Two,-Extends-Roadmap-to-Advance-Era-of-Quantum-Utility.

[4] P. Gopakumar, M. Jaya Bharata Reddy, and D. K. Mohanta, "Pragmatic multi-stage simulated annealing for optimal placement of synchrophasor measurement units in smart power grids," *Frontiers in Energy*, vol. 9, pp. 148–161, 2015.

[5] M. Hajian, A. Ranjbar, T. Amraee, and A. Shirani, "Optimal placement of phasor measurement units: particle swarm optimization approach," in *2007 International conference on intelligent systems applications to power systems*. IEEE, 2007, pp. 1–6.

[6] N. H. Abd Rahman and A. F. Zobaa, "Integrated mutation strategy with modified binary pso algorithm for optimal pmus placement," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 3124–3133, 2017.

[7] B. Milosevic and M. Begovic, "Nondominated sorting genetic algorithm for optimal phasor measurement placement," *IEEE Transactions on Power Systems*, vol. 18, no. 1, pp. 69–75, 2003.

[8] F. Aminifar, C. Lucas, A. Khodaei, and M. Fotuhi-Firuzabad, "Optimal placement of phasor measurement units using immunity genetic algorithm," *IEEE Transactions on power delivery*, vol. 24, no. 3, pp. 1014–1020, 2009.

[9] B.-H. Kim and H. Kim, "Pmu optimal placement algorithm using topological observability analysis," *Journal of Electrical Engineering & Technology*, vol. 16, no. 6, pp. 2909–2916, 2021.

[10] R. F. Nuqui and A. G. Phadke, "Phasor measurement unit placement techniques for complete and incomplete observability," *IEEE Transactions on Power Delivery*, vol. 20, no. 4, pp. 2381–2388, 2005.

[11] M. Zhang, Z. Wu, J. Yan, R. Lu, and X. Guan, "Attack-resilient optimal pmu placement via reinforcement learning guided tree search in smart grids," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1919–1929, 2022.

[12] Z. Pei, F. Liu, Z. He, G. Chen, H. Zheng, K. Zhu, and B. Yu, "Alphasyn: Logic synthesis optimization with efficient monte carlo tree search," in *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)*. IEEE, 2023, pp. 1–9.

[13] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM computing surveys (CSUR)*, vol. 35, no. 3, pp. 268–308, 2003.

[14] M. Gendreau, J.-Y. Potvin *et al.*, *Handbook of metaheuristics*. Springer, 2010, vol. 2.

[15] B. Gou, "Optimal placement of pmus by integer linear programming," *IEEE Transactions on power systems*, vol. 23, no. 3, pp. 1525–1526, 2008.

[16] ——, "Generalized integer linear programming formulation for optimal pmu placement," *IEEE transactions on Power Systems*, vol. 23, no. 3, pp. 1099–1104, 2008.

[17] S. Chakrabarti and E. Kyriakides, "Optimal placement of phasor measurement units for power system observability," *IEEE Transactions on power systems*, vol. 23, no. 3, pp. 1433–1440, 2008.

[18] K. G. Khajeh, E. Bashar, A. M. Rad, and G. B. Gharehpetian, "Integrated model considering effects of zero injection buses and conventional measurements on optimal pmu placement," *IEEE Transactions on Smart Grid*, vol. 8, no. 2, pp. 1006–1013, 2015.

[19] B. Cao, Y. Yan, Y. Wang, X. Liu, J. C.-W. Lin, A. K. Sangaiah, and Z. Lv, "A multiobjective intelligent decision-making method for multi-

