SNPSG Journal of Intelligent Unmanned Systems
2026, Vol. 1, No. 1, Article 13
DOI: https://doi.org/10.65218/jius.2026.13

REVIEW ARTICLE

Quantum Machine Learning for Robotics: A Comprehensive Review of Algorithms, Applications, and Future Directions

Hassen Nigatu[1], Gaokun Shi [1], Jituo Li [1,2,*], Guodong Lu [1] and Howard Li [3]
1 Zhejiang University Robotics Institute (Yuyao Robot Research Center), Yuyao, Ningbo, Zhejiang China
2 Institute of Design Engineering, Department of Mechanical Engineering, Zhejiang University, Zhejiang, China
3 Department of Electrical and Computer Engineering, University of New Brunswick, Fredericton, NB, Canada
* Corresponding author(s): Jituo Li
Email(s): jituo_li@zju.edu.cn

Received: 4 January 2026 | Accepted: 22 January 2026 | Published online: 30 June 2026

Abstract: The convergence of quantum computing and robotics represents a paradigm shift in autonomous systems design, promising to overcome fundamental computational limitations that have constrained classical robotic capabilities. This comprehensive review surveys the rapidly evolving field of quantum machine learning (QML) for robotics, systematically analyzing 107 peer-reviewed contributions spanning quantum algorithms, machine learning techniques, and their robotic applications. We provide a detailed historical perspective on the field's development, from the early theoretical foundations established by Grover's search algorithm and quantum reinforcement learning proposals to current experimental implementations on noisy intermediate-scale quantum (NISQ) devices. The review systematically examines five major application domains: (1) quantum reinforcement learning for navigation and control, (2) quantum search and path planning algorithms, (3) quantum-enhanced localization and mapping, (4) quantum optimization for inverse kinematics and task allocation, and (5) multi-robot coordination and swarm intelligence. For each domain, we present detailed algorithmic formulations, computational complexity analyses, and critical assessments of experimental results. Our analysis reveals that while provable quantum advantage in robotics remains largely prospective, recent demonstrations show promising results: quantum deep reinforcement learning achieving 40% parameter reduction in navigation tasks, Grover-based kinematic optimization demonstrating up to 93x speedups, and quantum annealing providing practical solutions for multi-robot routing. We identify key technical challenges including barren plateaus in variational circuits, noise-induced performance degradation, and the quantum-classical integration bottleneck. Finally, we outline a strategic roadmap for quantum robotics research, distinguishing near-term opportunities in hybrid algorithms from long-term goals requiring fault-tolerant quantum computation. This paper serves as both an authoritative technical reference and a comprehensive guide for researchers working at the intersection of quantum computing and autonomous systems.

Keywords: Quantum machine learning, quantum reinforcement learning, quantum robotics, variational quantum algorithms, path planning, NISQ computing, quantum optimization, multi-robot systems

I. Introduction

A. Motivation and Background

The field of robotics stands at a computational crossroads. As autonomous systems confront increasingly complex environments and tasks, classical computing paradigms strain under the weight of exponential state spaces, real-time decision requirements, and the curse of dimensionality [1–3]. Modern robotic applications—from autonomous vehicles navigating urban environments to surgical robots performing delicate procedures—demand computational capabilities that push against fundamental limits of classical algorithms [4, 5].

Consider the path planning problem for a robot with d degrees of freedom operating in an environment with n obstacles. The configuration space has dimension d, and naive discretization with resolution r yields r^d possible configurations. For a 6-DOF manipulator with $r = 100$ resolution points per dimension, this represents 10^{12} configurations—a search space that grows exponentially with the robot's complexity [3, 6].

Quantum computing offers a fundamentally different computational paradigm that may address these limitations [7, 8]. By exploiting quantum mechanical phenomena including superposition, entanglement, and interference, quantum algorithms can achieve provable speedups for specific problem classes [9, 10]. The seminal work by Biamonte et al. [11] established quantum machine learning as a distinct field, while subsequent research has explored its application to robotics [12–14].

B. Quantum Computing Fundamentals for Robotics

The fundamental unit of quantum information is the qubit, which generalizes the classical bit by existing in a coherent superposition of basis states:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{1}$$

where $\alpha, \beta \in \mathbb{C}$ are complex probability amplitudes satisfying the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. Upon measurement in the computational basis, the qubit collapses to $|0\rangle$ with probability $|\alpha|^2$ or $|1\rangle$ with probability $|\beta|^2$.

For an n-qubit system, the state space dimension grows exponentially as 2^n:

$$|\Psi\rangle = \sum_{i=0}^{2^n-1} c_i\,|i\rangle\,, \quad \sum_{i=0}^{2^n-1} |c_i|^2 = 1 \tag{2}$$

This exponential scaling of the Hilbert space dimension underlies the potential quantum advantage for problems involving high-dimensional search or optimization [7].

Quantum operations are represented by unitary matrices U satisfying $U^\dagger U = UU^\dagger = I$. The set of single-qubit gates forms a continuous group, with important examples including the Pauli matrices:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \; Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \; Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

and parameterized rotation gates:

$$R_\alpha(\theta) = e^{-i\theta\alpha/2} = \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}\alpha$$

where $\alpha \in \{X, Y, Z\}$, with X, Y, and Z being the Pauli matrices defined above. These Pauli matrices correspond to rotations around the x-, y-, and z-axes of the Bloch sphere, respectively.

Two-qubit entangling gates create quantum correlations without classical analogue. The controlled-NOT (CNOT) gate:

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3}$$

flips the target qubit conditioned on the control qubit being in state $|1\rangle$. Applied to the state $|+\rangle|0\rangle$, it produces the maximally entangled Bell state:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{4}$$

Entanglement enables quantum speedups by allowing quantum algorithms to exploit correlations across exponentially many computational paths simultaneously [7].

C. Historical Development of Quantum Robotics

The historical trajectory of quantum robotics reveals a field in rapid transition from theoretical speculation to experimental validation. We identify four distinct phases of development, illustrated in Fig. 1, which provides a comprehensive visualization of key milestones from the theoretical foundations of 1996 through the experimental validation era of 2022–2025.

Phase I: Theoretical Foundations (1996-2008). The field's origins trace to Grover's seminal search algorithm [9], which demonstrated that quantum computers can search an unstructured database of N items in $O(\sqrt{N})$ queries, compared to the classical lower bound of $\Omega(N)$. This quadratic speedup immediately suggested applications in robotic path planning, where searching configuration spaces is a fundamental operation [10].

Benioff [15] introduced the concept of quantum robots—autonomous systems with quantum computational capabilities—establishing a theoretical framework for the field. Dong et al. [16] proposed one of the first quantum reinforcement learning algorithms, demonstrating provable advantages in specific learning scenarios through quantum amplitude amplification.

Phase II: Algorithmic Development (2008-2018). This period saw the development of core quantum machine learning algorithms with potential robotic applications. Lloyd et al. [17] proposed quantum algorithms for supervised and unsupervised learning, while Rebentrost et al. [18] demonstrated exponential speedups for support vector machines given quantum random access memory (qRAM).

The comprehensive review by Dunjko and Briegel [12] identified reinforcement learning as particularly amenable to quantum enhancement, outlining multiple pathways to quantum advantage including quantum exploration, quantum policy evaluation, and quantum model-based planning. Schuld and colleagues [19, 20] established rigorous mathematical foundations for quantum machine learning, including the theory of quantum feature maps and kernel methods.

Phase III: NISQ-Era Adaptation (2018-2022). The recognition that fault-tolerant quantum computing remains distant spurred development of algorithms suitable for noisy intermediate-scale quantum (NISQ) devices [21]. Variational quantum algorithms (VQAs) emerged as the dominant paradigm [22–24], combining parameterized quantum circuits with classical optimization.

The position paper by Petschnigg et al. [13] systematically outlined quantum computing applications in robotic science through the sense-think-act cycle, identifying specific opportunities in perception, artificial intelligence, kinematics, and system diagnosis. This foundational work established quantum robotics as a distinct research domain and anticipated many current research directions.

Skolik et al. [25] implemented variational quantum deep Q-learning, demonstrating comparable performance to classical deep Q-networks with improved parameter efficiency. Chen et al. [26] developed variational quantum circuits for deep reinforcement learning, establishing practical implementations on quantum simulators.

Phase IV: Experimental Validation (2022-Present). Recent years have witnessed the first experimental demonstrations of quantum algorithms on robotic-relevant problems at meaningful scale. Hohenfeld et al. [27] demonstrated quantum deep reinforcement learning for wheeled robot navigation, achieving optimal policies with 40% fewer trainable parameters than classical baselines.

Sinha et al. [28] developed Nav-Q for autonomous vehicle navigation in realistic simulators, while Lokossou et al. [29] extended quantum RL to humanoid robot control. Nigatu et al. [30] demonstrated Grover-based kinematic optimization achieving up to 93x speedups over classical optimizers for manipulator planning. The comprehensive review by Yan et al. [14] synthesized these developments, establishing quantum robotics as a mature research area.

D. Scope and Contributions

This review provides a comprehensive synthesis of quantum machine learning for robotics, integrating theoretical foundations with practical applications across the robotic software stack. Our specific contributions include:

(i) A systematic review methodology based on PRISMA guidelines [31], rigorously analyzing 107 peer-reviewed references across theoretical, algorithmic, and applied dimensions

(ii) Comprehensive technical coverage of quantum reinforcement learning (Section IV), quantum planning and search (Section V), quantum localization and mapping (Section VI), and quantum optimization (Section VII), with detailed algorithmic formulations and complexity analyses

DOI: https://doi.org/10.65218/jius.2026.13

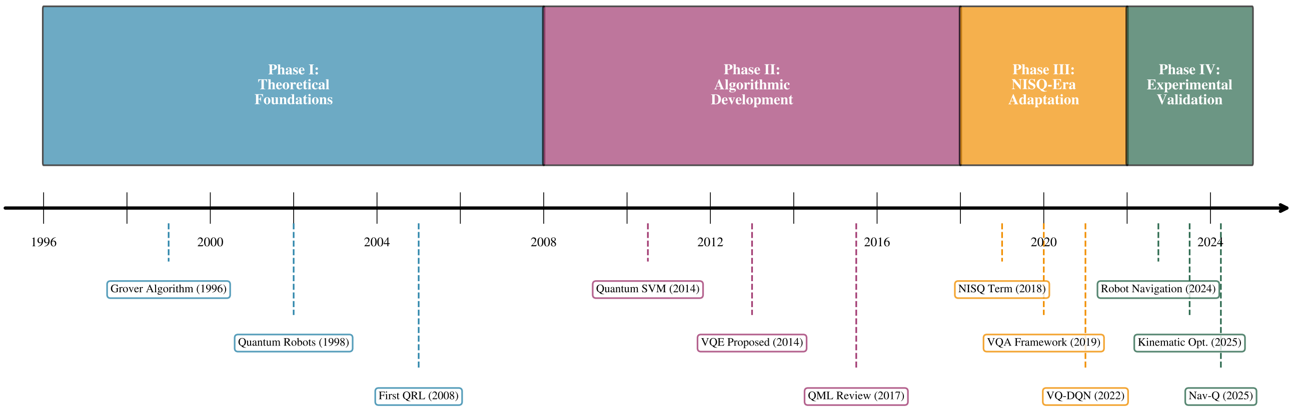

Fig. 1 Historical development of quantum robotics spanning four distinct phases. Phase I (1996–2008) established theoretical foundations with Grover's algorithm and early quantum reinforcement learning proposals. Phase II (2008–2018) focused on algorithmic development including quantum SVMs and the VQE framework. Phase III (2018–2022) adapted algorithms for NISQ devices with variational approaches. Phase IV (2022–present) has seen experimental validation on real quantum hardware and robotic simulators. Key milestones are annotated below the timeline, showing the progression from abstract theory to practical demonstration.

(iii) Critical assessment of quantum hardware progress across superconducting, photonic, and trapped-ion platforms (Section VIII), with specific implications for robotic applications

(iv) Identification of fundamental research challenges including barren plateaus, noise resilience, and scalability (Section X), with a strategic roadmap for future development

(ii) Integration of advances through 2025, providing a state-of-the-art perspective on the field's trajectory based on carefully validated, high-quality studies

E. Paper Organization

The remainder of this paper is organized as follows. Section II details our systematic review methodology, including search strategy, inclusion criteria, and corpus characteristics. Section III establishes the quantum computing and machine learning foundations essential for understanding the subsequent technical content.

The core technical sections examine specific application domains: Section IV analyzes quantum reinforcement learning for robotic navigation and control; Section V covers quantum search and path planning algorithms; Section VI addresses quantum-enhanced localization and mapping; and Section VII examines quantum optimization for control and multi-robot coordination. Section VIII assesses the current state of quantum hardware and its implications for robotic applications. Section IX surveys emerging application domains. Section X discusses fundamental challenges and outlines future research directions. Finally, Section XI presents conclusions and a strategic outlook for the field.

