

Novel Resolution of Unit Commitment Problems through Quantum Surrogate Lagrangian Relaxation

Fei Feng, *Student Member, IEEE*, Peng Zhang, *Senior Member, IEEE*, Mikhail A. Bragin, *Member, IEEE*, Yifan Zhou, *Member, IEEE*

Abstract—Unit commitment (UC) problems faced by Independent System Operators on a daily basis are becoming increasingly complex due to the recent push for renewables and the consideration of sub-hourly UC to accommodate the increasing variability in the net load. A disruptive solution methodology to address the growing complexity is therefore required. Quantum computing offers a promise to overcome the combinatorial complexity through the use of the so-called “qubits.” To make the best use of quantum computers available currently or in the foreseeable future to solve UC problems with a much larger number of binary variables than the number of qubits available, this paper devises a novel solution methodology based on a synergistic combination of quantum computing and Surrogate Lagrangian Relaxation (SLR) to solve UC problems. Our new contributions include: 1) A Quantum-SLR (QSLR) algorithm incorporating quantum approximate optimization algorithm (QAOA) into the SLR method, which overcomes the fundamental difficulties of previous LR-based quantum methods such as zigzagging of multipliers and the need to know or estimate the optimal dual value for convergence; 2) A Distributed QSLR framework (D-QSLR) capable of coordinating local quantum/classical computing resources with those within neighborhoods and, in the meantime, protecting data privacy; 3) A Quantized UC model to obtain accurate commitment unit subproblems decision by using a quantum machine; and 4) A time-unit-decomposed quantum UC approach to overcoming the quantum resources’ limitations. Promising quantum test results validate the effectiveness of QSLR and the scalability of the UC-oriented D-QSLR algorithm, which demonstrate QSLR’s enormous potential in UC optimization.

Index Terms—Quantum Computing, Unit Commitment, Surrogate Lagrangian Relaxation, Distributed Quantum Optimization, Quantum Approximate Optimization Algorithm.

I. INTRODUCTION

UNIT commitment (UC) problems are faced by Independent System Operators on a daily basis and are becoming increasingly complex with high penetration of renewables and the consideration of sub-hourly UC to accommodate the increasing variability in the demand. The scale and complexity of the problem are, thus, expected to drastically increase with the growth of the number of distributed energy resources (DER), controllable loads providing fluctuating demands, and complicated structures [1], [2]. The number of discrete decision variables associated with the increased number of units as well as within the sub-hourly UC leads to the much increased combinatorial complexity. To address the complexity

of discrete optimization problems (such as the UC problems), traditional auxiliary technologies (e.g. parallel computing [3] and parameter tuning strategies for optimization algorithms (e.g. ADMM-based approaches) [4]) may become inadequate in the network application with discrete decision variables since the search region increases tremendously [5]. To enable the successful resolution of UC problems in the presence of the increasing system scale, a disruptive solution methodology is required.

Quantum computing [6] has been proven promising to tackle classical computers’ barriers [7]–[11]. It offers a great potential to overcome the combinatorial complexity through the so-called “qubits,” which are capable of holding an exponential amount of information as compared to the usual “bits” within classical computer [12]–[15]. Current quantum computing algorithms enable solving binary optimization problems. There are different paradigms to realize quantum optimization including adiabatic quantum computing (AQC) [16], and gate model quantum computing [17]. The primary objective of these two kinds of quantum algorithms is to devise methods that require less time and quantum resources to obtain an optimal or near-optimal solution. In these existing methods, the Grover-based adaptive search approaches are implemented for quadratic unconstrained binary optimization (QUBO) problems to find the optimum values with a quadratic speed-up by applying Grover search [18]. Variational quantum eigensolver (VQE) is one of the first attempts for solving binary problems by finding the ground states of a Hamiltonian of a system [19]. An improved VQE-type algorithm is devised to find the minimum objective, meanwhile, reducing the initial state preparation [20]. Another quantum optimization approach is the quantum approximate optimization algorithm (QAOA) to solve QUBO problems [21], which obtains the optimal objective value [22], [23] through the parametrization evolution of the quantum gates. In a broader sense, the aforementioned variational optimal approaches work by choosing a set of quantum states [17], then by using classical optimization routines to determine parameter values of the above quantum states that maximize or minimize a given objective function.

Quantum computing has the potential to solve large-scale mixed binary programming (MBP) problems through quantum entanglement and superposition in the near future. Nevertheless, the state-of-the-art quantum optimizer is inapplicable to UC problems (also many other MBP problems) as they consist of binary decision valuables and continuous generation valuables in mathematical theory. Hybrid algorithms with a trade-off between quantum depth and computing performance

This work was supported in part by the National Science Foundation under Grant Nos. CNS-2006828, ECCS-2018492, ECCS-1810108, OIA-2134840 and OIA-2040599.

The authors are with the Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794-2350, USA (e-mail: P.Zhang@stonybrook.edu).

offer the potential to solve MBP problems, which are suited to today's noisy intermediate-scale quantum devices [24] with a relatively small number of qubits and limited quality. A decomposition-based hybrid quantum/classical optimization approach is extended for MBP problems by using the alternating direction method of multiplier (ADMM), which splits MBP problems into QUBO subproblems solved by quantum computer and continuous constrained convex subproblems on classical computer [25], [26]. While this method is suitable to address combinatorial complexity by encoding binary subproblems through the entangled states within qubits, the method generally does not converge when solving large discrete combinatorial programming problems such as UC problems.

Surrogate Lagrangian Relaxation (SLR) [27], [28], on the other hand, is an appropriate solution technique with convergence proved to cope with the scale and number of variables by splitting into a few smaller-sized subproblems. Inspired by the SLR method, this paper devises a Quantum Surrogate Lagrangian Relaxation algorithm (QSLR), which is expected to solve the MBP problem through quantum computing. Then, a UC-oriented distributed QSLR is established to tackle UC problems, meanwhile, protecting the privacy and saving communication expenses. Our contributions are threefold:

- A QSLR algorithm is devised by incorporating QAOA into the SLR scheme in which binary subproblems are solved through quantum computing and continuous subproblems are solved by a classical solver. The method has the potential for solving rapidly-growing large-scale mixed-integer programming problems, including the UC problem, in the near future.
- A distributed QSLR (D-QSLR) algorithm is established for practical engineering systems to coordinate local quantum/classical computing resources with neighborhoods. This scheme enables the QSLR algorithm to protect data privacy, flexibly regulate quantum/classical sources and support the plug-and-play of subsystems.
- A quantum-encoded UC optimal model is formulated for the quantum solver to find the minimum objective values of unit-wise subproblems. Furthermore, the time-unit decomposition through D-QSLR is devised for large-scale UC problems to address the quantum resource limitations.

The remainder of this paper is organized as follows: Section II gives a description of Quantum Surrogate Lagrangian Relaxation as well as the distributed QSLR framework. Section III focuses on the D-QSLR implementation for UC problems. Case studies that verify the effectiveness and accuracy performance of QSLR and UC-oriented D-QSLR methods in Section IV, followed by the Conclusion in Section V.

II. QUANTUM SURROGATE LAGRANGIAN RELAXATION

Benefiting from quantum superposition and entanglement, currently available quantum computing capabilities have the potential to solve MBP problems. In this section, a QSLR algorithm is developed to efficiently solve MBP problems. The key innovation of the method is the decomposition of MBP problems into binary and continuous subproblems, which are efficiently solved through quantum and classical

computing, respectively, and are subsequently coordinated by updating Lagrangian multipliers. Then, a privacy-friendly D-QSLR framework is developed by coordinating local quantum/classical resources.

A. Framework Design of QSLR

We start with the prototypical MBP problem in the following form:

$$\begin{aligned} \min_{x \in \chi, y \in \mathbb{R}} \quad & \sum_{i=1}^I \{f_i(x_i) + h_i(y_i)\} \\ \text{s.t.} \quad & \sum_{i=1}^I g_i(x_i, y_i) = 0 \end{aligned} \quad (1)$$

where $x = x_1, \dots, x_I$, $y = y_1, \dots, y_I$, χ is the binary set of $\{0,1\}$. Functions $f_i(x_i)$ and $g_i(x_i, y_i)$ are linear and $h_i(y_i)$ is convex. It is assumed that the feasible set is nonempty.

The separability of the problem is exploited by relaxing coupling constraints by introducing Lagrangian multipliers $\lambda^T = (\lambda_1, \dots, \lambda_n)$ and by decomposing the resulting relaxed problem into individual subproblems. A subproblem associated with unit i can be represented as follows:

$$\min_{x_i, y_i, \lambda} L_i(x_i, y_i, \lambda) = \min_{x_i, y_i, \lambda} \{f_i(x_i) + h_i(y_i) + \lambda^T g_i(x_i, y_i)\} \quad (2)$$

The resulting subproblem is further decomposed into several subproblems: binary and continuous, and the high-level description of the overall methodology is presented below:

- Binary subproblems: These subproblems are encoded into quantum formulations and solved by a QAOA-based quantum algorithm which will be devised in subsection B. The solution to the binary subproblem is obtained as follows:

$$\{x_i^{k+1}\} = \arg \min \{L_i(x_i^k, y_i^k, \lambda^k)\} \quad (3)$$

where $x_i^{k+1} = \{x_1^k, \dots, x_i^k, \dots, x_I^k\}$.