Algorithm 2: Proposed observability modeling algorithm: channel limitation

Continue the algorithm...;
for $\theta = 1$ **to** Θ **do**
  $j \leftarrow$ Dequeue($\mathbf{Q}$)
  $\chi = Q[j]$;
  if $\chi > \omega + 1$ **then**
    $\rho \leftarrow$ sorted.$\mathcal{L}(j)$; // Sort the neighbors of j in descending
    $\vartheta \leftarrow [p \in \rho$ and $\mathcal{Y}[jp] = \mathcal{Y}[pj] = 0]$; // Find out the unvisited branches of j
    $\Upsilon \leftarrow \vartheta[1 : \omega]$; // place the channel at top ω buses
    if $len(\Upsilon) < \omega$ **then**
      $\Xi \leftarrow [p \in \rho$ and $\mathcal{Y}[pj] = \mathcal{Y}[jp] = 1]$
      $\Upsilon \mathrel{+}= \Xi[1 : (\omega - len(\Upsilon))]$; // Ensure ω channels are installed
    end if
    $\Delta \leftarrow [p \in \mathcal{Y}[jp] = \mathcal{Y}[pj] = 1]]$; // Find the visited branches of j
    $\mathcal{P}[j] = len(\Delta \cup \Upsilon)$
    if $\mathcal{P}[j] \geq Q[j]$ **then**
      $\mathcal{R}.add(j)$; // j is observable
      Enqueue($\mathbf{Q}, j$)
      delete $\mathcal{P}[j]$
    end if
    for $p \in \Upsilon$ **do**
      if $\mathcal{Y}[jp] = \mathcal{Y}[pj] = 0$ **then**
        $\mathcal{Y}[jp] = \mathcal{Y}[pj] = 1$
        if $p \notin O$ **then**
          $\mathcal{P}[p] \leftarrow \mathcal{P}[p] + 1$
          if $\mathcal{P}[p] \geq Q[p] - 1$ **then**
            $\mathcal{R}.add(p)$
            Enqueue($\mathbf{Q}, p$)
            delete $\mathcal{P}[p]$
          end if
        end if
      end if
    end for
  end if
  else
    $\mathcal{R}.add(j)$
    for $p \in \mathcal{L}(j)$ **do**
      if $\mathcal{Y}[jp] = \mathcal{Y}[pj] = 0$ **then**
        $\mathcal{Y}[jp] = \mathcal{Y}[pj] = 1$
        if $p \notin O$ **then**
          $\mathcal{P}[p] \leftarrow \mathcal{P}[p] + 1$
          if $\mathcal{P}[p] \geq Q[p] - 1$ **then**
            $\mathcal{R}.add(p)$
            Enqueue($\mathbf{Q}, p$)
            delete $\mathcal{P}[p]$
          end if
        end if
      end if
    end for
  end if
end for
Continue the algorithm...;

Algorithm 2: Proposed observability modeling algorithm: channel limitation

Continue the algorithm...;
while $\mathbf{Q}$ *is not empty* **do**
  $\kappa \leftarrow$ Dequeue($\mathbf{Q}$) **for** $p \in \mathcal{L}(\kappa)$ **do**
    if $\mathcal{Y}[\kappa p] = \mathcal{Y}[p\kappa] = 0$ **then**
      $\mathcal{Y}[\kappa p] = \mathcal{Y}[p\kappa] = 1$
      if $p \notin O$ **then**
        $\mathcal{P}[p] \leftarrow \mathcal{P}[p] + 1$
        if $\mathcal{P}[p] \geq Q[p] - 1$ **then**
          $\mathcal{R}.add(p)$
          Enqueue($\mathbf{Q}, p$)
          delete $\mathcal{P}[p]$
        end if
      end if
    end if
  end for
end while
$O \leftarrow$ length of $\mathcal{R}$
return O;

stage placement of pmu in power grid enterprises," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 6, pp. 7636–7644, 2022.

[20] A. Pal, A. K. S. Vullikanti, and S. S. Ravi, "A pmu placement scheme considering realistic costs and modern trends in relaying," *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 552–561, 2016.

[21] A. Pal, G. A. Sanchez-Ayala, V. A. Centeno, and J. S. Thorp, "A pmu placement scheme ensuring real-time monitoring of critical buses of the network," *IEEE transactions on power delivery*, vol. 29, no. 2, pp. 510–517, 2013.

[22] F. Aminifar, A. Khodaei, M. Fotuhi-Firuzabad, and M. Shahidehpour, "Contingency-constrained pmu placement in power networks," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 516–523, 2009.

[23] D. Dua, S. Dambhare, R. K. Gajbhiye, and S. Soman, "Optimal multistage scheduling of pmu placement: An ilp approach," *IEEE Transactions on Power delivery*, vol. 23, no. 4, pp. 1812–1820, 2008.

[24] S. Azizi, A. S. Dobakhshari, S. A. N. Sarmadi, and A. M. Ranjbar, "Optimal pmu placement by an equivalent linear formulation for exhaustive search," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 174–182, 2012.

[25] Y. Wang, C. Wang, W. Li, J. Li, and F. Lin, "Reliability-based incremental pmu placement," *IEEE Transactions on Power Systems*, vol. 29, no. 6, pp. 2744–2752, 2014.

[26] N. M. Manousakis and G. N. Korres, "An advanced measurement placement method for power system observability using semidefinite programming," *IEEE Systems Journal*, vol. 12, no. 3, pp. 2601–2609, 2017.

[27] A. Monticelli, "Electric power system state estimation," *Proceedings of the IEEE*, vol. 88, no. 2, pp. 262–282, 2000.

[28] G. N. Korres, N. M. Manousakis, T. C. Xygkis, and J. Löfberg, "Optimal phasor measurement unit placement for numerical observability in the presence of conventional measurements using semi-definite programming," *IET Generation, Transmission & Distribution*, vol. 9, no. 15, pp. 2427–2436, 2015.