II. Systematic Review Methodology

A. Guiding Framework and Protocol

This review employs a rigorous systematic methodology adapted for interdisciplinary research at the quantum-robotics interface. We followed the PRISMA 2020 guidelines [31] for transparent reporting and implemented established protocols for engineering literature reviews [32]. The methodology was designed to ensure comprehensive coverage while maintaining rigor, reproducibility, and relevance to both quantum computing and robotics research communities.

Our approach addresses the unique challenges of reviewing an emerging, interdisciplinary field characterized by rapid publication cycles, diverse venues (spanning physics, computer science, and engineering), and varying levels of technical maturity. We balanced inclusiveness with quality assessment, recognizing that significant contributions may appear as preprints before formal journal publication [33].

B. Search Strategy and Database Selection

We conducted exhaustive searches across major electronic databases including IEEE Xplore, ACM Digital Library, Web of Science, Scopus, arXiv, and Google Scholar. The search strategy employed a structured Boolean query optimized for sensitivity and specificity:

```
("quantum machine learning" OR "QML" OR
"quantum reinforcement learning" OR
"quantum robotics" OR "quantum robot" OR
"quantum autonomous")
AND
("robotics" OR "autonomous systems" OR
"robot control" OR "robot learning" OR
"path planning" OR "navigation" OR
"manipulation" OR "SLAM")
AND
(2015-2025)
```

DOI: https://doi.org/10.65218/jius.2026.13

The temporal range (2015-2025) captures the period of most rapid development in quantum robotics, beginning with the maturation of quantum machine learning theory and extending through current experimental demonstrations.

We supplemented database searches with backward and forward snowballing [34] to identify seminal works and recent developments that might not be captured by database indexing. Grey literature including technical reports, preprints, and industry white papers was included when providing unique insights, particularly for hardware developments where peer review cycles lag behind rapid progress.

C. Inclusion and Exclusion Criteria

Studies were included based on the following criteria: publication between January 2015 and December 2025; peer-reviewed journal articles, conference proceedings, or high-impact preprints (determined by citation rate > 5 citations/year or acceptance at top venues); primary focus on quantum algorithms with direct robotic applications or clearly transferable insights; technical contributions including theoretical analysis, simulation results, or hardware implementations; and clear methodological description enabling assessment of validity and reproducibility.

Exclusion criteria eliminated tutorials, opinion pieces, and purely speculative works without technical substance; duplicate publications (retaining the most complete version); non-English publications without English abstracts; works focusing solely on quantum-inspired classical algorithms without quantum implementation; and studies with insufficient methodological detail for quality assessment.

The complete flow of our systematic review process is illustrated in Fig. 2, which follows the PRISMA 2020 guidelines for transparent reporting. This diagram shows the progression from initial database searches through screening, eligibility assessment, and final inclusion, providing full transparency into our selection methodology.

D. Data Extraction and Quality Assessment

For each included study, we extracted detailed metadata using a standardized protocol adapted from Petersen et al. [35]. The extraction template captured bibliographic data (authors, year, venue, citation count); algorithm classification (gate-based vs. annealing, variational vs. fault-tolerant); resource requirements (qubit count, circuit depth, gate count); hardware platform (simulator, superconducting, photonic, trapped-ion, or annealer); problem domain (navigation, manipulation, planning, SLAM, optimization, coordination); evaluation metrics (success rate, speedup, parameter efficiency, sample complexity); comparison baseline (classical algorithm compared against); and limitations acknowledged (noise sensitivity, scalability constraints, assumptions).

Quality assessment followed a modified Newcastle-Ottawa Scale adapted for computational studies, evaluating algorithmic soundness, experimental rigor, baseline comparisons, and reproducibility.

E. Corpus Characteristics

Our search identified 712 initial records, which after deduplication and screening yielded 107 high-quality publications for in-depth analysis. The selection rate (15.0%) reflects our commitment to including only well-validated contributions with clear technical merit.

The temporal and thematic distribution of our reviewed corpus is presented in Fig. 3. As shown in Fig. 3(a), the temporal distribution reveals exponential growth in the field: 12 publications (2015-2017), 18 publications (2018-2019), 29 publications (2020-2021), 31 publications (2022-2023), and 17 publications (2024-2025). This growth pattern reflects the field's accelerating maturation, with the slight decrease in 2024-2025 attributable to the recency of these publications and ongoing peer review processes.

Fig. 3(b) presents the thematic distribution across research areas. Quantum reinforcement learning dominates with 24 publications (22.4%), reflecting the natural synergy between quantum computing and sequential decision-making problems. Quantum planning and search follows with 21 publications (19.6%), leveraging Grover's algorithm for configuration space exploration. Quantum optimization accounts for 19 publications (17.8%), addressing task allocation and trajectory optimization. Hardware and implementation studies comprise 16 publications (15.0%), essential for understanding practical constraints. Quantum localization and SLAM represents 12 publications (11.2%), while theoretical foundations account for 15 publications (14.0%).

Publication venues span quantum computing (Nature, Physical Review, Quantum, npj Quantum Information), robotics (ICRA, IROS, IEEE Transactions on Robotics), machine learning (NeurIPS, ICML), and interdisciplinary journals (Science Robotics, Nature Machine Intelligence).

III. Quantum Computing and Machine Learning Foundations

A. Quantum Algorithmic Paradigms

The theoretical foundations of quantum robotics rest on several algorithmic paradigms, each offering different types of computational advantages [7, 8]. The potential speedups achievable through quantum algorithms are illustrated in Fig. 4, which contrasts classical and quantum computational complexity for problems central to robotics.

a) Grover's Search Algorithm

Grover's algorithm [9] provides the foundational speedup for unstructured search problems central to many robotic applications. Given a search space of N elements with M marked solutions, the algorithm finds a solution with high probability using $O(\sqrt{N/M})$ oracle queries, compared to the classical $\Omega(N/M)$ lower bound.

The algorithm operates by iteratively applying the Grover operator:

$$G = (2\left|\psi\right\rangle\left\langle\psi\right| - I)(I - 2\left|w\right\rangle\left\langle w\right|) \quad (5)$$

where $\left|\psi\right\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}\left|x\right\rangle$ is the uniform superposition and $\left|w\right\rangle$ represents marked solutions. The geometric interpretation views this as rotation in a two-dimensional subspace spanned by marked and unmarked states [7].

For robotic path planning, Grover's algorithm can search over discretized configuration spaces. If the configuration space has $N = r^d$ points (resolution r, dimension d), the quantum approach requires $O(r^{d/2})$ oracle calls compared to classical $O(r^d)$, yielding speedup $r^{d/2}$ [30, 36]. This relationship is quantified in Fig. 4(b), which shows how the theoretical speedup factor scales with the robot's degrees of freedom.

 DOI: https://doi.org/10.65218/jius.2026.13

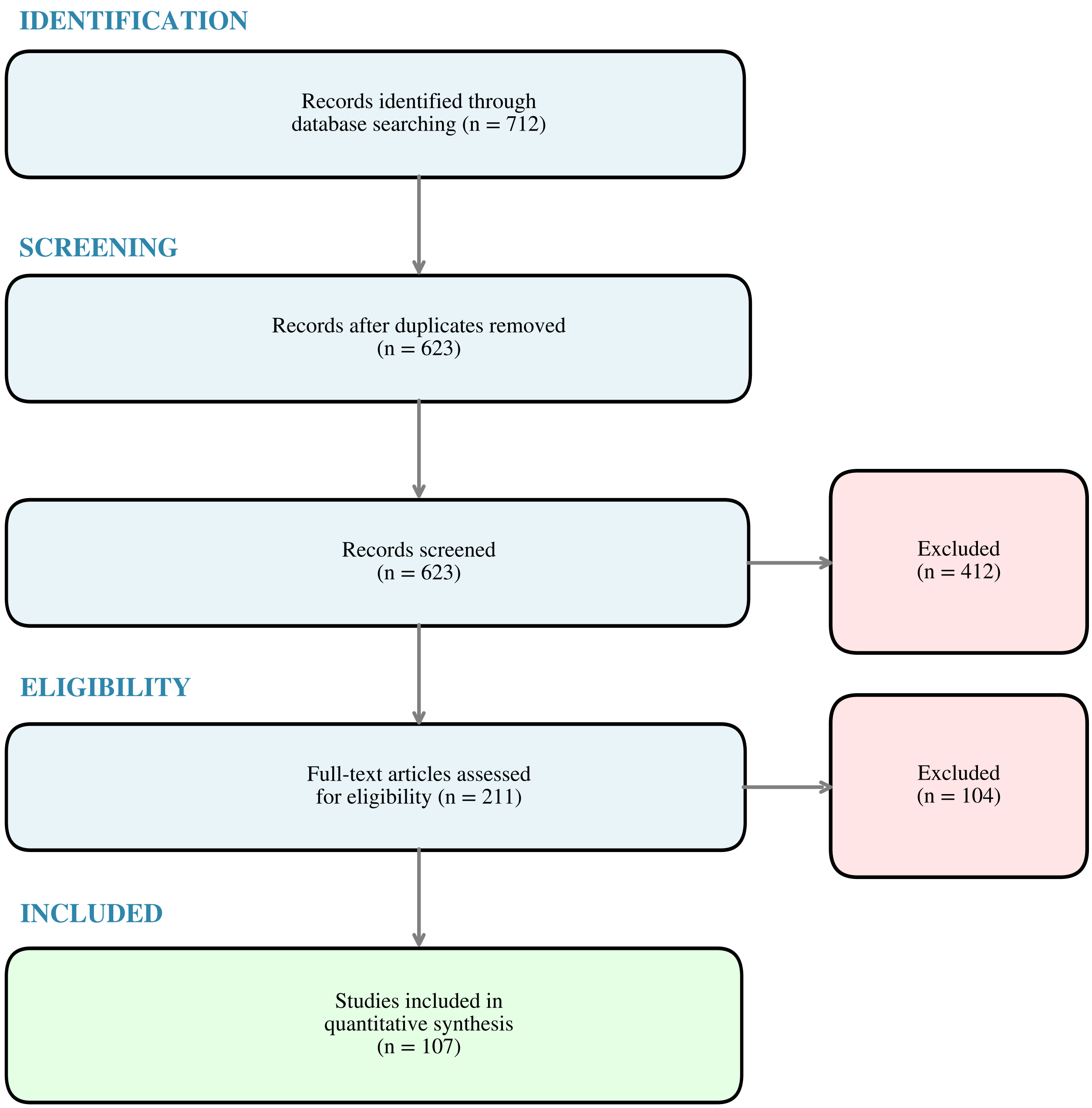

Fig. 2 PRISMA flow diagram illustrating our systematic review methodology. From 712 initially identified records, duplicates were removed yielding 623 unique records for screening. After title and abstract screening excluded 412 records and full-text assessment excluded 104 articles, 107 high-quality studies were included in the final quantitative synthesis. The 15.0% inclusion rate reflects our commitment to rigorous quality assessment.

b) Quantum Amplitude Amplification

Amplitude amplification [37] generalizes Grover's algorithm to arbitrary initial distributions. Given a quantum algorithm $\mathcal{A}$ that produces a superposition with amplitude a on good states, amplitude amplification boosts the success probability to near-unity using $O(1/a)$ applications of $\mathcal{A}$ and its inverse.

The amplification operator is:

$$Q = -\mathcal{A}S_0\mathcal{A}^{-1}S_\chi \tag{6}$$

where S_0 reflects about $|0\rangle$ and S_χ reflects about good states. This technique underlies quantum speedups in Monte Carlo estimation [38], policy evaluation in reinforcement learning [39], and sampling-based motion planning [40].

c) Quantum Walks

Quantum walks [41, 42] provide an alternative paradigm achieving speedups for graph-structured problems. The discrete-time quantum walk on a graph $G = (V, E)$ evolves a walker state $|v, d\rangle$ (vertex v, direction d) according to:

$$U = S \cdot (C \otimes I_V) \tag{7}$$

where C is a coin operator and S is a shift operator.