- Continuous subproblems: To solve continuous subproblems, classical methods are used. The solution to the continuous subproblem is obtained as follows:

$$\{y_i^{k+1}\} = \arg \min \{L_i(x_i^{k+1}, y_i^k, \lambda^{k+1})\} \quad (4)$$

where $y_i^{k+1} = \{y_1^k, \dots, y_i^k, \dots, y_I^k\}$.

After solving the binary and continuous subproblems, and the ‘‘surrogate optimality condition’’ is checked [29, eq. (12), p. 178], multipliers are updated in the following way:

$$\lambda^{k+1} = \lambda^k + s^k \cdot g(x^{k+1}, y^{k+1}) \quad (5)$$

In the above, stepsizes are updated following [29, eq. (20), p. 180]. The process repeats until convergence.

To efficiently solve binary subproblems, the subproblems will be reformulated in a quantum form amenable for quantum optimization as presented next.

B. Quantum Reformulation of MBP Problems

The main underlying feature of quantum computing is the discretized (more commonly referred to as “quantized”) nature of the quantum mechanical system [30]. Unlike that of classical objects, the rotation (spin) of quantum particles is not continuous. For example, electrons have a spin of $-1/2$ or $+1/2$ and photons can have a spin of -1 or $+1$. This quantization feature will be exploited to capture and solve binary subproblems presented above whereby decision variables, by definition, can only take the values of either 0 or 1. Moreover, unlike those of classical systems, the exact value of a spin is unknown with certainty before measurement. Rather, a spin is an entangled superposition of two possible values. Accordingly, one quantum bit (“qubit”) can hold two bits of information. Two quantum bits, analogously to a quantum mechanical system with two particles with two possible values of spin each, can hold up to 4 bits of information, etc. Generally, a system with N particles will result in a quantum mechanical system that entangles 2^N possible states, which in terms of quantum computing means that N qubits can hold up to 2^N bits of information. Upon measurement of the entangled state, out of 2^N , only one possible combination of spins is possible. In terms of the binary subproblems under consideration, the sought-for optimal solution to the subproblem is also a combination of binary values. The ensuing questions are: 1) How to use quantum computing to encode the binary variables by using qubits and 2) How to make sure that upon measurement, the resulting binary subproblem solution is optimal.

To answer the above questions, an Ising model is used. Historically, the Ising model was used to capture the interaction of spins arranged on a lattice in the presence of an external magnetic field ϱ_i [31]. Mathematically, an Ising model consists of a set of spins ϖ , each taking a value of $\kappa_i \in \{0, 1\}^{|\varpi|}$ [32]. Denoting ς as a set of pair-wise interactions $\Theta_{i,l}$ between spins, we can formulate the energy E of the spin system using a Hamiltonian function as follows:

$$E = \sum_{(i,l) \in \varsigma} \Theta_{i,l} \kappa_i \kappa_l + \sum_{i \in \varpi} \varrho_i \kappa_i \quad (6)$$

The system tends to a state with the lowest energy. A noteworthy feature of the system is that there are no constraints and the system eventually settles down at the ground state (the state with the corresponding minimum value of the energy function) purely through pairwise interactions and through the interaction with the magnetic field.

Analogously, a discrete subproblem can be viewed as a system of interacting binary variables, although, each subproblem is constrained. To map the binary subproblems onto the Ising model, the subproblem is converted into the QUBO problem by penalizing constraint violation within the objective function. That particular binary subproblem will be explained in the next section. The relationship between an Ising model and a QUBO formulation as follows:

$$L_p(x_i^k) = \sum_{1 \leq i, l \leq n} K_{i,l} x_i^k x_l^k + \sum_{i=1}^n \sigma_i x_i^k + \sigma_0 \quad (7)$$

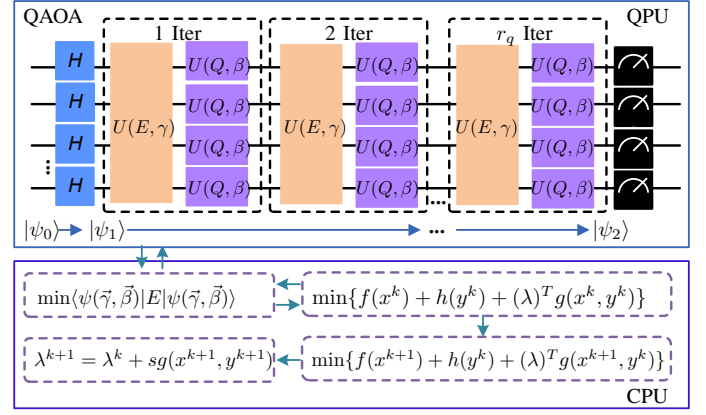


Fig. 1. Quantum circuit architecture for QSLR

where $K_{i,l}, \sigma_i$ are coefficients and σ_0 is a constant value which can be ignored. The mapping ($\kappa_i = 2x_i^k - 1$) between (6) and (7) enables the QSLR algorithm for searching optimal binary solutions by finding the ground state of a Hamiltonian function. Once the ground state of the Ising model is reached, the optimal solution to the QUBO problem is obtained.

C. Quantum optimization mechanism of QAOA

For the aforementioned combinatorial optimization problems, QAOA is used to find the optimal solutions by minimizing the expectation value with respect to the energy E . Such expectation function is reformulated by quantum states entangling the entire possible states. Eventually, the minimization of objective values can be achieved by optimizing the parameters of quantum gates.

Specifically, the expectation function with respect to the energy E is formulated by using the quantum state $|\psi(\vec{\gamma}, \vec{\beta})\rangle$ following [33] as:

$$F_p(\vec{\gamma}, \vec{\beta}) = \langle \psi(\vec{\gamma}, \vec{\beta}) | E | \psi(\vec{\gamma}, \vec{\beta}) \rangle \quad (8)$$

In order to express quantum state $|\psi(\vec{\gamma}, \vec{\beta})\rangle$, a transverse field Hamiltonian is used to construct the quantum state together with E , which is expressed as $Q = \sum_{i,q}^{n_q} U_3(\pi, 0, \pi)$, where Pauli gate $U_3(\pi, 0, \pi) = \begin{bmatrix} \cos(\pi/2) & -e^{j\pi} \sin(\pi/2) \\ e^{j0} \sin(\pi/2) & e^{j\pi} \cos(\pi/2) \end{bmatrix}$. Eventually, the state $|\psi(\vec{\gamma}, \vec{\beta})\rangle$ can be prepared for optimization by applying r_q loops:

$$|\psi(\vec{\gamma}, \vec{\beta})\rangle = Q_{r_q} \cdot E_{r_q} \cdot Q_{r_q-1} \cdot E_{r_q-1} \cdots Q_1 \cdot E_1 |\psi_0\rangle. \quad (9)$$

Thus, QAOA enables the minimization of the expectation by adjusting the quantum states as follows,

$$(\vec{\gamma}^*, \vec{\beta}^*) = \arg \min_{\vec{\gamma}, \vec{\beta}} F_p(\vec{\gamma}, \vec{\beta}), \quad (10)$$

which is typically carried out with numerical optimization routines.

To benchmark QAOA, a desired approximation ratio ℓ^* [34] is defined to quantify how close the expectation value of the Hamiltonian with respect to the ground state energy.

$$\frac{F_p(\vec{\gamma}, \vec{\beta})}{C_{min}} \geq \ell^* \quad (11)$$

where C_{min} denotes the ground state of the objective function. With the increase of r_q loops in a quantum circuit, the evaluated expectation would be approaching the minimal value: $\lim_{r_q \rightarrow \infty} F_p(\vec{\gamma}, \vec{\beta}) = C_{min}$ in theory.

The minimization of expectation (10) is achieved by operating the established QAOA quantum circuit in Fig.1 several times. Each time the near-optimal values of the parameters $(\vec{\gamma}, \vec{\beta})$ is measured. Subsequently, a classical optimizer is used to optimize the parameters [35]. Eventually, a sequence of unitary quantum gates acting on the initial state of the system will generate a final state close to the ground state $|\psi(\vec{\gamma}^*, \vec{\beta}^*)\rangle$ by repeatedly measuring the quantum state in the computational basis.

D. Overall Procedure of QSLR Algorithm

In the QSLR framework, after decomposition, the original MBP problem is split into a QUBO as (7) solved by QAOA-based quantum solver and a convex problem as (4) decoded with off-the-shelf solvers, such as CPLEX and Gurobi. Once multipliers reach convergence tolerance, MBP feasible solutions are obtained. The entire QSLR algorithm is summarized in the following steps:

1) *Initialization*: Subproblem solutions x^0, y^0 as well as stepsizes and multipliers s^0, λ^0 are initialized.

2) *Binary-Subproblem Solving*: Binary subproblems x_i^{k+1} are solved by a QAOA-based solver.

To achieve the aforementioned QAOA theoretical procedure, a quantum-circuit-based prototype is developed as shown in Fig.1. At the beginning step, the quantum state $|\psi_0\rangle$ is initialized as $|0\rangle^{\otimes n_q}$ in the computational basis.