[29] M. Elimam, Y. J. Isbeih, M. S. El Moursi, K. Elbassioni, and K. H. Al Hosani, "Novel optimal pmu placement approach based on the network parameters for enhanced system observability and wide area damping control capability," *IEEE Transactions on Power Systems*, vol. 36, no. 6, pp. 5345–5358, 2021.

[30] A. Monticelli, A. Garcia, and I. Slutsker, "Handling discardable measurements in power system state estimation," *IEEE transactions on power systems*, vol. 7, no. 3, pp. 1333–1340, 1992.

[31] E. B. Jones, E. Kapit, C.-Y. Chang, D. Biagioni, D. Vaidhynathan, P. Graf, and W. Jones, "On the computational viability of quantum optimization for pmu placement," in *2020 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2020, pp. 1–5.

[32] M. Z. Islam, Y. Lin, V. M. Vokkarane, and J. Ogle, "Observability-aware resilient pmu networking," *IEEE Transactions on Power Systems*, 2024.

[33] B. Xu and A. Abur, "Observability analysis and measurement placement for systems with pmus," in *IEEE PES Power Systems Conference and Exposition, 2004.* IEEE, 2004, pp. 943–946.

[34] V. Bečejac and P. Stefanov, "Groebner bases algorithm for optimal pmu placement," *International Journal of Electrical Power & Energy Systems*, vol. 115, p. 105427, 2020.

[35] R. Sodhi, S. Srivastava, and S. Singh, "Optimal pmu placement method for complete topological and numerical observability of power system," *Electric Power Systems Research*, vol. 80, no. 9, pp. 1154–1159, 2010.

[36] E. Abiri, F. Rashidi, T. Niknam, and M. R. Salehi, "Optimal pmu placement method for complete topological observability of power system under various contingencies," *International Journal of Electrical Power & Energy Systems*, vol. 61, pp. 585–593, 2014.

[37] F. Glover, G. Kochenberger, R. Hennig, and Y. Du, "Quantum bridge analytics i: a tutorial on formulating and using qubo models," *Annals of Operations Research*, vol. 314, no. 1, pp. 141–183, 2022.

[38] T. Vyskočil, S. Pakin, and H. N. Djidjev, "Embedding inequality constraints for quantum annealing optimization," in *Quantum Technology and Optimization Problems: First International Workshop, QTOP 2019, Munich, Germany, March 18, 2019, Proceedings 1*. Springer, 2019, pp. 11–22.

[39] M. Marzec, "Portfolio optimization: Applications in quantum computing," *Handbook of High-Frequency Trading and Modeling in Finance*, pp. 73–106, 2016.

[40] D. Zhang, J. Wang, H. Fan, T. Zhang, J. Gao, and P. Yang, "New method of traffic flow forecasting based on quantum particle swarm optimization strategy for intelligent transportation system," *International Journal of Communication Systems*, vol. 34, no. 1, p. e4647, 2021.

[41] Y. Cao, J. Romero, and A. Aspuru-Guzik, "Potential of quantum computing for drug discovery," *IBM Journal of Research and Development*, vol. 62, no. 6, pp. 6–1, 2018.

[42] A. Ajagekar and F. You, "Quantum computing for energy systems optimization: Challenges and opportunities," *Energy*, vol. 179, pp. 76–89, 2019.

[43] P. Rebentrost and S. Lloyd, "Quantum computational finance: quantum algorithm for portfolio optimization," *arXiv preprint arXiv:1811.03975*, 2018.

[44] F. Neukart, G. Compostella, C. Seidel, D. Von Dollen, S. Yarkoni, and B. Parney, "Traffic flow optimization using a quantum annealer," *Frontiers in ICT*, vol. 4, p. 29, 2017.

[45] K. Batra, K. M. Zorn, D. H. Foil, E. Minerali, V. O. Gawriljuk, T. R. Lane, and S. Ekins, "Quantum machine learning algorithms for drug discovery applications," *Journal of chemical information and modeling*, vol. 61, no. 6, pp. 2641–2647, 2021.

[46] A. Ajagekar and F. You, "Quantum computing based hybrid deep learning for fault diagnosis in electrical power systems," *Applied Energy*, vol. 303, p. 117628, 2021.

[47] R. Mahroo and A. Kargarian, "Hybrid quantum-classical unit commitment," in *2022 IEEE Texas Power and Energy Conference (TPEC)*. IEEE, 2022, pp. 1–5.

[48] Z. Kaseb, M. Möller, G. T. Balducci, P. Palensky, and P. P. Vergara, "Quantum neural networks for power flow analysis," *Electric Power Systems Research*, vol. 235, p. 110677, 2024.