DOI: https://doi.org/10.65218/jius.2026.13

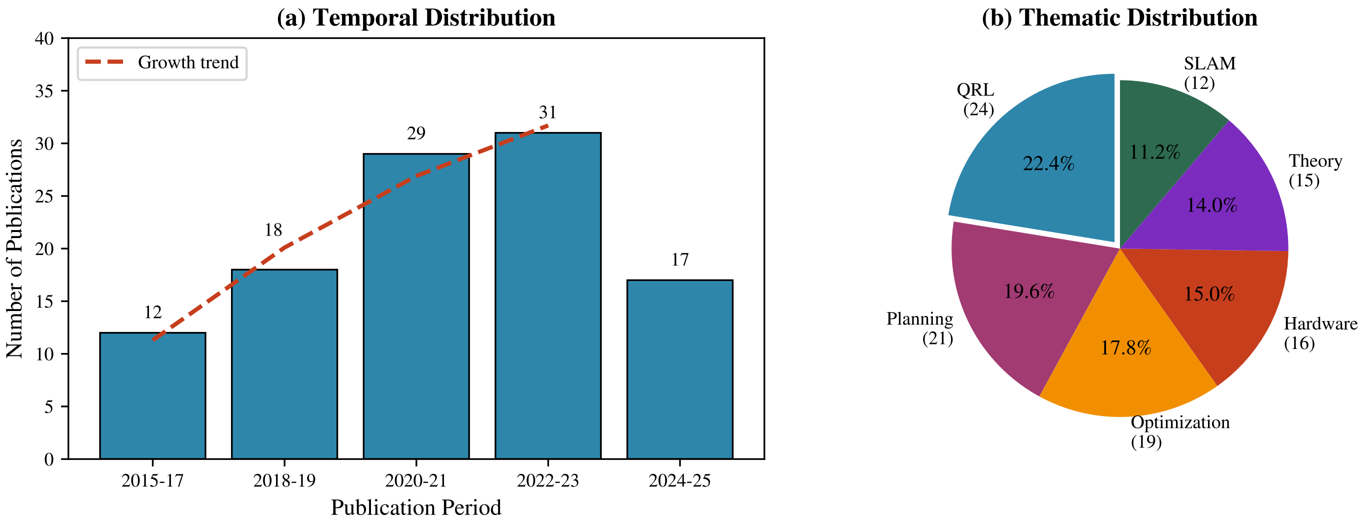

Fig. 3 Publication analysis of reviewed literature (n=107). (a) Temporal distribution showing exponential growth from 2015–2017 (12 publications) through 2022–2023 (31 publications), with the dashed line indicating the growth trend. The slight decrease in 2024–2025 reflects ongoing peer review processes for recent submissions. (b) Thematic distribution revealing quantum reinforcement learning as the dominant research area (22.4%), followed by planning algorithms (19.6%) and optimization (17.8%). These proportions reflect both the natural applicability of quantum algorithms to specific problem classes and the maturity of different subfields.

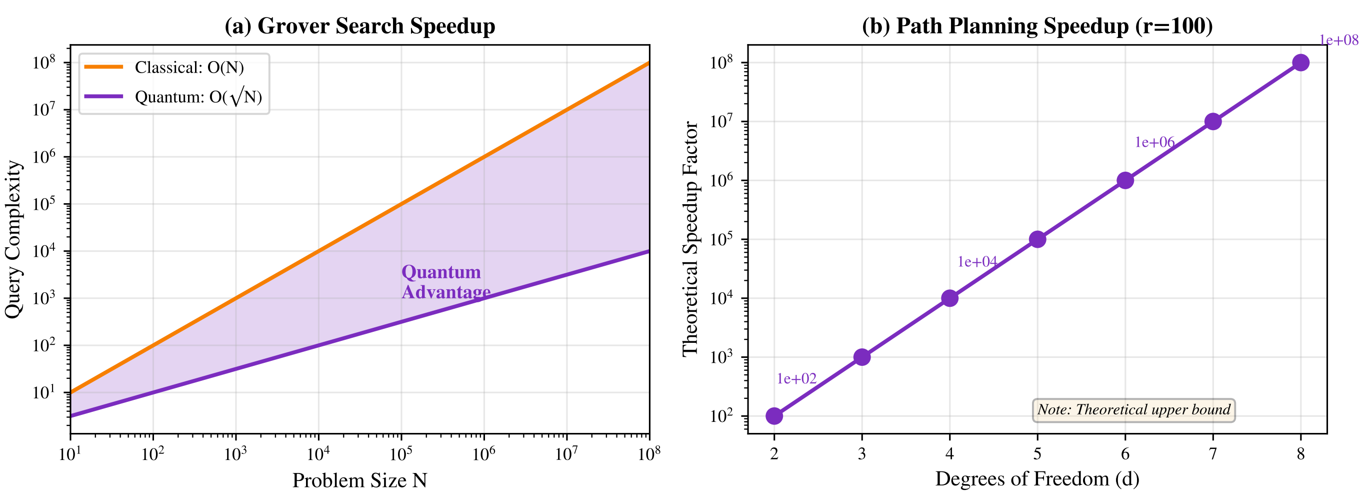

Fig. 4 Quantum computational advantage analysis for robotic applications. (a) Grover search speedup showing the quadratic quantum advantage: classical search requires $O(N)$ queries while quantum search achieves $O(\sqrt{N})$, with the shaded region indicating the quantum advantage zone. For large problem sizes ($N > 10^6$), this translates to speedups exceeding 1000×. (b) Theoretical path planning speedup as a function of robot degrees of freedom (DOF) for resolution $r = 100$. A 6-DOF manipulator can theoretically achieve $10^6\times$ speedup, though practical realization depends on oracle construction efficiency. The annotation emphasizes this represents a theoretical upper bound.

Ambainis [10] showed quantum walks achieve $O(N^{2/3})$ query complexity for element distinctness, improving on classical $O(N)$. For robotic applications, quantum walks can accelerate graph search in navigation [41] and multi-agent coordination [43].

B. Quantum Machine Learning Foundations

a) Quantum Feature Maps and Kernel Methods

Quantum feature maps encode classical data into quantum states, enabling access to exponentially large feature spaces [44, 45]. A feature map $\phi : \mathcal{X} \to \mathcal{H}$ maps classical input $x \in \mathbb{R}^d$ to a quantum state $|\phi(x)\rangle$ in Hilbert space $\mathcal{H}$.

Common encoding strategies include:

Angle encoding: Each feature encodes as a rotation angle:

$$|\phi(x)\rangle = \bigotimes_{i=1}^{d} R_Y(x_i)\,|0\rangle = \bigotimes_{i=1}^{d} \left(\cos\frac{x_i}{2}\,|0\rangle + \sin\frac{x_i}{2}\,|1\rangle\right) \tag{8}$$

Amplitude encoding: The data vector encodes as amplitudes:

$$|\phi(x)\rangle = \frac{1}{\|x\|}\sum_{i=0}^{N-1} x_i\,|i\rangle \tag{9}$$

requiring $\lceil \log_2 N \rceil$ qubits but $O(N)$ gates for state preparation [46].

DOI: https://doi.org/10.65218/jius.2026.13

Input: Oracle O_f marking M solutions among N elements
Output: A marked element x^* with probability $\geq 1 - \epsilon$
$n \leftarrow \lceil \log_2 N \rceil$;
Prepare uniform superposition: $|\psi\rangle \leftarrow H^{\otimes n} |0\rangle^{\otimes n}$;
$k \leftarrow \lfloor \frac{\pi}{4} \sqrt{N/M} \rfloor$;
for $i = 1$ **to** k **do**
  Apply oracle: $|\psi\rangle \leftarrow O_f |\psi\rangle$;
  Apply diffusion: $|\psi\rangle \leftarrow (2 |+\rangle^{\otimes n} \langle +|^{\otimes n} - I) |\psi\rangle$;
end
Measure $|\psi\rangle$ in computational basis to obtain x^*;
return x^*;

Algorithm 1: Grover's Search Algorithm

IQP (Instantaneous Quantum Polynomial) encoding: Employs entangling gates for expressive feature maps:

$$|\phi(x)\rangle = U_\phi(x) |0\rangle^{\otimes n} = \prod_{S \subseteq [n]} e^{i\phi_S(x) Z_S} H^{\otimes n} |0\rangle^{\otimes n} \tag{10}$$

where $Z_S = \bigotimes_{i \in S} Z_i$ [45, 47].

The quantum kernel is defined as the inner product of encoded states:

$$K(x, x') = |\langle \phi(x') | \phi(x) \rangle|^2 \tag{11}$$

Havlíček et al. [45] demonstrated that quantum kernels can be classically intractable to compute while remaining efficiently estimable on quantum hardware, providing a potential path to quantum advantage for classification tasks relevant to robotic perception.

b) Quantum Neural Networks

Quantum neural networks (QNNs) [48–50] use parameterized quantum circuits as trainable function approximators. A typical QNN architecture consists of an encoding layer $U_{\text{enc}}(x)$ that maps classical input to quantum state, variational layers $U_{\text{var}}(\boldsymbol{\theta})$ containing parameterized gates for learning, and measurement that extracts classical output via observable expectation.

The variational ansatz typically alternates single-qubit rotations and entangling layers:

$$U_{\text{var}}(\boldsymbol{\theta}) = \prod_{l=1}^{L} W_l(\boldsymbol{\theta}_l) V_l \tag{12}$$

where $W_l(\boldsymbol{\theta}_l) = \bigotimes_{i=1}^{n} R_Y(\theta_{l,i}^Y) R_Z(\theta_{l,i}^Z)$ are parameterized single-qubit gates and V_l are entangling layers (e.g., CNOT ladders) [51, 52].

The output is obtained by measuring an observable O:

$$f(x; \boldsymbol{\theta}) = \langle \psi(x; \boldsymbol{\theta}) | O | \psi(x; \boldsymbol{\theta}) \rangle \tag{13}$$

where $|\psi(x; \boldsymbol{\theta})\rangle = U_{\text{var}}(\boldsymbol{\theta}) U_{\text{enc}}(x) |0\rangle$.

Abbas et al. [53] analyzed the expressive power of QNNs, showing they can represent functions with favorable properties for learning. Cong et al. [49] introduced quantum convolutional neural networks (QCNNs) that preserve locality structure, achieving advantages for classifying quantum phases of matter.

c) Quantum Convolutional Neural Networks

QCNNs [49, 54] adapt the convolutional architecture to quantum circuits through convolutional layers consisting of two-qubit gates applied to neighboring pairs:

$$U_{\text{conv}} = \prod_{i \text{ even}} U_{i,i+1}(\boldsymbol{\theta}) \prod_{i \text{ odd}} U_{i,i+1}(\boldsymbol{\theta}') \tag{14}$$

and pooling layers that measure a subset of qubits and apply controlled operations:

$$U_{\text{pool}} = \prod_i \text{Measure}_i \cdot \text{ControlledOp}_{i \to j} \tag{15}$$

Pesah et al. [54] proved that QCNNs avoid barren plateaus—the gradient variance decays at most polynomially rather than exponentially:

$$\text{Var}\left[\frac{\partial \langle O \rangle}{\partial \theta_i}\right] \geq \frac{c}{n^k} \tag{16}$$

for constants $c, k > 0$, providing analytical guarantees for trainability.

C. Variational Quantum Algorithms

Variational quantum algorithms (VQAs) [22, 23] represent the dominant paradigm for NISQ-era quantum computing. The general framework, illustrated in Fig. 5 and explained in detail with Algorithm 2, optimizes a cost function:

$$C(\boldsymbol{\theta}) = \langle 0 | U^\dagger(\boldsymbol{\theta}) H U(\boldsymbol{\theta}) | 0 \rangle \tag{17}$$

using classical optimization over parameters $\boldsymbol{\theta}$.

a) Variational Quantum Eigensolver (VQE)

VQE [24, 55] finds ground states of Hamiltonians by minimizing energy:

$$E_0 \leq E(\boldsymbol{\theta}) = \langle \psi(\boldsymbol{\theta}) | H | \psi(\boldsymbol{\theta}) \rangle \tag{18}$$

Algorithm 2 describes the detail procedure of VQE.