- **Step 1** n_q Hadamard gates are applied to qubits for superposition, which results in the state:

$$|\psi_1\rangle = H^{\otimes n_q} \otimes |\psi_0\rangle = \frac{1}{\sqrt{2^{n_q}}} \sum_{\Upsilon \in \{0,1\}^{n_q}} |\Upsilon\rangle \quad (12)$$

where H is hardmard gate.

- **Step 2** Evolve with the Hamiltonian and transverse field Hamiltonian by implementing unitary gates $U(E, \gamma)$ and $U(Q, \beta)$, respectively. Then, repeat r_q times with different parameters $(\vec{\gamma}, \vec{\beta})$ to form the state:

$$\begin{aligned} |\psi_2\rangle &= \prod_{k_q=1}^{r_q} (U(Q, \beta) \otimes U(E, \gamma)) \otimes |\psi_1\rangle \\ &= \frac{1}{\sqrt{2^{n_q}}} \prod_{k_q=1}^{r_q} \left(\prod_{j_q=1}^{n_q} e^{j\beta_{r_q} Q_{j_q}} \prod_{l_q=1}^{m_q} e^{j\gamma_{r_q} E_{l_q}} \right) \sum_{\Upsilon \in \{0,1\}^{n_q}} |\Upsilon\rangle \end{aligned} \quad (13)$$

- **Step 3** Measure in the computational basis to compute the expectation of Hamiltonian on (8).
- **Step 4** Use a classical optimization algorithm to compute the minimum value. Then, repeat the above steps. A sufficient number of repetitions will produce a state which represents a close enough solution.

3) *Continuous-Subproblem Solving*: Continuous subproblems are solved by the classical optimizer. After that, a surrogate optimality condition is checked based on the above Steps 2) and 3).

4) *Coordination and Updating*: The coupling constraints are relaxed by introducing Lagrangian multipliers.

5) *Coordination and Updating*: As discussed in Section II.A, the separability of the problem is exploited by relaxing coupling constraints by introducing Lagrangian multipliers. To improve the convergence performance of QSLR, we introduce the contraction-mapping stepsize without requiring the optimal dual value [29]. Appendix B presents how the stepsizes are derived. Multipliers are updated using the stepsizes s^k are set as:

$$s^k = \alpha^k \frac{s^{k-1} \|g(x^{k-1}, y^{k-1})\|}{\|g(x^k, y^k)\|}, 0 < \alpha^k < 1, k = 1, 2, \dots \quad (14)$$

with

$$\alpha^k = 1 - \frac{1}{Mk^p}, p = 1 - \frac{1}{k^r} \quad (15)$$

where $M > 1, 0 < r < 1$.

Update α^k by using (15) firstly. Given that (α_k, x^k, y^k) , update stepsizes s^k . Then, update multipliers λ^{k+1} according to (5). Use the updated λ^{k+1} to minimize the decomposed Lagrangian functions of (3,4).

6) *Feasible solution search*: To obtain feasible solutions, penalty terms $\rho \|g(x, y)\|^2$ are added. The penalty function of the binary part consists of $\rho \|g(x^k, y^k)\|^2$ components, while, that of the continuous part is added as $\rho \|g(x^{k+1}, y^k)\|^2$.

7) *Stopping criteria*: if $\|g(x^k, y^k)\| \leq \xi$ is satisfied, obtain feasible solutions, otherwise go to step 1).

E. Distributed Coordination of Subproblems

Generally, engineering systems are composed of several subsystems with local QPU/CPU. This subsection establishes a D-QSLR scheme adaptive to the practical feature. Meanwhile, it is friendly to privacy protection and better efficiency.

Fig. 2 demonstrates the architecture of D-QSLR, where each local sub-problem is assigned and regulated by the coordinator. Consider a system consisting of one coordinator and n distributed sub-systems with QPU/CPU resources and communication capabilities. Each subsystem is assigned solvers (quantum/classical optimizers) and capable of solving the subproblem. In the physical assignment, each subproblem corresponds to a thread, and the coordinator corresponds to a separate thread to update stepsizes and multipliers.

The entire D-QSLR scheme can be summarized in the following steps:

Step 1: Coordinator updates stepsizes s^k based on received solutions (α^k, x^k, y^k) . Then, updates and broadcasts multipliers λ^{k+1} to all subsystems.

Step 2: For the received λ^{k+1} , minimize decomposed Lagrangian functions of (3, 4). Binary subsystems are optimized by QAOA-based solver to find solutions x^{k+1} , and send them to continuous parts for y^{k+1} updating. Then, a surrogate optimality condition is checked based on the updated binary/continuous subproblems.

Step 3: Check stopping criteria, if it's satisfied, obtain feasible solutions, otherwise go to Step 1.

Within the D-QSLR scheme, each subproblem can be further decomposed into smaller, more computationally manageable subproblems.

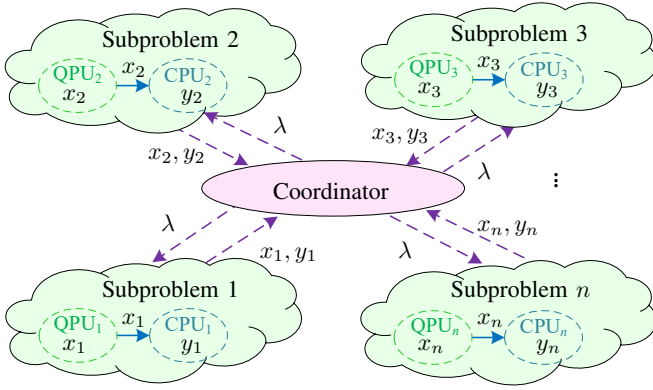


Fig. 2. Coordination architecture of D-QSLR

In summary, QSLR is an enhanced hybrid quantum optimization algorithm which leverages the decomposition and coordination aspect of the SLR method. Specifically, the UC problem is decomposed into the binary subproblems and continuous subproblems which are efficiently solved through quantum and classical computing, respectively, and are subsequently coordinated by updating Lagrangian multipliers. The binary subproblems are first converted into QUBO problems by penalizing constraint violation within the objective functions and then encoded into quantum formulations by mapping them into Ising models.

III. DISTRIBUTED QSLR FOR THE UC PROBLEM

This section establishes a D-QSLR-enabled UC algorithm for scalable and efficient UC decision. After presenting a conventional UC problem formulation in subsection III.A, the quantum UC formulation is devised in subsection III.B. Given the limited number of qubits, the UC problem cannot be solved by using quantum computers in its entirety. To enable the efficient use of quantum computing, the problem is decomposed into more manageable subproblems, which are then efficiently coordinated by using Lagrangian multipliers in subsection III.C.

A. Conventional UC Problem Formulation

The objective of the UC problem is to minimize the total generation cost:

$$\min_{P_{i,t}} \sum_{i=1}^G \sum_{t=1}^T (a_i \cdot P_{i,t}^2 + b_i \cdot P_{i,t} + c_i) \quad (16)$$

where $P_{i,t}$ is the generation of unit i at time t , a_i , b_i and c_i are coefficients of unit i^{th} cost function, T is the number of operation periods, and G is the number of units.

The minimization is subject to the following constraints [36]:

- **System Demand Constraints:** The total generated power should meet the demand D_t at time t :

$$\sum_{i=1}^G P_{i,t} = D_t \quad (17)$$

- **Generation Capacity Constraints:** Generation $P_{i,t}$ of online units are constrained between minimum P_i^{min} and maximum P_i^{max} :

$$x_{i,t} \cdot P_i^{min} \leq P_{i,t} \leq x_{i,t} \cdot P_i^{max} \quad (18)$$

where $x_{i,t}$ is the on/off decision variable of unit i at time t .

- **Ramp-Rate Constraints:** Ramp-rate constraints require that the change of power generation levels between two consecutive time periods does not exceed ramp rates:

$$P_{i,t} - P_{i,t-1} \leq R_i \cdot x_{i,t-1} + V_i \cdot (1 - x_{i,t-1}) \quad (19)$$

$$P_{i,t-1} - P_{i,t} \leq R_i \cdot x_{i,t} + V_i \cdot (1 - x_{i,t}) \quad (20)$$

where R_i is the ramp rate of unit i and V_i is the start-up/shut-down ramp rate.

B. Quantum-Encoded UC model

To exploit the reduction of complexity of the overall problem as well as to make use of the limited quantum computing resources, the decomposition and coordination of the D-QSLR method will be used to relax the system demand constraints and to decompose the resulting relaxed problem into hourly-unit-wise subproblems. The subproblem of unit i at time t is represented as follows:

$$\min_{P_{i,t}, x_{i,t}} \{a_i \cdot P_{i,t}^2 + b_i \cdot P_{i,t} + c_i + \lambda_t(P_{i,t})\} \quad (21)$$

subject to constraints (16)-(18).

At iteration k , after all hourly subproblems (21) are solved for unit i and checked to satisfy a surrogate optimality condition [37], multipliers are updated as:

$$\lambda_t^{k+1} = \lambda_t^k + s^k \cdot g^k(P_{i,t}) \quad (22)$$

where $g^k(P_{i,t}) = \sum_{i'=1, i' \neq i}^G P_{i',t}^{k-1} + P_{i,t}^k - D_t$.