[49] S. K. Mishra, K. B. Swain, and M. Cherukuri, "Optimal placement of phasor measurement unit using quantum particle swarm optimization," in *2021 1st International Conference on Power Electronics and Energy (ICPEE)*. IEEE, 2021, pp. 1–4.

[50] M. H. R. Koochi, P. Dehghanian, and S. Esmaeili, "Pmu placement with channel limitation for faulty line detection in transmission systems," *IEEE Transactions on Power Delivery*, vol. 35, no. 2, pp. 819–827, 2019.

[51] M. Korkali and A. Abur, "Placement of pmus with channel limits," in *2009 IEEE power & energy society general meeting*. IEEE, 2009, pp. 1–4.

[52] M. Dupont, N. Didier, M. J. Hodson, J. E. Moore, and M. J. Reagor, "Entanglement perspective on the quantum approximate optimization algorithm," *Physical Review A*, vol. 106, no. 2, p. 022423, 2022.

[53] 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," *Physical Review X*, vol. 10, no. 2, p. 021067, 2020.

[54] P. Koch, S. Bagheri, W. Konen, C. Foussette, P. Krause, and T. Bäck, "A new repair method for constrained optimization," in *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, 2015, pp. 273–280.

[55] D. Fortunato, J. CAMPOS, and R. ABREU, "Mutation testing of quantum programs: A case study with qiskit," *IEEE Transactions on Quantum Engineering*, vol. 3, pp. 1–17, 2022.

[56] IBM Quantum, "Ibm quantum simulators," 2024, accessed: 2024-02-13. [Online]. Available: https://cloud.ibm.com/quantum/resources/simulators

[57] M. AbuGhanem, "Ibm quantum computers: Evolution, performance, and future directions," *arXiv preprint arXiv:2410.00916*, 2024.

[58] 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.

[59] S. Morita and H. Nishimori, "Mathematical foundation of quantum annealing," *Journal of Mathematical Physics*, vol. 49, no. 12, 2008.

[60] N. H. Rahman and A. F. Zobaa, "Optimal pmu placement using topology transformation method in power systems," *Journal of advanced research*, vol. 7, no. 5, pp. 625–634, 2016.

[61] X. Chen, F. Wei, S. Cao, C. B. Soh, and K. J. Tseng, "Pmu placement for measurement redundancy distribution considering zero injection bus and contingencies," *IEEE Systems Journal*, vol. 14, no. 4, pp. 5396–5406, 2020.

[62] S. Kumar, "Optimal placement of pmu using probabilistic approach," in *2014 Recent Advances in Engineering and Computational Sciences (RAECS)*. IEEE, 2014, pp. 1–6.

[63] V. S. Devendran, J. Jasni, M. A. Mohd Radzi, and N. Azis, "Optimal placement of pmu for complete observability of power system considering zero injection and islanding condition," in *2020 IEEE International Conference on Power and Energy (PECon)*, 2020, pp. 107–112.

[64] R. Saini, "Optimal placement of phasor measurement units for power system observability without considering zero injection buses," *Journal of Smart Sensor and Adhoc Network: Vol*, vol. 2, no. 2, p. 9, 2012.

[65] C. Lu, Z. Wang, M. Ma, R. Shen, and Y. Yu, "An optimal pmu placement with reliable zero injection observation," *IEEE Access*, vol. 6, pp. 54 417–54 426, 2018.

[66] S. Almasabi and J. Mitra, "A fault-tolerance based approach to optimal pmu placement," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6070–6079, 2019.

[67] H. Ye, C. Tian, Y. Ge, and L. Wu, "Revisit optimal pmu placement with full zero-injection cluster and redundancy sharing," *IEEE Transactions on Power Systems*, 2024.

[68] R. Bhattacharjee and A. De, "A novel bus-ranking-algorithm-based heuristic optimization scheme for pmu placement," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 9, pp. 9921–9932, 2023.

[69] M. Khokhlov, A. Obushevs, I. Oleinikova, and A. Mutule, "Optimal pmu placement for topological observability of power system: Robust measurement design in the space of phasor variables," in *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*. IEEE, 2016, pp. 1–6.

[70] V. Rokkamand and R. Bhimasingu, "A novel approach for optimal pmu placement considering channel limit," in *2014 International Conference on Power System Technology*. IEEE, 2014, pp. 1164–1171.

[71] M. Fellous-Asiani, J. H. Chai, R. S. Whitney, A. Auffèves, and H. K. Ng, "Limitations in quantum computing from resource constraints," *PRX Quantum*, vol. 2, no. 4, p. 040335, 2021.

[72] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature communications*, vol. 9, no. 1, p. 4812, 2018.

[73] 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, 2019.