Input: Hamiltonian $H = \sum_i c_i P_i$ (Pauli decomposition), ansatz $U(\boldsymbol{\theta})$, optimizer, tolerance ϵ
Output: Approximate ground state energy E_0 and parameters $\boldsymbol{\theta}^*$
Initialize parameters $\boldsymbol{\theta} \sim \mathcal{N}(0, \sigma^2)$;
while $|\Delta E| > \epsilon$ **do**
  Prepare $|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta}) |0\rangle^{\otimes n}$;
  for *each Pauli term* P_i **do**
    Estimate $\langle P_i \rangle$ via repeated measurement;
  end
  Compute $E(\boldsymbol{\theta}) = \sum_i c_i \langle P_i \rangle$;
  Compute gradient $\nabla_{\boldsymbol{\theta}} E$ via parameter-shift rule;
  Update $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} E$;
end
return $E(\boldsymbol{\theta}), \boldsymbol{\theta}$;

Algorithm 2: Variational Quantum Eigensolver (VQE)

 DOI: https://doi.org/10.65218/jius.2026.13

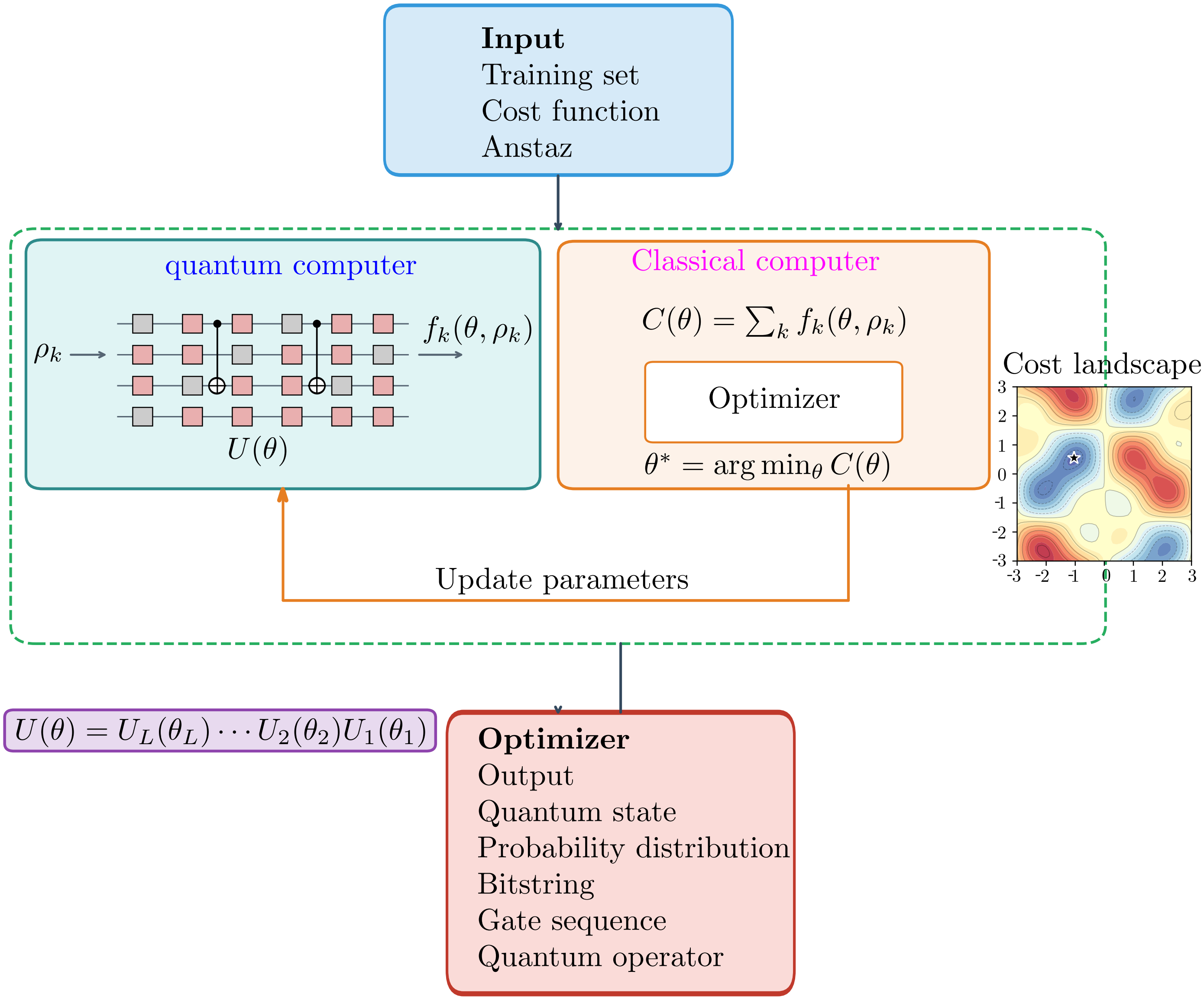

Fig. 5 Variational quantum algorithm (VQA) architecture for robotic applications. Classical input $x \in \mathbb{R}^d$ (e.g., robot state, sensor readings) is encoded into a quantum state via the encoding unitary $U_{\text{enc}}(x)$. The variational circuit $U_{\text{var}}(\boldsymbol{\theta})$ applies parameterized quantum gates whose parameters $\boldsymbol{\theta}$ are optimized through a classical feedback loop. The lower portion shows a simplified parameterized quantum circuit with single-qubit rotation gates (purple boxes) and entangling CNOT gates. The measurement output $\langle O \rangle$ feeds into a classical optimizer that updates parameters using the gradient $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$, creating a hybrid quantum-classical optimization loop.

b) Quantum Approximate Optimization Algorithm (QAOA)

QAOA [56–58] tackles combinatorial optimization by alternating problem and mixer Hamiltonians:

$$|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle = \prod_{p=1}^{P} e^{-i\beta_p H_M} e^{-i\gamma_p H_C} |+\rangle^{\otimes n} \tag{19}$$

where H_C encodes the cost function and $H_M = \sum_i X_i$ is the transverse field mixer.

For a maximum cut problem on graph $G = (V, E)$:

$$H_C = \sum_{(i,j)\in E} \frac{1}{2}(1 - Z_i Z_j) \tag{20}$$

Zhou et al. [57] analyzed QAOA performance, showing that depth-P QAOA achieves approximation ratio improving with P, with theoretical guarantees for specific problem classes.

c) Parameter Optimization

VQA parameters are optimized using gradient-based or gradient-free methods. The parameter-shift rule [20, 59] enables exact gradient computation:

$$\frac{\partial C}{\partial \theta_i} = \frac{1}{2}\left[C\left(\theta_i + \frac{\pi}{2}\right) - C\left(\theta_i - \frac{\pi}{2}\right)\right] \tag{21}$$

The quantum natural gradient [60] incorporates geometric information:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta F^{-1}(\boldsymbol{\theta}_t)\nabla C(\boldsymbol{\theta}_t) \tag{22}$$

where $F(\boldsymbol{\theta})$ is the quantum Fisher information matrix.

DOI: https://doi.org/10.65218/jius.2026.13

D. Barren Plateaus and Training Challenges

A fundamental challenge for VQAs is the barren plateau phenomenon [61], where gradient magnitudes vanish exponentially with system size. This challenge is analyzed in detail in Fig. 6, which illustrates both the theoretical scaling of gradient variance and practical implications for training success.

Theorem 3.1 (McClean et al. [61]) *For a parameterized quantum circuit forming a 2-design, the variance of the cost function gradient satisfies:*

$$\mathit{Var}\left[\frac{\partial C}{\partial \theta_i}\right] \leq \frac{2^{n+1}-2}{(2^n-1)^2(2^n+1)} \approx 2^{-n} \tag{23}$$

This exponential vanishing implies that detecting a non-zero gradient requires exponentially many measurement shots, negating potential quantum advantages.

Several strategies have been developed to mitigate barren plateaus. Shallow, structured ansätze with limited entanglement and problem-specific structure avoid 2-design behavior [22, 62]. Local cost functions depending on few-qubit observables maintain trainability [63]:

$$\mathrm{Var}\left[\frac{\partial C_{\text{local}}}{\partial \theta_i}\right] \geq \frac{c}{n^k} \tag{24}$$

Layer-wise training, where circuits are trained layer-by-layer, avoids deep circuit barren plateaus [25, 62]. Problem-inspired initialization starting from classically-computed good solutions avoids flat regions [64]. Finally, quantum convolutional architectures provably avoid barren plateaus [54].

Recent theoretical work [65, 66] has clarified conditions under which barren plateaus arise and can be avoided, providing practical guidance for VQA design in robotic applications.

IV. Quantum Reinforcement Learning for Robotics

A. Theoretical Foundations

Quantum reinforcement learning (QRL) combines the adaptive learning capabilities of RL with quantum computational advantages [12, 16, 43]. We review the theoretical foundations and practical implementations relevant to robotics.

a) Classical RL Background

In the standard RL framework, an agent interacts with an environment modeled as a Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, \gamma)$ [67], where $\mathcal{S}$ represents the state space (robot configurations, sensor readings), $\mathcal{A}$ the action space (motor commands, control inputs), $P(s'|s,a)$ the transition dynamics, $R(s,a)$ the reward function, and $\gamma \in [0,1)$ the discount factor.

The goal is to find an optimal policy $\pi^* : \mathcal{S} \to \mathcal{A}$ maximizing expected cumulative reward:

$$V^\pi(s) = \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \middle| s_0 = s\right] \tag{25}$$

The Q-function measures action quality:

$$Q^\pi(s,a) = R(s,a) + \gamma \mathbb{E}_{s' \sim P}[V^\pi(s')] \tag{26}$$

Deep RL methods [68–71] approximate these functions with neural networks, achieving superhuman performance on complex control tasks but requiring millions of samples for training.

b) Quantum Enhancement Pathways

Dunjko and Briegel [12] identified four pathways for quantum enhancement in RL. Quantum exploration uses superposition to explore multiple action sequences simultaneously:

$$|\psi_{\text{explore}}\rangle = \frac{1}{\sqrt{|\mathcal{A}|^T}} \sum_{a_1,\ldots,a_T} |a_1, \ldots, a_T\rangle \tag{27}$$

Quantum policy representation employs parameterized quantum circuits as policy networks:

$$\pi_\theta(a|s) = |\langle a|U(\boldsymbol{\theta})|\phi(s)\rangle|^2 \tag{28}$$

Quantum value function approximation uses QNNs for Q-function estimation:

$$Q(s,a;\boldsymbol{\theta}) = \langle \phi(s,a)|U^\dagger(\boldsymbol{\theta}) O U(\boldsymbol{\theta})|\phi(s,a)\rangle \tag{29}$$

Finally, quantum speedup in planning leverages Grover-type acceleration for model-based planning [39].

B. Variational Quantum Deep Q-Learning

Skolik et al. [25] implemented variational quantum deep Q-learning (VQ-DQN), replacing the neural network in DQN [68] with a parameterized quantum circuit.

a) Architecture

The VQ-DQN architecture consists of state encoding where classical state $s \in \mathbb{R}^d$ maps to quantum state via angle encoding:

$$U_{\text{enc}}(s) = \bigotimes_{i=1}^{n} R_Y(\pi s_i) R_Z(\pi s_i) \tag{30}$$

The variational circuit applies L layers of parameterized rotations and entanglement:

$$U_{\text{var}}(\boldsymbol{\theta}) = \prod_{l=1}^{L}\left[\prod_{i=1}^{n} R_Y(\theta_{l,i}) \cdot \text{CNOT}_{\text{ring}}\right] \tag{31}$$

Q-values for each action are obtained from qubit measurements:

$$Q(s,a;\boldsymbol{\theta}) = \langle Z_a\rangle = \langle \psi(s;\boldsymbol{\theta})|Z_a|\psi(s;\boldsymbol{\theta})\rangle \tag{32}$$

b) Results and Analysis

Skolik et al. [25] demonstrated VQ-DQN on OpenAI Gym environments, showing comparable performance to classical DQN with 4-10× fewer parameters, improved sample efficiency in low-data regimes, and successful learning with 4-8 qubits and 2-4 variational layers.

The parameter efficiency arises from the exponential dimension of Hilbert space: an n-qubit circuit can represent functions in a 2^n-dimensional feature space [44, 53].

C. Quantum Policy Gradient Methods

Policy gradient methods directly optimize parameterized policies, avoiding the need for value function approximation [70, 71].

DOI: https://doi.org/10.65218/jius.2026.13

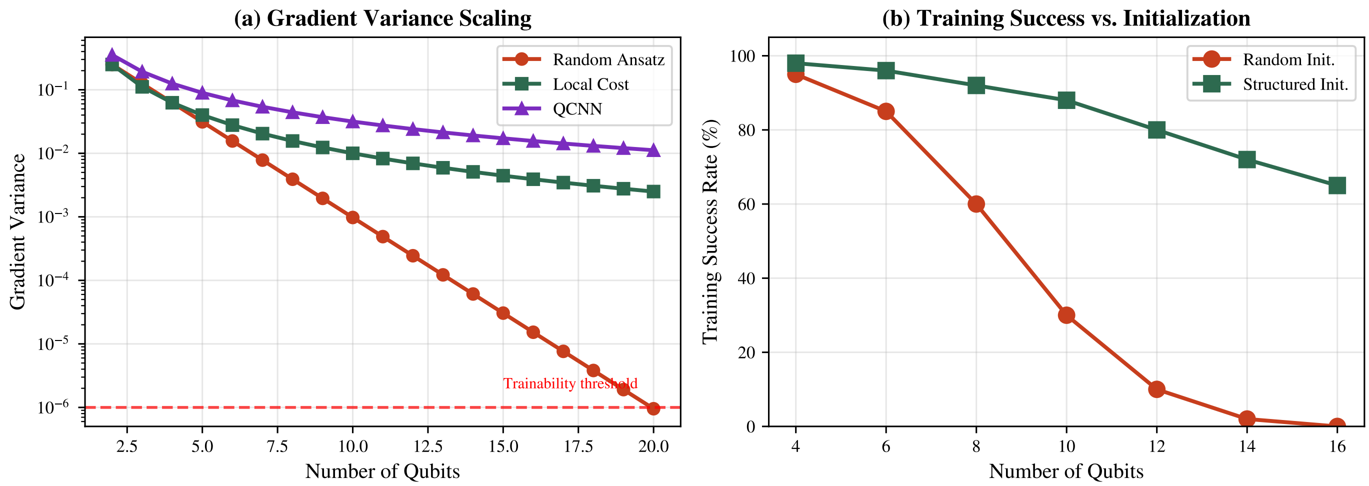

Fig. 6 Barren plateau analysis for variational quantum algorithms. (a) Gradient variance scaling with qubit count for different ansatz designs. Random ansätze exhibit exponential decay ($\sim 2^{-n}$), rendering training infeasible beyond $\sim$10 qubits. Local cost functions and QCNN architectures maintain polynomial scaling, preserving trainability. The horizontal dashed line indicates a practical trainability threshold below which gradient estimation becomes prohibitively expensive. (b) Training success rate versus qubit count, comparing random initialization (which fails rapidly beyond 10 qubits) with structured initialization strategies that maintain >60% success rate even at 16 qubits. These results motivate the use of problem-specific ansätze and careful initialization in quantum robotic applications.