Binary Subproblem. In the following, to solve subproblems by using quantum computers, hourly-unit-wise constraints will be converted to quadratic terms and the resulting quadratic unconstrained mixed-binary subproblems will be decomposed into QUBO subproblems (which will be solved by using quantum computers), and quadratic unconstrained continuous subproblems (which will be solved by using classical computers). The conversion into subproblems through the Ising model amenable for quantum solvers will be explained ahead.

Firstly, the generation capacity inequality constraints (18) need to be transformed into equality constraints. In this case, non-negative slack variables $\xi_{c,t}$, $\xi_{\bar{c},t}$: $\xi_{c,t} \geq 0$, $\xi_{\bar{c},t} \geq 0$ are added to the binary subproblem to convert the inequality constraints into equality constraints as follows:

$$x_{i,t} \cdot P_i^{min} - P_{i,t} + \xi_{c,t} = 0 \quad (23)$$

$$P_{i,t} - x_{i,t} \cdot P_i^{max} + \xi_{\bar{c},t} = 0 \quad (24)$$

Then, based on the quantum reformulation technique introduced in Section II.B, the UC-based quantum functions for (23) and (24) are mapped into the Ising model [38] as follows:

$$\begin{aligned} E_{\underline{c}} &= \rho ||x_{i,t} \cdot P_i^{\min} - P_{i,t} + \xi_{\underline{c},t}||^2 \\ &= \rho P_i^{\min} (P_i^{\min} - 2P_{i,t} + 2\xi_{\underline{c},t})x_{i,t} + \rho (P_{i,t} - \xi_{\underline{c},t})^2 \\ &= \sigma_{\underline{c}(i,t)} \kappa_{i,t} + \zeta_{\underline{c}(i,t)} \end{aligned} \quad (25)$$

$$\begin{aligned} E_{\bar{c}} &= \rho ||P_{i,t} - x_{i,t} \cdot P_i^{\max} + \xi_{\bar{c},t}||^2 \\ &= \rho P_i^{\max} (P_i^{\max} - 2P_{i,t} - 2\xi_{\bar{c},t})x_{i,t} + \rho (P_{i,t} + \xi_{\bar{c},t})^2 \\ &= \sigma_{\bar{c}(i,t)} \kappa_{i,t} + \zeta_{\bar{c}(i,t)} \end{aligned} \quad (26)$$

where $\sigma_{\underline{c}(i,t)}$, $\sigma_{\bar{c}(i,t)}$, $\zeta_{\underline{c}(i,t)}$ and $\zeta_{\bar{c}(i,t)}$ denote coefficients related to the unit output. $E_{\underline{c}}$ and $E_{\bar{c}}$ are energy functions of lower and upper capacity constraints, respectively.

Meanwhile, the constraints (19, 20) are converted into equality constraints by introducing slack variables $\varphi_{\underline{r},t}$, $\varphi_{\bar{r},t}$: $\varphi_{\underline{r},t} \geq 0$, $\varphi_{\bar{r},t} \geq 0$ as follows:

$$P_{i,t} - P_{i,t-1} + R_i x_{i,t-1} + V_i(1 - x_{i,t-1}) - \varphi_{\underline{r},t} = 0 \quad (27)$$

$$R_i x_{i,t} + V_i(1 - x_{i,t}) - P_{i,t-1} + P_{i,t} - \varphi_{\bar{r},t} = 0 \quad (28)$$

Then, the UC-based quantum functions for (27) and (28) are established as follows:

$$\begin{aligned} E_{\underline{r}} &= \rho ||P_{i,t} - P_{i,t-1} + R_i x_{i,t-1} + V_i(1 - x_{i,t-1}) - \varphi_{\underline{r},t}||^2 \\ &= \rho (P_{i,t} - P_{i,t-1} - \varphi_{\underline{r},t} + V_i)^2 + \rho (R_i - V_i)[R_i + V_i + 2(P_{i,t} - P_{i,t-1} - \varphi_{\underline{r},t})]x_{i,t-1} \\ &= \sigma_{\underline{r}(i,t)} \kappa_{i,t} + \zeta_{\underline{r}(i,t)} \end{aligned} \quad (29)$$

$$\begin{aligned} E_{\bar{r}} &= \rho ||R_i x_{i,t} + V_i(1 - x_{i,t}) - P_{i,t-1} + P_{i,t} - \varphi_{\bar{r},t}||^2 \\ &= \rho (P_{i,t} - P_{i,t-1} - \varphi_{\bar{r},t} + V_i)^2 + \rho (R_i - V_i)[R_i + V_i + 2(P_{i,t} - P_{i,t-1} - \varphi_{\bar{r},t})]x_{i,t} \\ &= \sigma_{\bar{r}(i,t)} \kappa_{i,t} + \zeta_{\bar{r}(i,t)} \end{aligned} \quad (30)$$

where $\sigma_{\underline{r}(i,t)}$, $\sigma_{\bar{r}(i,t)}$, $\zeta_{\underline{r}(i,t)}$ and $\zeta_{\bar{r}(i,t)}$ are coefficients related to unit i output at time t ; $E_{\underline{r}}$ and $E_{\bar{r}}$ denote energy functions of lower and upper ramp-rate constraints.

In the quantum language, all constraints are integrated into the quantum energy function. In detail, the total energy function of unit i at time t is represented as a weighted sum of the aforementioned sub-energy formulations as follows:

$$E_{i,t} = \eta_{\underline{c}} \cdot E_{\underline{c}} + \eta_{\bar{c}} \cdot E_{\bar{c}} + \eta_{\underline{r}} \cdot E_{\underline{r}} + \eta_{\bar{r}} \cdot E_{\bar{r}} \quad (31)$$

where, $\eta_{\underline{c}}$ and $\eta_{\bar{c}}$ are weighted coefficients of generation constraints, $\eta_{\underline{r}}$ and $\eta_{\bar{r}}$ denote weighted coefficients of ramp rate constraints.

After the above mapping the binary subproblem to the Hamiltonian, the UC problem is prepared for the quantum optimization algorithm.

Continuous Subproblem. After receiving binary decision values, classical methods are implemented for such subproblems by (16)-(20). To obtain feasible solutions, penalty terms $\rho ||g^k(P_{i,t})||^2$ are added into continuous subproblems.

In the next subsection, the D-QSLR procedure to solve the UC problem is discussed.

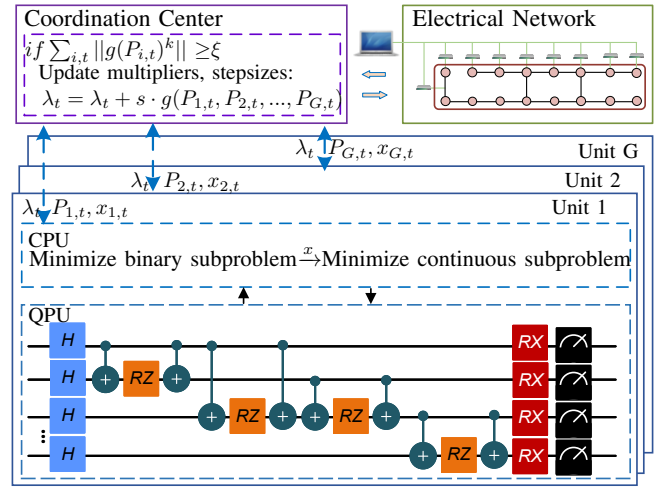


Fig. 3. D-QSLR-based UC architecture

C. D-QSLR-based UC optimization

As shown in Fig. 3, each unit in the distributed operation of the power grid minimizes its operation cost. Then, the entire power generations of all units are regulated to satisfy the network demand. The detailed D-QSLR process of solving the UC problem is described as algorithm 1.

Given that the multipliers and stepsizes at all time sequences $t \in T$ are initialized, the coordinator delivers multipliers λ_t to each corresponding subproblem (i, t) .

For the binary part of the subproblem (i, t) , the energy functions of decision variable $x_{i,t}$ contained constraints are encoded based on (23-31). Then, the expectation value $F_p(\vec{\gamma}, \vec{\beta}) = \langle \psi(\vec{\gamma}, \vec{\beta}) | E_i | \psi(\vec{\gamma}, \vec{\beta}) \rangle$ is optimized and measured by the QAOA-based optimal solver on the local QPU.

Next, the updated $x_{i,t}$, $P_{i,t-1}^k$ and $x_{i,t-1}^k$ are sent to the continuous part of the subproblem (i, t) . The generation $P_{i,t}$ is optimized by classical optimization solver on its local CPU.

After checking the surrogate optimality condition based on the updated $\{P_{i,t}, x_{i,t}\}$, the subproblem (i, t) sends $\{P_{i,t}, x_{i,t}\}$ to the coordinator for updating multipliers. Other subproblems follow the same procedure. Finally, the iterative calculation continues until the convergence of the entire UC problem reaches the tolerance.

D. Practical considerations of D-QSLR-based UC optimization on quantum machine

Current quantum machines are crippled by noise and short of quantum coherence. Therefore, when testing quantum UC algorithm, proper settings of the quantum machine are required which would affect the performance of the quantum UC algorithm as follows:

- Sufficient quantum shots (i.e., the operation number of the quantum circuit) need to be executed to achieve a relatively precise measurement when calculating the binary part of each UC subproblem.
- The depth of the quantum circuit is one of the major factors that determine the performance of the quantum algorithm, because a large depth may cause inaccurate measurement results due to high level noises. A small repeat time often

Algorithm 1: D-QSLR-based UC Algorithm

Initialize: Coordinator: $\lambda_t^0, k, s^0, M, p, r, Iter_{max}, \xi, \|g(P_{i,t})^0\|, N, H$; Subproblem: $a_i, b_i, c_i, P_{i,t}, x_{i,t}, P_{i,t}^{min}, P_{i,t}^{max}, V_i, R_i, \rho$;
while $k \leq Iter_{max}$ and $\sum_{i,t} \|g(P_{i,t})^k\| \geq \xi$ **do**
 Update(Coordinator): $s^k, \alpha^k, \|g(P_{i,t})^k\|, \lambda_t^{k+1}$
 Eq. (14, 15, 22);
 For subproblem (i, t) : Input: $\lambda_t^{k+1}, P_{i,t-1}^k, x_{i,t-1}^k$;
 if binary part then
 Convert constraints to Ising function
 E_i **Eq.** (23-31);
 Prepare computational basis and Hamiltonian;
 Execute QAOA, Optimize: $\vec{\gamma}, \vec{\beta}$;
 Min: $\langle \psi(\vec{\gamma}, \vec{\beta}) | E_i | \psi(\vec{\gamma}, \vec{\beta}) \rangle$, Output: $x_{i,t}^{k+1}$
 else
 Input: $x_{i,t}^{k+1}, P_{i,t-1}^k, x_{i,t-1}^k$;
 Optimize: $L_p(P_{i,t}, x_{i,t})$ **Eq.** (16-21);
 Update: $P_{i,t}^{k+1}$; Send to Coordinator;
 end
end
Result: $P_{i,t}, x_{i,t}$.

enables the reduction of the depth of the QAOA quantum circuit and relieves the stress on the optimization of gate parameters. In general, problems containing more variables and constraints may cause a deeper quantum circuit. Meanwhile, different quantum machines also affect the depth because of different hardware configurations and basic quantum gates. For example, if the Pauli-Z gate doesn't exist, the Hardmard gate and Pauli-X gate are compiled together to realize the function of the nonexistent Pauli-Z gate, which would increase the depth.

IV. NUMERICAL TESTING

In the case studies, we validate the correctness and effectiveness of QSLR and D-QSLR-based UC methods in two typical examples. The scalability of the D-QSLR-based UC method is verified as well. In subsection A, a generalized MBP example that is decomposed into continuous subproblems and binary subproblems is used to demonstrate the correctness and convergence of QSLR and to compare it with that of QADMM in terms of mathematical aspect. In subsection B, a 3-unit 4-hour UC example is used to test the performance of D-QSLR and to compare results with those of classical¹ sequential SLR. The characteristics of each generator are described in Table I. Stepsizing related parameters within D-QSLR are: $M = 50$ and $r = 0.05$. In subsection C, the 24-hour UC problems of the 3-unit, 6-unit, 102-unit, and 1020-unit systems are respectively solved in order to test the scalability of the new method. The aforementioned tests are implemented on the noise-free quantum simulator (statevector) with IBM's Qiskit (0.23.4), Terra (0.16.3), and IBMQ provider (0.11.1). In subsection D, the MBP and 3-unit 4-hour UC cases are executed on the practical quantum machine (*IBMQ_belem*) to

¹Here, the term "classical" is used as opposed in meaning to "quantum."

TABLE I
UNIT DATA FOR 3-UNIT SYSTEM

Unit	P_{max}	P_{min}	a	b	c
1	600	100	0.0020	10	500
2	400	100	0.0025	8	300
3	200	100	0.0050	6	100

TABLE II
OPTIMAL SOLUTIONS OF DIFFERENT QUANTUM METHODS

Method	v	w	t	u	obj	Iteration
QSLR	1	0	0	2	1	2
QADMM	1	0	0	2	1	19

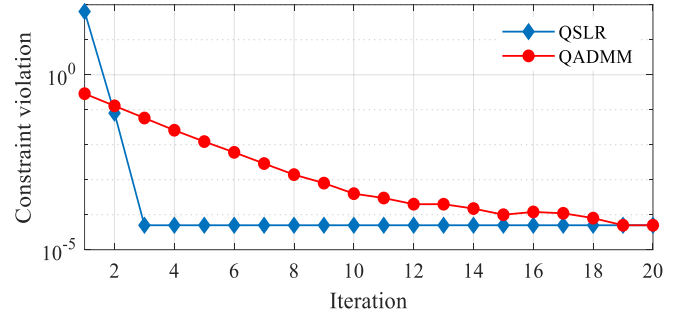


Fig. 4. Convergence violations g_b of different quantum methods

analyze the effect of a noisy quantum environment on QSLR and D-QSLR-based UC methods. The original SLR method is implemented in python 3.7 on a 2.40 GHz PC with an i9-1085H CPU.

A. Demonstration of QSLR by using an MBP Example.

This subsection compares the performance of QSLR with the existing QADMM. An MBP example [25] is as follows:

$$\begin{aligned}
 \min_{v, w, t \in \mathbb{X}, u \in \mathbb{R}} \quad & v + w + t + 5(u - 2)^2, \\
 \text{s.t.} \quad & v + w = 1, \\
 & v + w + t \geq 1, \\
 & v + 2w + t + u \leq 3.
 \end{aligned} \tag{32}$$

For QSLR, the initial parameters are set as $g^0 = 100$ and $s^0 = 0.019$. For QADMM, the y decision and the penalty coefficient for constraints are set 1000 and 900. The initial value of ρ is 1001 [25].

Based on the devised QSLR procedure, the MBP problem is decomposed into a binary subproblem $L_b = v + w + t + \lambda(v + 2w + t)$ solved by the QAOA method and a continuous subproblem $L_c = 5(u - 2)^2 + \lambda(u + \zeta_b - 3)$ solved by the classical optimizer, where ζ_b denotes a slack variable. Fig. 4 shows the reduction of constraint violations g_b within QSLR and QADMM. The optimal solutions of QSLR and QADMM are shown in Table II.² Fig. 5 presents the optimal objective values L_b of the binary subproblem with different

²For this small example, optimal solutions are known. Generally, feasible solutions are always obtained.

initial multipliers. Figs. 6 and 7 show the optimal binary objective values L_b and constraint violations g_b of QSLR under different initial stepsizes, respectively. For the small problem instance, the following insights are drawn:

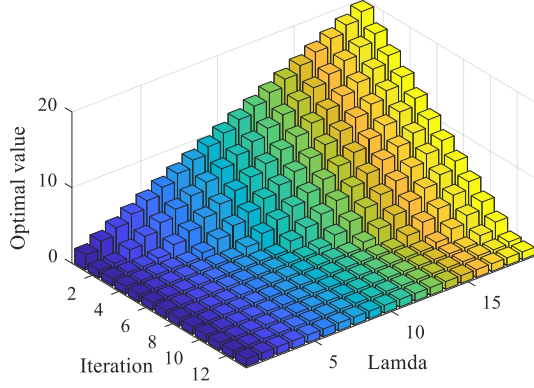


Fig. 5. Optimal values of binary subproblem L_b under different initial multipliers

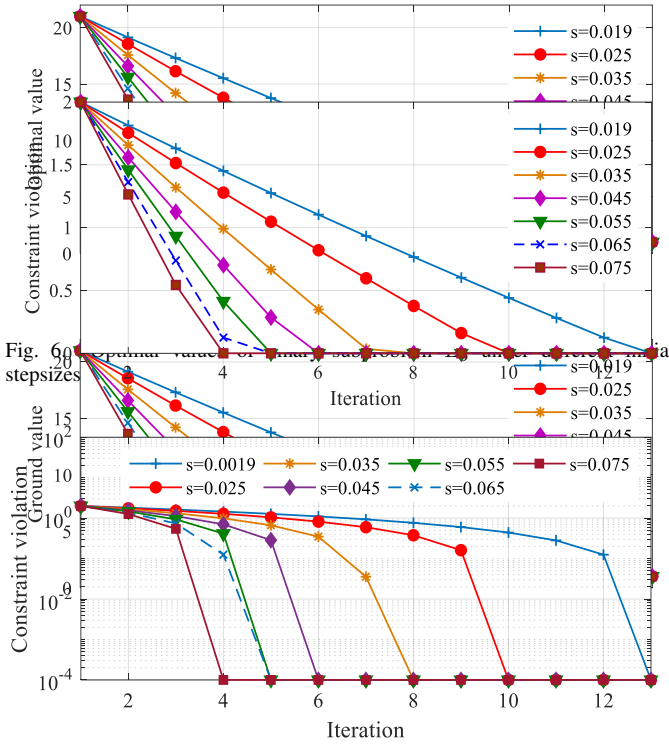


Fig. 7. Constraint violations g_b under different initial stepsizes

- QSLR obtains the same optimal solution as existing QADMM algorithm. For example, as shown in the Table.II, the results yield: $[v, w, t] = [1, 0, 0]$, $u = 2$ and the objective is 1, which validates the correctness of QSLR.
- QSLR has a better convergence performance than the existing QADMM method. For instance, QSLR takes 2 iterations to find the optimal solutions, which is less 15 iterations than that of QADMM. The faster convergence of QSLR is attributed to the intrinsic “contraction-mapping” nature of stepsizes, which overcome difficulties due to the presence of binary variables.