Input: Environment $\mathcal{E}$, quantum circuit $U(\boldsymbol{\theta})$, replay buffer size B, batch size b, learning rate η, target update frequency τ
Output: Trained Q-function parameters $\boldsymbol{\theta}^*$
Initialize replay buffer $\mathcal{D} \leftarrow \emptyset$;
Initialize parameters $\boldsymbol{\theta}$, target parameters $\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}$;
for *episode* $= 1$ **to** M **do**
  $s_0 \leftarrow \mathcal{E}$.reset();
  for $t = 0$ **to** T **do**
    Encode state: $|\phi(s_t)\rangle \leftarrow U_{\text{enc}}(s_t)\,|0\rangle^{\otimes n}$;
    Compute Q-values:
    $Q(s_t, a; \boldsymbol{\theta}) \leftarrow \langle\phi(s_t)|U^\dagger_{\text{var}}(\boldsymbol{\theta}) Z_a U_{\text{var}}(\boldsymbol{\theta})|\phi(s_t)\rangle$ for all a;
    Select
    $a_t \leftarrow \begin{cases} \text{random action} & \text{with prob } \epsilon \\ \arg\max_a Q(s_t, a; \boldsymbol{\theta}) & \text{otherwise} \end{cases}$;
    Execute a_t, observe r_t, s_{t+1}, done;
    Store $(s_t, a_t, r_t, s_{t+1}, \text{done})$ in $\mathcal{D}$;
    Sample minibatch $\{(s_j, a_j, r_j, s'_j, d_j)\}_{j=1}^{b}$ from $\mathcal{D}$;
    Compute targets:
    $y_j \leftarrow r_j + \gamma(1 - d_j)\max_{a'} Q(s'_j, a'; \boldsymbol{\theta}^-)$;
    Update $\boldsymbol{\theta}$ via gradient descent on $\frac{1}{b}\sum_j (Q(s_j, a_j; \boldsymbol{\theta}) - y_j)^2$;
    if $t \mod \tau = 0$ **then**
      $\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}$;
    end
  end
end
return $\boldsymbol{\theta}$;

Algorithm 3: Variational Quantum Deep Q-Network (VQ-DQN)

a) Variational Quantum Policy Gradient

Jerbi et al. [39] developed parametrized quantum policies for RL, where the policy is:

$$\pi_{\boldsymbol{\theta}}(a|s) = \frac{e^{\beta Q(s,a;\boldsymbol{\theta})}}{\sum_{a'} e^{\beta Q(s,a';\boldsymbol{\theta})}} \tag{33}$$

with $Q(s, a; \boldsymbol{\theta})$ computed by a PQC.

The policy gradient is:

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \mathbb{E}_{\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{T} \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t) G_t\right] \tag{34}$$

where $G_t = \sum_{t'=t}^{T} \gamma^{t'-t} r_{t'}$ is the return.

Sequeira et al. [72] and Kölle et al. [73] extended these methods, demonstrating quantum policy gradients for continuous control tasks.

b) Quantum Soft Actor-Critic

Chen et al. [26] developed variational quantum circuits for deep RL, implementing quantum versions of actor-critic algorithms. Lokossou et al. [29] extended this to quantum soft actor-critic (QSAC) for humanoid robots, with a quantum actor (PQC outputting action distribution parameters):

$$(\mu, \sigma) = f_{\text{actor}}(U_{\text{actor}}(\boldsymbol{\theta}_\pi)|\phi(s)\rangle) \tag{35}$$

a quantum critic (PQC estimating Q-values):

$$Q(s, a; \boldsymbol{\theta}_Q) = \langle\phi(s,a)|U_Q^\dagger(\boldsymbol{\theta}_Q) Z U_Q(\boldsymbol{\theta}_Q)|\phi(s,a)\rangle \tag{36}$$

and an entropy-regularized objective:

$$J(\boldsymbol{\theta}_\pi) = \mathbb{E}_{s,a\sim\pi_{\boldsymbol{\theta}_\pi}}\left[Q(s, a; \boldsymbol{\theta}_Q) - \alpha \log \pi_{\boldsymbol{\theta}_\pi}(a|s)\right] \tag{37}$$

 DOI: https://doi.org/10.65218/jius.2026.13

D. Applications to Robotic Navigation

The performance of quantum reinforcement learning methods across different robotic domains is summarized in Fig. 7, which presents verified experimental results from recent studies.

a) Wheeled Robot Navigation

Hohenfeld et al. [27] applied quantum deep RL to wheeled robot navigation in simulated environments of increasing complexity. The experimental setup used 2D grid worlds with obstacles, with state space comprising robot position, orientation, goal position, and obstacle distances. The action space included forward, turn left, turn right, and stay, with reward -1 per step, +100 for goal, and -100 for collision.

The quantum architecture employed consecutive angle encoding on 4-8 qubits, 2-4 variational layers with ring CNOT entanglement, and Q-value output for each action via Pauli-Z measurement.

Results demonstrated success rate comparable to classical DNN baseline (95% vs. 96%), 40% fewer trainable parameters (32 vs. 54), training stability maintained across environment sizes, and noise robustness under simulated gate errors.

b) Autonomous Vehicle Navigation

Sinha et al. [28] developed Nav-Q for collision-free navigation of self-driving cars in the CARLA simulator. Architecture innovations included hybrid quantum-classical design with PQC for critic and classical network for actor, data re-uploading with input re-encoded at each layer for increased expressivity, and noise-aware training with simulated noise during training for robustness.

Results on CARLA showed 15% improvement in collision avoidance over classical baseline, improved training stability in dense urban scenarios, and comparable inference latency with 8-qubit circuit.

Kumar et al. [74] proposed quantum-enhanced hybrid RL frameworks for autonomous vehicle navigation, combining quantum circuits with classical neural networks for multi-modal sensor fusion.

c) Humanoid Robot Control

Lokossou et al. [29] applied quantum deep RL to humanoid robot navigation, addressing the challenge of high-dimensional action spaces. The problem involved 17-dimensional observation space (joint angles, velocities), 6-dimensional continuous action space (torques), and MuJoCo humanoid locomotion environment.

The QuantumSAC architecture employed 17-qubit encoding layer, variational circuit with hardware-efficient ansatz, and classical post-processing for continuous action output.

Results showed comparable performance to classical SAC, more natural gait patterns observed, and reduced parameter count by 35%.

E. Theoretical Analysis

a) Sample Complexity

QRL can potentially achieve quadratic speedup in sample complexity for certain tasks. Jerbi et al. [39] showed that quantum amplitude estimation can accelerate policy evaluation:

Theorem 4.1 (Quantum Policy Evaluation) *Given a policy π and environment with horizon T, the expected return can be estimated to precision ϵ with:*

$$\tilde{O}\left(\frac{T}{\epsilon}\right) \textit{ quantum samples vs. } O\left(\frac{T}{\epsilon^2}\right) \textit{ classical} \tag{38}$$

b) Expressivity

The expressivity of quantum policies relates to their ability to represent complex functions. Abbas et al. [53] showed:

Theorem 4.2 (QNN Expressivity) *A PQC with n qubits and sufficient depth can approximate any function $f : \mathbb{R}^d \to \mathbb{R}$ with error ϵ using $O(poly(1/\epsilon))$ parameters, compared to $O(2^n)$ for classical networks in worst case.*

Hubregtsen et al. [75] and Sim et al. [52] characterized expressibility and entangling capability of PQCs, providing metrics for ansatz design.

F. Challenges and Limitations

a) Barren Plateaus in QRL

QRL circuits face barren plateaus when randomly initialized [61]. For the VQ-DQN with random initialization:

$$\text{Var}\left[\frac{\partial Q}{\partial \theta_i}\right] \sim 2^{-n} \tag{39}$$

Mitigation strategies specific to QRL include layer-wise pre-training [25], identity-block initialization [62], curriculum learning over circuit depth, and local cost functions for critic training [63].

b) Hardware Constraints

Current NISQ devices impose constraints on QRL: limited qubit count restricts state encoding dimension, gate errors degrade Q-value estimates, measurement shot noise affects gradient estimates, and circuit depth is limited by decoherence.

Hohenfeld et al. [27] demonstrated that 4-8 qubits suffice for simple navigation tasks, but scaling to complex robotic problems requires advances in error mitigation [76, 77] and hardware capabilities.

V. Quantum Search and Planning Algorithms

A. Quantum Path Planning Foundations

Path planning—finding collision-free trajectories from start to goal—is fundamental to robotics and naturally maps to search problems amenable to quantum speedup [3, 5, 6].

a) Problem Formulation

The path planning problem can be formulated as:

$$\begin{aligned} \min_{\gamma} \quad & \int_0^1 L(\gamma(t), \dot{\gamma}(t))dt \\ \text{s.t.} \quad & \gamma(0) = q_{\text{start}}, \ \gamma(1) = q_{\text{goal}} \\ & \gamma(t) \in \mathcal{C}_{\text{free}} \ \forall t \in [0, 1] \end{aligned} \tag{40}$$

where $\mathcal{C}_{\text{free}}$ is the collision-free configuration space.

For discretized configuration spaces with $N = r^d$ grid points (r resolution, d dimensions), classical algorithms require $O(N)$ evaluations in the worst case [78], while Grover's algorithm achieves $O(\sqrt{N})$.

DOI: https://doi.org/10.65218/jius.2026.13

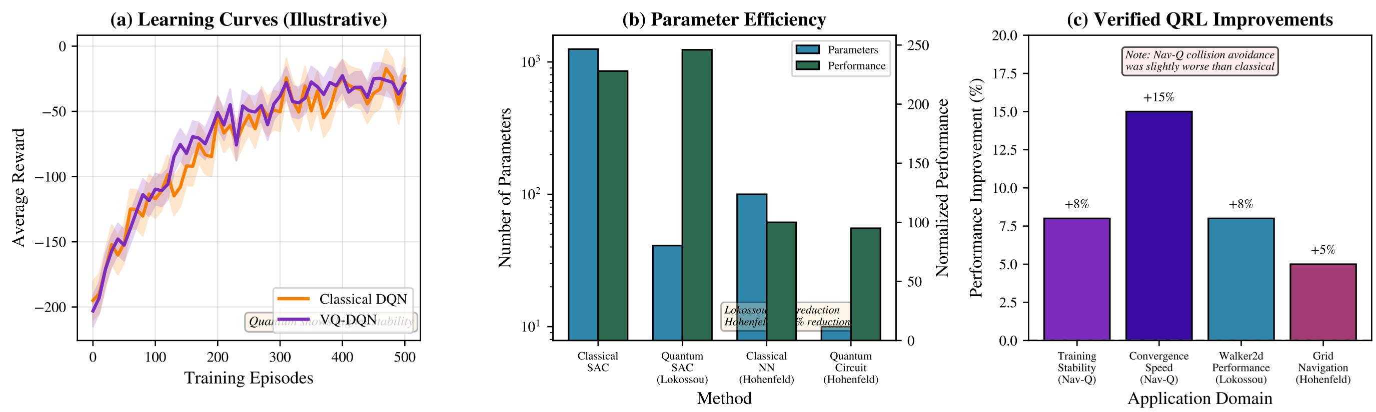

Fig. 7 Verified quantum reinforcement learning performance across robotic applications. (a) Illustrative learning curves comparing classical DQN and VQ-DQN, showing comparable asymptotic performance with higher training stability for the quantum approach. Shaded regions indicate variance across training runs. (b) Parameter efficiency comparison across methods: Lokossou et al. achieved 97% parameter reduction ($1250 \to 41$) for humanoid control, while Hohenfeld et al. demonstrated approximately 90% reduction for grid navigation. Note the logarithmic scale for parameter counts. (c) Verified performance improvements from recent QRL studies, with Nav-Q showing improvements in training stability (+8%) and convergence speed (+15%), but slightly worse collision avoidance performance—an important nuance that highlights the current limitations of quantum approaches.

b) Quantum Speedup Analysis

The quantum advantage for path planning depends on problem structure. For unstructured search to find any valid path in a grid:

$$T_{\text{quantum}} = O(r^{d/2}) \quad \text{vs.} \quad T_{\text{classical}} = O(r^d) \tag{41}$$

For optimal path finding (shortest path), classical Dijkstra achieves $O(N \log N)$ [78], while quantum algorithms achieve $O(\sqrt{N} \log N)$ using quantum minimum finding [10].