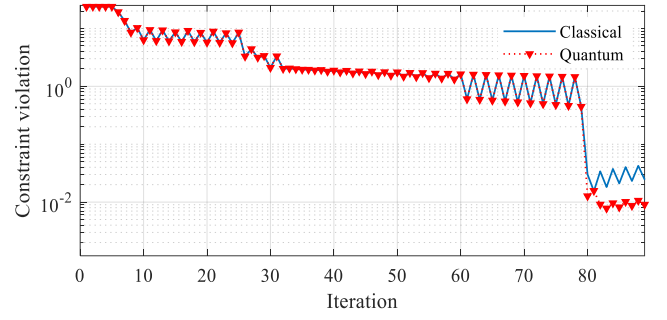


Fig. 8. Constraint violation performance of 3-unit system

- The initial values of multipliers can affect the convergence performance of the binary subproblem objective values. For instance, as shown in Fig.5, more iterations are required for QSLR to find the eventual optimal value of binary subproblem with the increase of multiplier. When the multiplier is initialized as 20, 13 iterations are required to reach the eventual optimal solution. However, 3 iterations are needed under $\lambda = 1$.
- A proper stepsize benefits to the convergence of QSLR. For instance, as shown in Fig.6, 13 iterations are required to achieve the optimal value for the binary subproblem when the stepsize is initialized as 0.019. However, when the stepsize is increased to 0.045, the iterations are decreased to 8. Nevertheless, the beneficial effects decrease when the stepsize is increased to 0.075. In detail, the iterations under $s = 0.075$ are identical to those under $s = 0.065$. Meanwhile, the stepsize has the same impacts on the convergence performance of the constraint violation in Fig.7.

B. Demonstration of D-QSLR-based UC method.

TABLE III
OPTIMAL POWER GENERATION OF 3-UNIT SYSTEM 100MW)

Unit	T_1	T_2	T_3	T_4	P_{tot}	cost
1	1.62/1.60	3.40/3.40	1.00/1.00	0.00/0.00	6.02/6.00	83.24/83.02
2	4.00/4.00	4.00/4.00	2.19/2.20	1.00/1.00	11.19/11.20	111.06/111.06
3	2.00/2.00	2.00/2.00	2.00/2.00	1.00/1.00	7.00/7.00	52.50/52.50
D_t	7.60	9.40	5.20	2.00	24.20	246.80/246.58

Note: SLR/D-QSLR

This subsection mainly demonstrates the correctness and effectiveness of D-QSLR. The performance of D-QSLR is compared with that of classical SLR. Based on the devised D-QSLR-based UC optimization procedure, the 3-unit problem is decomposed into 3-unit subproblems by relaxing the demand constraint. Further, each unit's subproblem is divided into 4 hourly subproblems to overcome the difficulties associated with a limited number of qubits. Specifically, each on/off decision variable is optimized by the quantum method; power generation variables are optimized by the classical optimizer. In Table III, the optimal results of the 3-unit UC system are described. In Fig. 8, the reduction of demand constraint violations at each hour is shown. It can be seen that:

TABLE IV
PARTIAL RESULT OF 1020-UNIT SYSTEM (100MW)

Unit	T_1	T_2	T_3	T_4	T_5	T_6
1	1.0021	2.1574	3.9590	3.3612	1.5585	1.0594
2	2.9042	4.0001	4.0012	3.9120	2.7107	3.9107
550	2.0000	2.0001	2.0001	1.9265	1.3252	1.9253
551	1.0000	1.1005	2.9013	1.1013	0.9972	1.0000
1019	1.1947	2.3956	3.5973	3.0162	1.8136	2.8475
1020	2.0001	2.0000	2.0001	1.9194	1.3181	1.9182
D_t	1713.7	2318.6	3136.1	2594.4	1657.5	2154.3
Unit	T_7	T_8	T_9	T_{10}	T_{11}	T_{12}
1	1.1071	2.9082	4.7102	5.6803	3.8795	2.0794
2	4.0000	4.0002	4.0002	4.0016	3.2238	2.0238
550	2.0001	2.0003	2.0002	2.0010	2.0003	2.0000
551	1.0996	2.9007	4.7026	5.6728	3.8721	2.0719
1019	2.3156	3.5158	4.0002	4.0016	3.2139	2.0138
1020	2.0000	2.0003	2.0021	2.0017	2.0002	1.8597
D_t	2127.5	2943.9	3640.6	3970.3	3091.2	2036.8
Unit	T_{13}	T_{14}	T_{15}	T_{16}	T_{17}	T_{18}
1	3.5599	5.3620	3.7850	1.9848	3.3075	4.2022
2	2.8021	4.0021	3.2061	2.0060	3.2061	4.0000
550	2.0000	2.0023	2.0001	1.4001	2.0001	2.0002
551	1.1980	3.0001	1.5526	1.0000	1.0000	1.8965
1019	2.8020	4.0021	3.1986	1.9985	3.1985	4.0000
1020	2.0000	2.0022	2.0002	1.4000	2.0000	2.0001
D_t	2439.4	3460.9	2676.3	1664.2	2501.1	3075.7
Unit	T_{19}	T_{20}	T_{21}	T_{22}	T_{23}	T_{24}
1	6.0001	5.8004	4.0003	2.2002	1.0143	2.8145
2	4.0023	4.0021	4.0020	2.8021	2.8002	4.0002
550	2.0002	2.0023	2.0002	2.0000	2.0000	2.0001
551	3.6987	5.5007	3.9821	2.1820	1.0042	2.8045
1019	4.0022	4.0020	3.4005	2.2004	1.5964	2.7966
1020	2.0023	2.0021	2.0022	2.0002	1.9384	2.0003
D_t	3689.6	3960.5	3293.2	2272.5	1762.4	2790.6

- For the 3-unit UC problem, D-QSLR obtains the same optimal solution as that of the classical SLR method. For instance, under high power demands at hour 1, generators 1, 2, and 3 are combined with the grid to support the power consumption (see Table III). When the power demand is decreased to 200 MW at hour 4, the output of generator 1 is decreased to 0 MW, which means it's disconnected from the grid.
- D-QSLR method has a similar convergence performance of demand constraint as the classical method. As shown in Fig.8, the demand violation is gradually approaching zero. At the 80th iteration, a penalty function is added, and the demand violation eventually reaches convergence tolerance.

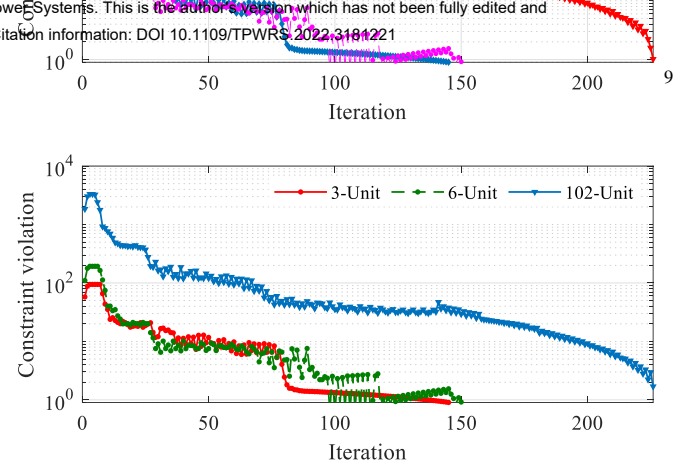


Fig. 9. Constraint violation performance of different system scales

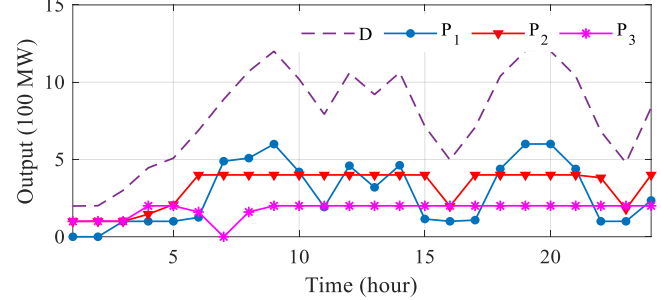


Fig. 10. Power scheduling of 3-unit system

C. Performance of D-QSLR in Large UC Problems.

This subsection explores the performance of the D-QSLR method in real-scale UC problems. The number of generators is increased from 3 to 6, 102, and 1020, respectively, and the time horizon is increased to 24 hours. The optimization process shares the same procedure with that in Part B. Results for a few selected units of a 1020-unit problem are shown in Table IV. Fig. 9 shows the reduction of constraint violation for the 3-unit, 6-unit, and 102-unit systems. Fig. 10 presents the generation scheduling curve of the 3-unit system. A few insights are:

- D-QSLR obtains the optimized generators' output results for larger problems. For example, as shown in Fig. 10 and Table IV, the generation of each generator is scheduled to balance the power demand. .
- D-QSLR retains powerful convergence abilities with the increase of the system scale. For instance, as shown in Fig. 9, when the unit scale is 102, the demand constraint violation is swiftly decreased after several iterations. This is because the contraction-mapping stepsizes provide more flexible adjustments of multipliers.
- Although the currently available quantum computers cannot directly handle 24-hour subproblems because of the qubit limitation. We developed a scheme to further decompose the unit-wise subproblems into several hourly subproblems amenable for quantum computing. As a result of efficient QC-enabled subproblem showing as well as the efficient coordination of the SLR method, the overall D-QSLR demonstrated a potential to solve large-scale UC problems.

D. QSLR Test on Noisy-contained Quantum Machine.

In this subsection, we explore the QSLR and UC-oriented D-QSLR performance on a noisy quantum machine

IBMQ_belem. The QSLR-based MBP example is tested at first. The probability distributions of the MBP problem with different quantum shots are shown in Fig. 11. In this figure, each quantum state indicates one possible binary result of $[v, w, t]$. Table V presents the calculation time of MBP on the quantum machine. Then, the accuracy of D-QSLR on *IBMQ_belem* is tested by simulating the 3-unit 4-hour UC problem. Table VI shows the probabilities of the correct unit on/off decisions that *IBMQ_belem* and quantum simulator achieve. A quantum circuit of the UC problem is established in Fig. 12. The quantum circuits for solving the aforementioned problems are implemented using the IBM Qiskit package (see Appendix A for detailed representations of quantum gates).

- Fig. 11 illustrates the effects of different quantum shots on the measurement results. The quantum state in the x -axis presents the potential binary values of the MBP problem. The probability of each quantum state is obtained by measuring the quantum circuit. The quantum states with the highest probability denote the results of the binary subproblem of the MBP case. Due to the noisy environment, current quantum devices are required to run multiple times to get a relatively accurate result. From Fig. 11, we can see that when quantum shots are above 3024, the optimal results can get a higher measurement probability. For example, when the shots are 5024, the optimal results obtain 30.8%. However, when the quantum shots are equal to 1024, optimal results fail to get the highest probability measurement.
- Table VI further validates that a high accuracy performance of D-QSLR on the quantum machine can be guaranteed in a noisy environment. To demonstrate the effectiveness, each subproblem is simulated for 100 times on *IBMQ_belem* and the quantum simulator, respectively. For example, x_1 at T_2 and x_2 at T_1 are achieved with 99% accuracy probability. Besides, the rest of the on/off decision results are reached with a 100% accuracy probability. This is because sufficient shots and a low-depth quantum circuit (the depth is 22 in Fig.12) relieve the effect of noise disturbance on the measurement and enable the quantum machine to get a relatively accurate probability of the optimal quantum state.
- Quantum computing provides a potential direction for calculation speed. Table V presents the simulation number and time consumption of each procedure on *IBMQ_belem*. As shown in Table V, the current quantum device spends almost 12 seconds on the quantum calculation per iteration. Specifically, the running time/iteration takes 12.4 seconds when shots are 8024, which means 1.5 milliseconds are cost per shot. However, because of the limit of existing quantum devices, queue and repeating shots take the majority of the time.

It's noted that quantum computers with hundreds of qubits to solve the UC problem in its entirety are not available yet. Meanwhile, today's quantum machines are limited to support quantum circuits with several tens of depths because of the short coherence time of noisy quantum hardware and other roadblocks including large error correction overhead, a limited number of qubits, and limited connectivity between qubits. For example, quantum computing needs to repeat the

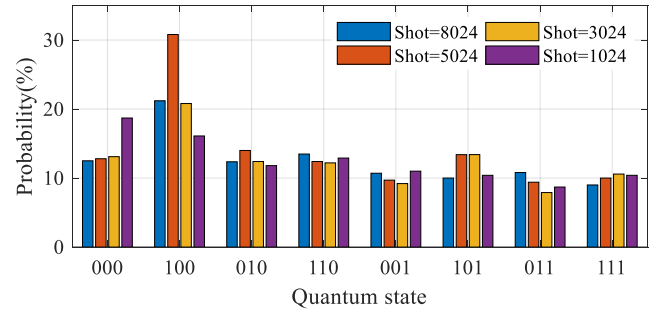


Fig. 11. Probability distribution of the MBP example under different shots

TABLE V
CALCULATION PERFORMANCE OF THE MBP EXAMPLE

Number of Iterations	204	129	198	202
Shots/Iteration	8024	5024	3024	1024
Run time (s/Iteration)	12.4	12.2	11.9	12.9
Validating time (ms/Iteration)	785	789	914	755

Note: Number of Iterations: the total iterations of gate parameters optimization; Shot/Iteration: the measurement number of each gate parameters optimization; Run time: the time consumption of each optimization; Validating time: the time consumption to verify that the quantum circuit is able to be run.

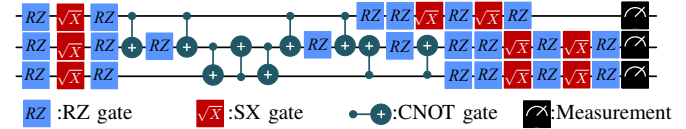


Fig. 12. Quantum circuit for D-QSLR based UC problem

calculation thousands of times to catch an accurate result which is shown in Subsection D. During this process, time costs consist of time delay, rebuilding circuits, repeating calculations, and so on. Fortunately, more improved quantum machines (e.g., the 27-qubit IBM quantum machine with 10 times accuracy improvement in 2020, and the 1000-plus-qubit quantum computer by 2023 [39]) are under development. The swift growth of quantum machine capacities is promising to support the devised algorithm in near future, thereby enabling the developed QSLR method, supported by solid convergence principles, to solve larger instances with an increased speed. The new method thus opens the door for efficient power

TABLE VI
PROBABILITY OF DECISION RESULTS IN 3-UNIT SYSTEM

	T_1	T_2	T_3	T_4
x_1	1/1	1/1	1/1	0/0
Prob.	100%/100%	99%/100%	100%/100%	100%/100%
x_2	1/1	1/1	1/1	1/1
Prob.	99%/100%	100%/100%	100%/100%	100%/100%
x_3	1/1	1/1	1/1	1/1
Prob.	100%/100%	100%/100%	100%/100%	100%/100%

Note: *IBMQ_belem* /simulator. x denotes the binary results on the near-term quantum machine (*IBMQ_belem*) and the quantum simulator. Prob. denotes the frequency of correct binary results in 100 times.

system quantum optimization.

V. CONCLUSION

This paper is the first of the kind to develop a Quantum-based Surrogate Lagrangian Relaxation (QSLR) algorithm suitable for large-scale UC problems. Although the number of qubits currently available is not sufficient to solve the UC problem in its entirety, the decomposition and coordination feature of the SLR method enables subproblem solving through quantum computers.

Case studies demonstrate the efficiency of QSLR in handling the MBP problem compared with QADMM. Moreover, the distributed QSLR solution scheme for UC optimization also enables privacy protection for community grids and power system safety and security management. Despite existing gaps for practical applications of D-QSLR-based UC optimization algorithm in system operation and planning due to qubit limitation, short coherence times, and noises on today's quantum computers, D-QSLR lays a solid foundation for power system optimization on the next-generation quantum computers.

APPENDIX A

QUANTUM GATES USED IN THE PAPER

The mathematical formulation of the Hadamard gate H is:

$$H = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

The mathematical formulation of the squared X gate $\sqrt{X}$ (or SX) is:

$$\sqrt{X} = \frac{1}{2} \begin{bmatrix} 1+i & 1-i \\ 1-i & 1+i \end{bmatrix}$$

The mathematical formulations of single-qubit rotation gates R_X and R_Z can be expressed as:

$$R_X(\theta) = \begin{bmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{bmatrix}$$

$$R_Z(\theta) = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}$$

The mathematical formulation of the controlled not gate CNOT (or CX) denotes:

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

APPENDIX B

CONTRACTION-MAPPING STEPSIZE

To guarantee the convergence to the optimal multipliers without requiring the optimal dual value, the distances between multipliers at consecutive iterations are required to be decreased as follows [29],

$$\|\lambda_t^{k+1} - \lambda_t^k\| = \alpha^k \|\lambda_t^k - \lambda_t^{k-1}\| \quad (33)$$

where, $0 < \alpha^k < 1$. The stepsizing formulation satisfying (33) can be derived by using (5),

$$\|s^k \cdot g(x^k, y^k)\| = \alpha^k \|s^{k-1} \cdot g(x^{k-1}, y^{k-1})\| \quad (34)$$

Here, stepsizes s^k satisfying (34) can be uniquely achieved, if norms of surrogate subgradients are not zero. Therefore, norms of surrogate subgradients are dependent on a strict positivity requirement: $g(x^k, y^k) > 0$.³ Since s^k and s^{k-1} are positive scalars, and norms of surrogate subgradients are strictly positive, (34) implies

$$s^k = \alpha^k \frac{s^{k-1} \|g(x^{k-1}, y^{k-1})\|}{\|g(x^k, y^k)\|}, 0 < \alpha^k < 1, k = 1, 2, \dots \quad (35)$$