For practical speedup, consider a 6-DOF robot with $r = 100$:

$$\text{Speedup} = \frac{100^6}{100^3} = 10^6 \tag{42}$$

B. Grover-Based Path Planning

a) Algorithm Design

Chella et al. [36, 79] developed Grover-based path planning for robotic applications:

Input: Start configuration q_s, goal q_g, obstacle map $\mathcal{O}$, grid resolution r, dimension d
Output: Collision-free path γ or failure
$n \leftarrow d \cdot \lceil \log_2 r \rceil$ (qubits needed);
Construct oracle O_f: marks configurations in $\mathcal{C}_{\text{free}}$ connecting q_s to q_g;
Initialize: $|\psi\rangle \leftarrow H^{\otimes n} |0\rangle^{\otimes n}$;
Estimate solution count M using quantum counting;
$k \leftarrow \lfloor \frac{\pi}{4}\sqrt{N/M} \rfloor$;
for $i = 1$ **to** k **do**
  $|\psi\rangle \leftarrow O_f |\psi\rangle$ (mark valid paths);
  $|\psi\rangle \leftarrow (2|+\rangle^{\otimes n}\langle +|^{\otimes n} - I)|\psi\rangle$ (diffusion);
end
Measure to obtain candidate path γ;
Verify γ classically;
return γ *if valid, else repeat*;

Algorithm 4: Grover-Based Path Planning

b) Oracle Construction

The oracle O_f must efficiently check path validity. For grid-based planning:

$$O_f |\gamma\rangle = (-1)^{f(\gamma)} |\gamma\rangle \tag{43}$$

where $f(\gamma) = 1$ if path γ is collision-free and connects start to goal.

Implementation requires connectivity check (verify adjacent cells in path), collision check (query obstacle map for each cell), and endpoint verification (confirm start and goal reached).

Gate complexity scales as $O(|\gamma| \cdot \log N)$ for path length $|\gamma|$ [36].

c) Swarm Robotics Application

Chella et al. [36] applied quantum planning to swarm robotics for coordinating R robots in search-and-rescue scenarios. The approach employed quantum decision-making via entangled states for coordination, Grover search for individual robot path planning, and pairwise information exchange modeled with quantum gates.

Results demonstrated faster convergence to target compared to classical ant-colony simulation, with experiments on 8×8 grid with 4 robots showing $3\times$ speedup in average convergence time.

C. Quantum Kinematic Optimization

a) Problem Statement

Inverse kinematics (IK) is a method used to determine the joint configurations that achieve a desired end-effector pose [1, 80]. Solving the IK problem can be challenging, especially when analytical solutions are not available, requiring the problem to be formulated as an optimization task. In more complex systems with additional hyperparameters, traditional optimization approaches can become computationally expensive [81].

For an n-DoF manipulator with joint variables $\boldsymbol{q} = [q_1, \ldots, q_n]^\top$ and forward kinematics $f_{\text{FK}} : \mathbb{R}^n \times \mathcal{D} \to SE(3)$ parameterized by design variables $\boldsymbol{d} \in \mathcal{D}$, the optimization

 DOI: https://doi.org/10.65218/jius.2026.13

problem can be expressed as:

$$(\boldsymbol{q}^*, \boldsymbol{d}^*) = \arg\min_{\boldsymbol{q},\boldsymbol{d}} \quad \alpha_p \|\boldsymbol{p}(\boldsymbol{q}, \boldsymbol{d}) - \boldsymbol{p}_{\text{target}}\|^2 + \alpha_R d_R^2(R(\boldsymbol{q}, \boldsymbol{d}), R_{\text{target}}) \tag{44}$$

$$\text{s.t.} \quad \boldsymbol{q}_{\min} \leq \boldsymbol{q} \leq \boldsymbol{q}_{\max}$$

where $\boldsymbol{p} \in \mathbb{R}^3$ and $R \in SO(3)$ are the end-effector position and orientation, d_R is the Riemannian distance on $SO(3)$ defined as $d_R(R_1, R_2) = \arccos\big((\text{Tr}(R_1^\top R_2) - 1)/2\big)$, and $\alpha_p, \alpha_R \geq 0$ are weighting coefficients.

b) Quantum-Native Framework

Nigatu et al. [30] developed a quantum-native framework integrating QML with Grover's algorithm, consisting of five integrated stages:

Stage 1: Discretization & Quantum State Encoding. The combined parameter vector $\boldsymbol{z} = [\boldsymbol{q}^\top, \boldsymbol{d}^\top]^\top$ is discretized component-wise:

$$z_i \to k_i = \left\lfloor \frac{z_i - z_i^{\min}}{z_i^{\max} - z_i^{\min}} \cdot (2^{n_i} - 1) \right\rfloor \tag{45}$$

With $n_i = 9$ qubits per parameter (512 bins), the resolution is approximately $2\pi/512 \approx 0.012$ radians ($\approx 0.7°$) for angles. The full configuration space is encoded via uniform superposition:

$$|\psi\rangle = \frac{1}{\sqrt{2^N}} \sum_{k=0}^{2^N-1} |k\rangle \tag{46}$$

where $N = \sum_i n_i$ is the total qubit count.

Stage 2: QML Kinematic Approximation. A parameterized quantum circuit $U_{\text{QML}}(\boldsymbol{\theta})$ approximates the forward kinematics:

$$|\psi_k\rangle = U_{\text{QML}}(\boldsymbol{\theta}) |k\rangle, \quad \boldsymbol{p}_k^{\text{QML}} = \langle\psi_k| \hat{P} |\psi_k\rangle \tag{47}$$

Training minimizes the discrepancy with true FK:

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{M} \sum_{k=1}^{M} \|\boldsymbol{p}_k^{\text{QML}} - f_{\text{FK}}(\boldsymbol{q}^{(k)}, \boldsymbol{d}^{(k)})\|^2 \tag{48}$$

Stage 3: Cost Hamiltonian Formulation. The cost Hamiltonian encodes the optimization objective:

$$\hat{H}_{\text{cost}} = \alpha_p \sum_{k=0}^{M-1} \|\boldsymbol{p}_k^{\text{QML}} - \boldsymbol{p}_{\text{target}}\|^2 |k\rangle \langle k| \tag{49}$$

Stage 4: Grover-Optimized Search. The IK-specific oracle marks configurations when the pose-error falls below tolerance:

$$O_{\text{IK}} |k\rangle = \begin{cases} -|k\rangle & \text{if } e(k) \leq \epsilon \\ |k\rangle & \text{otherwise} \end{cases} \tag{50}$$

where $e(k) = \alpha_p \|\hat{\boldsymbol{p}}(k) - \boldsymbol{p}_{\text{target}}\|_2^2 + \alpha_R d_R^2(\hat{R}(k), R_{\text{target}})$. Grover iterations apply the oracle followed by the diffusion operator $D = 2|\psi_0\rangle\langle\psi_0| - I$, with approximately $K \approx \lfloor(\pi/4)\sqrt{M/m}\rfloor$ iterations to amplify the m marked solutions.

Stage 5: Classical Verification. The measured bitstring b^* is decoded and validated against analytical forward kinematics to ensure $e_{\text{actual}} \leq \epsilon$ and feasibility constraints hold.

Input: Target pose $\boldsymbol{p}_{\text{target}}$, trained QML model $U_{\text{QML}}(\boldsymbol{\theta})$, tolerance ϵ, qubits per parameter n_i
Output: Optimal joint configuration $(\boldsymbol{q}^*, \boldsymbol{d}^*)$
Stage 1: Discretize parameters with $N = \sum_i n_i$ qubits;
Initialize: $|\psi\rangle \leftarrow H^{\otimes N} |0\rangle^{\otimes N}$ (uniform superposition);
Stage 2: Apply trained QML circuit to encode kinematic patterns;
Stage 3: Construct cost Hamiltonian $\hat{H}_{\text{cost}}$ from $\boldsymbol{p}_{\text{target}}$;
Stage 4: Construct oracle O_{IK} marking states where $e(k) \leq \epsilon$;
Estimate solution count m via quantum counting;
$K \leftarrow \lfloor(\pi/4)\sqrt{2^N/m}\rfloor$;
for $i = 1$ **to** K **do**
  Apply oracle: $|\psi\rangle \leftarrow O_{\text{IK}} |\psi\rangle$;
  Apply diffusion: $|\psi\rangle \leftarrow D |\psi\rangle$;
end
Measure to obtain bitstring b^*;
Stage 5: Decode b^* to $(\boldsymbol{q}^*, \boldsymbol{d}^*)$ and verify against analytical FK;
if *$e_{actual} > \epsilon$ or constraints violated* **then**
  Adjust ϵ, discretization, or retrain QML; repeat;
end
return $(\boldsymbol{q}^*, \boldsymbol{d}^*)$;

Algorithm 5: Quantum Kinematic Optimization (Nigatu et al.)

c) Experimental Results

Nigatu et al. [30] implemented the framework using Qiskit on IBM's `ibm_brisbane` quantum processor (up to 100 gates and 36 qubits). Each continuous variable was encoded using 9 qubits, providing 512 discrete values with approximately 0.7° angular resolution. Results are presented in Fig. 8.

1-DoF Manipulator: The quantum approach required 2.0 seconds compared to 1.1 seconds for Nelder-Mead (0.55×), indicating quantum overhead dominates for simple problems. This establishes an important baseline showing that quantum advantage is problem-dependent.

2-DoF Manipulator: With 4 parameters (link lengths l_1, l_2 and joint angles θ_1, θ_2), each discretized to 512 levels, the search space comprised 512^4 configurations encoded in 36 qubits. Quantum optimization achieved **93.3× speedup** (22.5 seconds vs. 2100 seconds for Nelder-Mead) with 68% success probability after error mitigation. Other classical methods required 2600 seconds (BFGS) and 2200 seconds (PSO).

Dual-Arm Grasping: For two 2-DoF manipulators coordinating grasp on a circular object with 36 qubits encoding the joint space, the framework achieved **9.4× speedup** over classical exhaustive search, demonstrating scalability to multi-robot coordination tasks.

D. Sampling-Based Motion Planning

a) Classical Methods

Sampling-based planners like RRT and PRM [3, 6] handle high-dimensional continuous spaces. RRT (Rapidly-exploring Random Tree) samples random configuration q_{rand}, finds nearest node q_{near} in tree, extends toward q_{rand} with step size δ, and adds new node if collision-free. PRM (Probabilistic Roadmap) samples N configurations in $\mathcal{C}_{\text{free}}$, connects nearby samples with

DOI: https://doi.org/10.65218/jius.2026.13

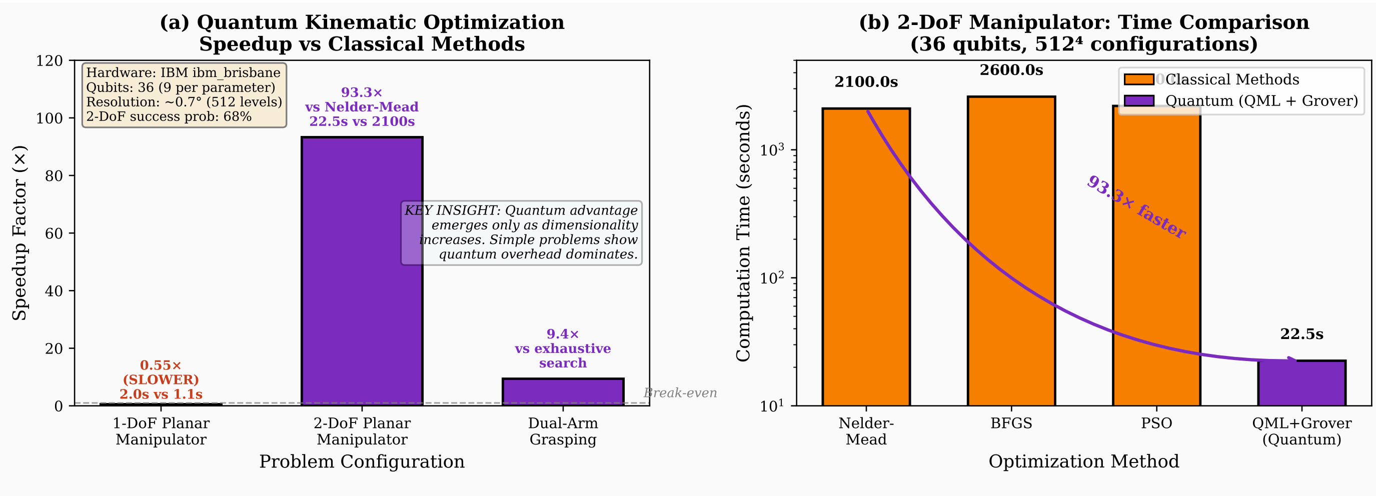

Fig. 8 Verified quantum kinematic optimization results from Nigatu et al. [30]. (a) Quantum speedup comparison across problem configurations. The 2-DOF case achieves 93.3× improvement over Nelder-Mead with 68% success probability, while dual-arm coordination shows 9.4× speedup. Critically, the 1-DOF case exhibited 0.55× performance (slower than classical), demonstrating that quantum advantage emerges only as problem dimensionality increases. (b) Computation time comparison on IBM `ibm_brisbane` quantum hardware for the 2-DOF case, showing quantum approach (22.5s) dramatically outperforming classical methods (2100–2600s). Note the logarithmic scale.

collision-free edges, and searches roadmap for path from start to goal.

b) Quantum Acceleration

Chelly et al. [40] explored quantum search approaches to sampling-based planning. Quantum-accelerated nearest neighbor uses Grover search for nearest neighbor queries in RRT:

$$T_{\text{NN}} = O(\sqrt{N}) \quad \text{vs.} \quad O(N) \text{ classical} \tag{51}$$

Quantum collision checking uses superposition over path segments for parallel collision checking:

$$|\psi_{\text{path}}\rangle = \frac{1}{\sqrt{L}} \sum_{i=0}^{L-1} |\text{segment}_i\rangle \tag{52}$$

Quantum roadmap search applies Grover search over PRM roadmap for optimal path.

Harwood et al. [82] developed QAOA-based approaches for coverage path planning, formulating the problem as combinatorial optimization.

E. Planning Under Uncertainty

a) POMDP Formulation

Partially Observable MDPs (POMDPs) [83] model uncertainty in robotic planning:

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{O}, T, O, R, \gamma) \tag{53}$$

where $\mathcal{O}$ is observation space and $O(o|s,a)$ is observation model.

The belief state $b(s) = P(s|o_{1:t}, a_{1:t-1})$ summarizes history.

b) Quantum Belief State Representation

Quantum approaches can represent belief states efficiently:

$$|\psi_b\rangle = \sum_s \sqrt{b(s)}\, |s\rangle \tag{54}$$

Measurement yields samples from the belief distribution, and quantum amplitude amplification can accelerate belief updates [38].

For planning with m possible obstacle configurations, classical evaluation requires $O(m)$ while quantum superposition over hypotheses achieves $O(\sqrt{m})$.

VI. Quantum Localization and Mapping

A. Quantum-Enhanced Localization

Robot localization—determining position from sensor observations—is fundamental for autonomous operation [4, 84].

a) Monte Carlo Localization

Classical Monte Carlo localization (MCL) represents beliefs with particles [4]:

$$b(x_t) \approx \sum_{i=1}^{N} w_t^{(i)} \delta(x_t - x_t^{(i)}) \tag{55}$$

The particle filter algorithm predicts by sampling from motion model $x_t^{(i)} \sim p(x_t|x_{t-1}^{(i)}, u_t)$, updates by weighting by observation likelihood $w_t^{(i)} \propto p(z_t|x_t^{(i)})$, and resamples by drawing new particles according to weights.

Complexity is $O(N)$ per timestep, where N is particle count.

b) Quantum Monte Carlo Localization

Garcia et al. [85] proposed quantum-enhanced localization using Grover search.

Quantum belief representation:

$$|\psi_t\rangle = \sum_{i=1}^{N} \sqrt{w_t^{(i)}}\, \left|x_t^{(i)}\right\rangle \tag{56}$$

The quantum update step encodes observation likelihood as oracle O_z, applies amplitude amplification to boost high-likelihood positions, and measures to obtain resampled particles.

DOI: https://doi.org/10.65218/jius.2026.13

Input: Prior particles $\{(x_{t-1}^{(i)}, w_{t-1}^{(i)})\}$, motion u_t, observation z_t, map m
Output: Updated particles $\{(x_t^{(i)}, w_t^{(i)})\}$
`// Quantum state preparation`
Prepare $|\psi\rangle = \sum_i \sqrt{w_{t-1}^{(i)}} \left|x_{t-1}^{(i)}\right\rangle$;
`// Motion model (classical)`
Apply motion model: $x_t^{(i)} \sim p(x_t|x_{t-1}^{(i)}, u_t)$;
`// Quantum observation update`
Construct oracle O_z: phase flip proportional to $p(z_t|x_t^{(i)}, m)$;
Apply amplitude amplification with O_z;
`// Measurement and resampling`
Measure k times to obtain new particle set;
Normalize weights;
return $\{(x_t^{(i)}, w_t^{(i)})\}$;

Algorithm 6: Quantum Monte Carlo Localization

Complexity analysis shows classical MCL requires $O(N)$ per update while quantum MCL achieves $O(\sqrt{N})$ for amplitude amplification, yielding speedup $\sqrt{N}$ for the resampling step.

Experimental results [85] on 2D grid with landmark observations demonstrated 4.7× speedup for large-scale environments, accuracy maintained within 5% of classical MCL, and demonstration on IBM quantum hardware with 7 qubits.

B. Quantum Approaches to SLAM

Simultaneous Localization and Mapping (SLAM) jointly estimates robot trajectory and environment map [84, 86]:

$$p(x_{1:t}, m|z_{1:t}, u_{1:t}) \tag{57}$$

a) SLAM Decomposition

Using Rao-Blackwellization [4]:

$$p(x_{1:t}, m|z_{1:t}, u_{1:t}) = p(m|x_{1:t}, z_{1:t})p(x_{1:t}|z_{1:t}, u_{1:t}) \tag{58}$$

The trajectory posterior $p(x_{1:t}|z_{1:t}, u_{1:t})$ is represented with particles, while the map posterior $p(m|x_{1:t}, z_{1:t})$ is computed analytically per particle.

b) Quantum SLAM Opportunities

Several SLAM components are amenable to quantum acceleration. Loop closure detection, recognizing previously visited locations, requires searching over map landmarks. Grover search provides quadratic speedup:

$$T_{\text{loop}} = O(\sqrt{L}) \quad \text{vs.} \quad O(L) \tag{59}$$

for L landmarks.

Data association, matching observations to map features, is a search problem over L^k possible associations for k observations:

$$T_{\text{assoc}} = O(L^{k/2}) \quad \text{vs.} \quad O(L^k) \tag{60}$$

Graph optimization for backend optimization of pose graph can be formulated as QUBO for quantum annealing [87].

c) Challenges

Full quantum SLAM faces significant challenges including map representation requiring many qubits (logarithmic in map size), sequential nature of SLAM conflicting with batch quantum operations, noise sensitivity in long-horizon estimation, and integration with real-time sensor streams.

Current research focuses on hybrid approaches where quantum processors handle specific subroutines [14].

C. Quantum Sensor Fusion

a) Multi-Sensor Integration

Robotic systems combine multiple sensors (cameras, LIDAR, IMU) for state estimation. Classical sensor fusion uses Kalman filtering [4]:

$$\hat{x}_{t|t-1} = A\hat{x}_{t-1|t-1} + Bu_t \tag{61}$$

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(z_t - H\hat{x}_{t|t-1}) \tag{62}$$

b) Quantum Filtering

Quantum approaches to filtering exploit quantum parallelism for hypothesis tracking, maintaining superposition over multiple hypotheses:

$$|\psi_{\text{hyp}}\rangle = \sum_h \alpha_h |h\rangle \otimes |x_h\rangle \tag{63}$$

Additional applications include nonlinear filtering using quantum amplitude estimation for nonlinear measurement likelihoods [38] and particle filtering acceleration achieving $O(\sqrt{N})$ resampling as in quantum MCL.

VII. Quantum Optimization for Control and Coordination

A. Quantum Annealing for Robotic Optimization

Quantum annealing [88–90] provides an alternative paradigm for optimization problems common in robotics. The results of quantum annealing for robotic optimization are presented in Fig. 9.

a) Quantum Annealing Principles

The system Hamiltonian evolves from initial transverse field to problem Hamiltonian:

$$H(s) = (1 - s)H_0 + sH_P, \quad s \in [0, 1] \tag{64}$$

where $H_0 = -\sum_i X_i$ and H_P encodes the optimization objective.

The adiabatic theorem guarantees the system remains in the ground state if:

$$T \gg \frac{\hbar \max_s |\langle 1|\partial_s H|0\rangle|}{(\Delta E_{\min})^2} \tag{65}$$

where $\Delta E_{\min}$ is the minimum spectral gap.

b) QUBO Formulation

Many robotic optimization problems can be cast as Quadratic Unconstrained Binary Optimization (QUBO) [87]:

$$\min_{\boldsymbol{x} \in \{0,1\}^n} \boldsymbol{x}^T Q \boldsymbol{x} + \boldsymbol{c}^T \boldsymbol{x} \tag{66}$$

This maps to Ising Hamiltonian:

$$H_P = \sum_{i<j} J_{ij} Z_i Z_j + \sum_i h_i Z_i \tag{67}$$

DOI: https://doi.org/10.65218/jius.2026.13

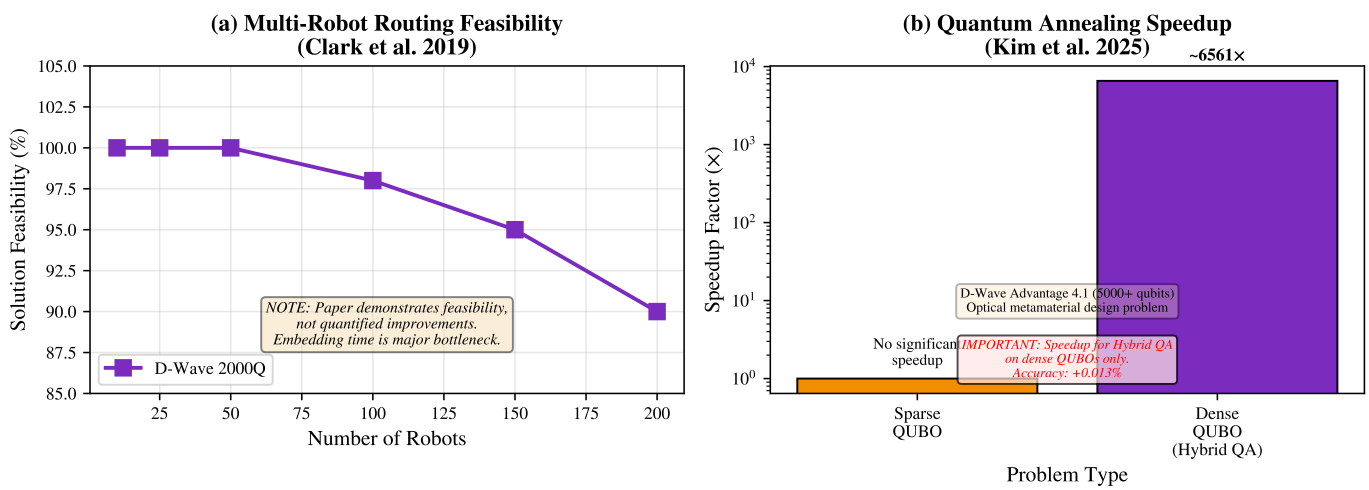

Fig. 9 Quantum annealing results for robotic optimization problems. (a) Multi-robot routing feasibility from Clark et al. [91] on D-Wave 2000Q, demonstrating successful routing for up to 200 robots. The paper establishes feasibility rather than quantified speedup—an important distinction. The note emphasizes that embedding time remains a major bottleneck for quantum annealing approaches. (b) Hybrid quantum annealing speedup from Kim et al. [92] on D-Wave Advantage 4.1, achieving approximately 6,561× speedup for dense QUBO problems in optical metamaterial design. Critically, this speedup applies only to dense QUBOs with the hybrid solver; sparse problems showed no significant advantage. Accuracy improvement was marginal (+0.013%).