REFERENCES

- [1] P. Zhang, *Networked Microgrids*. Cambridge, UK: Cambridge University Press, 2021.
- [2] W. Wan, M. A. Bragin, B. Yan, Y. Qin, J. Philhower, P. Zhang, and P. B. Luh, "Distributed and asynchronous active fault management for networked microgrids," *IEEE Transactions on Power Systems*, vol. 35, no. 5, pp. 3857–3868, 2020.
- [3] C. Wang and Y. Fu, "Fully parallel stochastic security-constrained unit commitment," *IEEE Transactions on Power Systems*, vol. 31, no. 5, pp. 3561–3571, 2016.
- [4] S. Boyd, N. Parikh, and E. Chu, *Distributed Optimization and Statistical Learning Via the Alternating Direction Method of Multipliers*, ser. Foundations and Trends in Machine Learning Series. Now Publishers, 2011.
- [5] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, "A survey of distributed optimization and control algorithms for electric power systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.
- [6] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, 2011.
- [7] Y. Zhou and P. Zhang, "Noise-resilient quantum machine learning for stability assessment of power systems," *IEEE Transactions on Power Systems*, under revision, 2021.
- [8] Z. Tang, Y. Qin, Z. Jiang, W. O. Krawec, and P. Zhang, "Quantum-secure microgrid," *IEEE Transactions on Power Systems*, vol. 36, no. 2, pp. 1250–1263, 2021.
- [9] Z. Tang, P. Zhang, and W. O. Krawec, "A quantum leap in microgrids security: The prospects of quantum-secure microgrids," *IEEE Electrification Magazine*, vol. 9, no. 1, pp. 66–73, 2021.
- [10] Z. Jiang, Z. Tang, Q. Yanyuan, C. Kang, and P. Zhang, "Quantum internet for resilient electric grids," *International Transactions on Electrical Energy Systems*, vol. 31, 04 2021.
- [11] Y. Zhou, P. Zhang, and F. Feng, "Noisy-intermediate-scale quantum electromagnetic transients program," *IEEE Transactions on Power Systems*, Accepted, 2022.
- [12] E. Farhi and A. W. Harrow, "Quantum supremacy through the quantum approximate optimization algorithm," *arXiv preprint arXiv:1602.07674v2*, 2019.
- [13] J. Preskill, "Quantum computing and the entanglement frontier," *arXiv preprint arXiv:1203.5813*, 2012.
- [14] Y. Zhou, F. Feng, and P. Zhang, "Quantum electromagnetic transients program," *IEEE Transactions on Power Systems*, vol. 36, no. 4, pp. 3813–3816, 2021.
- [15] F. Feng, Y. Zhou, and P. Zhang, "Quantum power flow," *IEEE Transactions on Power Systems*, vol. 36, no. 4, pp. 3810–3812, 2021.
- [16] T. Albash and D. A. Lidar, "Adiabatic quantum computation," *Rev. Mod. Phys.*, vol. 90, p. 015002, Jan 2018.
- [17] E. Zahedinejad and A. Zaribafian, "Combinatorial optimization on gate model quantum computers: A survey," *arXiv preprint arXiv:1708.05294*, 2017.
- [18] A. Gilliam, S. Woerner, and C. Gondiulea, "Grover adaptive search for constrained polynomial binary optimization," *Quantum*, vol. 5, p. 428, Apr 2021.
- [19] D. Wang, O. Higgott, and S. Brierley, "Accelerated variational quantum eigensolver," *Phys. Rev. Lett.*, vol. 122, p. 140504, Apr 2019.
- [20] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, "A variational eigenvalue solver on a photonic quantum processor," *Nature Communications*, vol. 5, no. 1, pp. 1–7, 2014.

³If surrogate subgradients are zero, a feasible solution is obtained, in which case, an iteration is skipped.

- [21] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv:1411.4028*, 2014.
- [22] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices," *Phys. Rev. X*, vol. 10, p. 021067, Jun 2020.
- [23] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm applied to a bounded occurrence constraint problem," *arXiv preprint arXiv:412.6062*, 2015.
- [24] J. Bae, P. Alsing, D. Ahn, and W. Miller, "Quantum circuit optimization using quantum karnaugh map," *Scientific Reports*, vol. 10, 2020.
- [25] C. Gambella and A. Simonetto, "Multiblock ADMM heuristics for mixed-binary optimization on classical and quantum computers," *IEEE Transactions on Quantum Engineering*, vol. 1, pp. 1–22, 2020.
- [26] N. Nikmehr, P. Zhang, and M. Bragin, "Quantum distributed unit commitment," *IEEE Transactions on Power Systems*, pp. 1–1, 2022.
- [27] M. A. Bragin, B. Yan, and P. B. Luh, "Distributed and asynchronous coordination of a mixed-integer linear system via surrogate lagrangian relaxation," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 1191–1205, 2021.
- [28] P. Ramanan, M. Yildirim, E. Chow, and N. Gebraeel, "An asynchronous, decentralized solution framework for the large scale unit commitment problem," *IEEE Transactions on Power Systems*, vol. 34, no. 5, pp. 3677–3686, 2019.
- [29] M. Bragin, P. Luh, J. Yan, N. Yu, and G. Stern, "Convergence of the surrogate Lagrangian relaxation method," *Journal of Optimization Theory and Applications*, vol. 164, pp. 173–201, 04 2014.
- [30] E. National Academies of Sciences, D. Sciences, I. Board, C. Board, C. Computing, M. Horowitz, and E. Grumbling, *Quantum Computing: Progress and Prospects*. National Academies Press, 2019. [Online]. Available: <https://books.google.com/books?id=jjiPDwAAQBAJ>
- [31] K. Tanahashi, S. Takayanagi, T. Motohashi, and S. Tanaka, "Application of Ising machines and a software development for Ising machines," *Journal of the Physical Society of Japan*, vol. 88, no. 6, p. 061010, 2019.
- [32] A. Lucas, "Ising formulations of many NP problems," *Frontiers in physics*, vol. 2, p. 5, 2014.
- [33] G. Pagano, A. Bapat, P. Becker, K. S. Collins, A. De, P. W. Hess, H. B. Kaplan, A. Kyprianidis, W. L. Tan, C. Baldwin, L. T. Brady, A. Deshpande, F. Liu, S. Jordan, A. V. Gorshkov, and C. Monroe, "Quantum approximate optimization of the long-range Ising model with a trapped-ion quantum simulator," *Proceedings of the National Academy of Sciences*, vol. 117, no. 41, pp. 25 396–25 401, 2020.
- [34] M. X. Goemans and D. P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *J. ACM*, vol. 42, no. 6, p. 1115–1145, nov 1995. [Online]. Available: <https://doi.org/10.1145/227683.227684>
- [35] A. Montanaro, "Quantum algorithms: an overview," *npj Quantum Information*, vol. 2, no. 1, Jan 2016.
- [36] Q. P. Zheng, J. Wang, and A. L. Liu, "Stochastic optimization for unit commitment—a review," *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 1913–1924, 2015.
- [37] J. Wu, P. Luh, Y. Chen, M. Bragin, and B. Yan, "A novel optimization approach for sub-hourly unit commitment with large numbers of units and virtual transactions," 05 2021.
- [38] B. A. Cipra, "An introduction to the Ising model," *American Mathematical Monthly*, vol. 94, pp. 937–959, 1987.
- [39] C. Adrian, "IBM promises 1000-qubit quantum computer-a milestone-by 2023," Sep 2020.



Fei Feng (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering from Southwest Jiaotong University, Chengdu, China, in 2016, and the M.S. degree in electrical engineering from Chongqing University, Chongqing, China, in 2019. He is currently working towards the Ph.D. degree in electrical engineering with the department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY, USA. His research interests include networked microgrids, power flow, state estimation and quantum computing.



Peng Zhang (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of British Columbia, Vancouver, BC, Canada, in 2009. He is a Full Professor of Electrical and Computer Engineering, and a SUNY Empire Innovation Professor at Stony Brook University, New York. He has a joint appointment at Brookhaven National Laboratory as a Staff Scientist. Previously, he was a Centennial Associate Professor and a Francis L. Castleman Associate Professor at the University of Connecticut, Storrs, CT, USA. He was a System Planning Engineer at BC Hydro and Power Authority, Canada, during 2006–2010. His research interests include AI-enabled smart grids, quantum-engineered power grids, networked microgrids, power system stability and control, cybersecurity, and formal methods and reachability analysis. Dr. Zhang is an individual member of CIGRÉ. He is an Editor for the IEEE Transactions on Power Systems, the IEEE Power and Energy Society Letters, and the IEEE Journal of Oceanic Engineering.



Mikhail A. Bragin (Member, IEEE) is an Assistant Research Professor in the Department of Electrical and Computer Engineering at the University of Connecticut. His research work has been supported by the U.S. National Science Foundation, BNL, MISO, ISO-NE, ABB, CESMII, and AFRL. His research is geared toward solving complex technical and societal challenges within smart grids, supply chains, and artificial intelligence. Accordingly, his research interests include operations research as well as AI- and quantum computing-supported mathematical optimization, including power system optimization, grid integration of renewables (wind and solar), energy-based operation optimization of distributed energy systems, stochastic scheduling within manufacturing systems, and machine learning through deep neural networks.



Yifan Zhou (Member, IEEE) received the B.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2014, and the Ph.D. degree in electrical engineering from Tsinghua University, Beijing, China, in 2019. She is currently a Postdoctoral Researcher with Stony Brook University, Stony Brook, NY, USA. Her research interests include microgrid stability and control, formal methods and reachability analysis, and quantum computing.