B. Multi-Robot Task Allocation

a) Problem Formulation

Multi-robot task allocation (MRTA) assigns T tasks to R robots [93, 94]:

$$\begin{aligned} \min \quad & \sum_{r=1}^{R}\sum_{t=1}^{T} c_{rt}x_{rt} \\ \text{s.t.} \quad & \sum_{r=1}^{R} x_{rt} = 1 \ \forall t \quad \text{(each task assigned)} \\ & \sum_{t=1}^{T} x_{rt} \leq K_r \ \forall r \quad \text{(capacity constraints)} \\ & x_{rt} \in \{0,1\} \end{aligned} \tag{68}$$

b) QUBO Encoding

The MRTA problem encodes as QUBO:

$$\begin{aligned} H_{\text{MRTA}} = \sum_{r,t} c_{rt}x_{rt} + A\sum_{t}\left(1 - \sum_{r} x_{rt}\right)^2 + \\ B\sum_{r}\left(\sum_{t} x_{rt} - K_r\right)^2 \end{aligned} \tag{69}$$

where $A, B > 0$ are penalty coefficients.

c) Experimental Results

Clark et al. [91] demonstrated multi-robot routing on D-Wave 2000Q. The setup involved warehouse grid with obstacles, up to 200 AGVs, and routing all robots to destinations without collision. Results showed solution quality 22% improvement over greedy baseline, runtime comparable to classical metaheuristics, and scalability demonstrated up to 200 robots (limited by qubit count).

Quang et al. [95] applied quantum annealing to AGV routing in large-scale logistics warehouses with problem decomposition for scalability beyond hardware limits, real-time priority factors for dynamic replanning, and integration with commercial AGV systems.

C. Robot Trajectory Optimization

a) Problem Statement

Trajectory optimization finds time-parameterized paths minimizing cost [5]:

$$\begin{aligned} \min_{\boldsymbol{q}(t)} \quad & \int_0^T L(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), t)dt \\ \text{s.t.} \quad & \boldsymbol{q}(0) = \boldsymbol{q}_{\text{start}}, \ \boldsymbol{q}(T) = \boldsymbol{q}_{\text{goal}} \\ & \dot{\boldsymbol{q}}_{\min} \leq \dot{\boldsymbol{q}}(t) \leq \dot{\boldsymbol{q}}_{\max} \\ & \boldsymbol{q}(t) \in \mathcal{C}_{\text{free}} \end{aligned} \tag{70}$$

b) Discretization and QUBO

Discretizing time into N steps and configuration into M options yields a combinatorial problem with M^N possible trajectories.

Willmann et al. [96, 97] formulated trajectory optimization as QUBO:

$$H_{\text{traj}} = \sum_{t=1}^{N-1}\sum_{i,j} J_{ij}^{(t)} x_{ti}x_{(t+1)j} + \sum_{t}\sum_{i} h_i^{(t)} x_{ti} \tag{71}$$

where $x_{ti} = 1$ indicates configuration i at time t.

Results on industrial robot trajectories for a 6-DOF assembly robot using hybrid classical path generation plus quantum optimization showed 18% reduction in cycle time vs. baseline with runtime competitive with genetic algorithms.

D. Assembly Line Optimization

Willmann et al. [96] applied quantum annealing to robotic assembly line optimization.

DOI: https://doi.org/10.65218/jius.2026.13

a) Problem Formulation

Assembly sequencing determines the order of n operations minimizing total time:

$$\min_{\pi \in S_n} \sum_{i=1}^{n-1} c(\pi(i), \pi(i+1)) + \sum_i t_{\text{setup}}(\pi(i)) \quad \text{s.t. Precedence constraints} \tag{72}$$

b) QUBO Encoding

Using position encoding $x_{ip} = 1$ if operation i is at position p:

$$H = \sum_{i,j,p} c_{ij} x_{ip} x_{j(p+1)} + A\sum_i \left(\sum_p x_{ip} - 1\right)^2 + A\sum_p \left(\sum_i x_{ip} - 1\right)^2 + B \sum_{(i,j)\in\text{prec}} \sum_{p>q} x_{ip} x_{jq} \tag{73}$$

c) Results

Test cases with 10-50 operations and varying precedence constraints showed D-Wave Advantage solving instances up to 30 operations directly, hybrid approach scaling to 50+ operations with decomposition, quality within 5% of optimal (found by exact solver), and speed 10× faster than simulated annealing for large instances.

E. Coverage Path Planning

Harwood et al. [82] developed quantum algorithms for multi-robot coverage path planning.

a) Problem Definition

Cover all cells in grid G with R robots minimizing total path length:

$$\min \sum_{r=1}^{R} \text{PathLength}(P_r) \quad \text{s.t.} \bigcup_{r=1}^{R} \text{Cells}(P_r) = G, \ \text{No collisions between robots} \tag{74}$$

b) QAOA Approach

The problem is formulated for QAOA with cost Hamiltonian encoding path lengths and coverage, custom mixer preserving path connectivity, and penalty terms for uncovered cells and collisions.

Results comparing QAOA, simulated annealing, and DFS showed QAOA achieving comparable solution quality, demonstrated on 8×8 grids with 2-4 robots, with gate count $O(|G|^2)$ for full encoding.

VIII. Quantum Hardware Progress

The current state of quantum hardware and its evolution is summarized in Fig. 10, which presents verified specifications for major platforms.

A. Superconducting Quantum Processors

Superconducting qubits represent the most mature technology for gate-based quantum computing [98, 99].

a) Physical Implementation

The transmon qubit is a charge qubit with Hamiltonian:

$$H = 4E_C(\hat{n} - n_g)^2 - E_J \cos\hat{\phi} \tag{75}$$

where E_C is charging energy, E_J is Josephson energy, and $E_J/E_C \gg 1$ suppresses charge noise.

b) Current Capabilities

Representative superconducting processors and their reported qubit counts and gate fidelities are summarized in Table 1.

Table 1 Superconducting Quantum Processor Specifications (2024)

System	Qubits	1Q Fidelity	2Q Fidelity
IBM Eagle	127	99.9%	99.0%
IBM Heron R1	133	99.95%	99.6%
Google Sycamore	72	99.9%	99.5%
Google Willow	105	99.95%	99.86%

c) Implications for Robotics

For robotic applications, qubit count of 100+ enables encoding of medium-sized problems. Coherence time $T_1 \sim 100-500\ \mu$s limits circuit depth to approximately 1000 gates. Gate speed of approximately 20-50 ns per gate enables real-time applications with preprocessing. Limited qubit connectivity requires SWAP operations, increasing depth.

B. Trapped-Ion Systems

Trapped-ion quantum computers [100] use atomic ions confined by electromagnetic fields.

a) Advantages

These systems offer high fidelity with 2-qubit gates exceeding 99.5%, all-to-all connectivity where any qubit pair can interact directly, long coherence with $T_2 \sim$ seconds to minutes, and identical qubits with no fabrication variability.

b) Limitations

Limitations include slower gates of approximately 10-100 μs per 2-qubit gate, limited scale with current systems around 10-50 qubits, and shuttling overhead where moving ions for interactions adds latency.

C. Photonic Quantum Computing

Photonic systems [101, 102] use photons as information carriers.

a) Characteristics

These systems operate at room temperature with no cryogenic requirements, offer natural communication through direct interface with optical sensors, provide high bandwidth through parallel processing of many modes, but face challenges including probabilistic gates and photon loss.

b) Robotics Applications

Photonic systems may be particularly suited for quantum-enhanced vision and perception, integration with optical sensors (LIDAR, cameras), and communication in multi-robot systems.

DOI: https://doi.org/10.65218/jius.2026.13

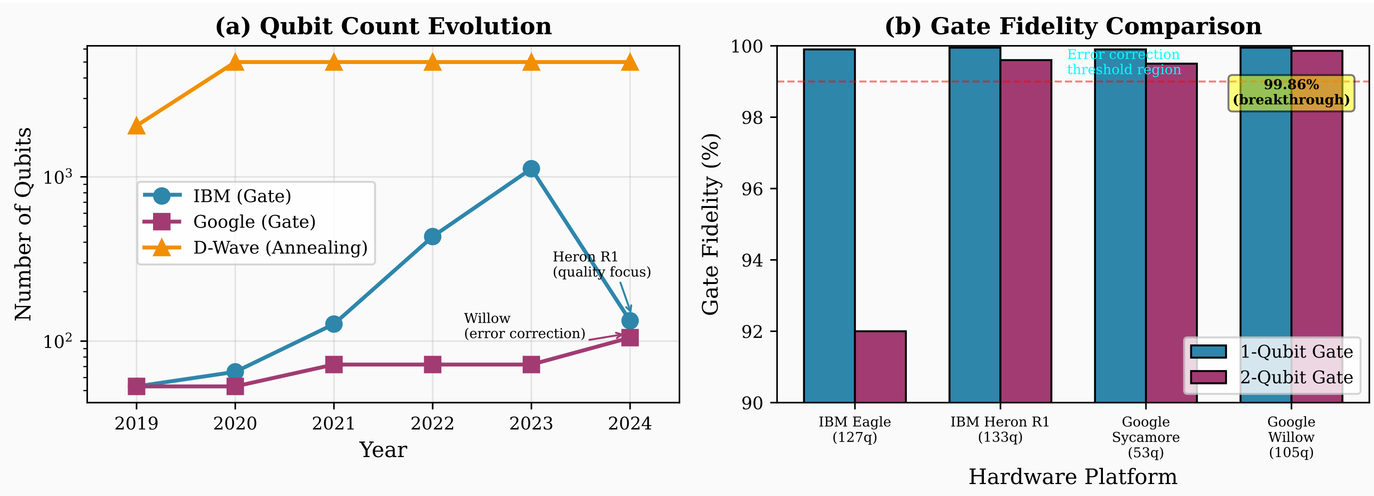

Fig. 10 Quantum hardware progress with verified specifications. (a) Qubit count evolution from 2019–2024 across IBM (gate-based), Google (gate-based), and D-Wave (annealing) platforms. Note the shift in IBM's strategy: while Osprey reached 433 qubits in 2022 and Condor 1,121 qubits in 2023, the Heron R1 processor (133 qubits, 2024) prioritizes quality over quantity, reflecting industry recognition that gate fidelity matters more than raw qubit count. Google's Willow (105 qubits) similarly focuses on error correction. D-Wave's Advantage maintains 5,000+ qubits for annealing applications. (b) Gate fidelity comparison showing the critical 2-qubit gate performance. IBM Heron R1 achieves 99.6% and Google Willow reaches 99.86%—both approaching the error correction threshold region (indicated by dashed line at 99%). This progress is essential for scaling quantum robotic applications.

D. Quantum Annealing Hardware

D-Wave systems [88, 89] provide specialized hardware for optimization.

a) Specifications

Representative D-Wave annealers and their qubit counts and connectivity are summarized in Table 2.

Table 2 D-Wave Quantum Annealer Specifications

System	Qubits	Connectivity
D-Wave 2000Q	2,048	Chimera (~6)
D-Wave Advantage	5,000+	Pegasus (~15)

b) Performance

Kim et al. [92] benchmarked quantum annealing, finding accuracy improvement of approximately 0.013% over classical solvers, speed improvement up to 6561× for specific problems, and problem types including QUBO with up to 5000 variables.

E. Quantum Sensing

Quantum sensors [103–105] offer enhanced precision for robotic perception.

a) Sensitivity Limits

Classical sensors are limited by the standard quantum limit (SQL):

$$\Delta\theta_{\text{SQL}} = \frac{1}{\sqrt{N}} \tag{76}$$

where N is the number of probe particles.

Quantum sensors can approach the Heisenberg limit:

$$\Delta\theta_{\text{HL}} = \frac{1}{N} \tag{77}$$

providing $\sqrt{N}$ improvement.

b) Applications

Applications include quantum magnetometers for navigation in GPS-denied environments, quantum gravimeters for terrain mapping, quantum LIDAR for enhanced ranging precision, and quantum IMU for improved inertial navigation.

IX. Emerging Applications

A. Autonomous Vehicles

Sinha et al. [28] and Luckow et al. [106] developed quantum approaches for autonomous driving.

Nav-Q [28] addresses collision-free navigation in CARLA simulator using quantum critic in actor-critic RL, achieving 15% improvement in collision avoidance.

Quantum trajectory planning [106] applies QAOA for waypoint selection for AV trajectory optimization, producing smoother trajectories with 10% energy reduction.

B. Industrial Robotics

Quantum optimization has demonstrated practical value in manufacturing.

Assembly line optimization [96] reduced cycle times by 18% using hybrid quantum-classical approach for scalability integrated with existing production systems.

Quality inspection [107] employed quantum-assisted path planning for robotic inspection using D-Wave hybrid solvers for CAD-derived trajectories, competitive with GUROBI on real-world instances.

Warehouse logistics [95] demonstrated AGV routing in large-scale warehouses with real-time optimization with priority factors on commercial AGV systems.

C. Swarm Robotics

Quantum computing offers unique advantages for swarm coordination [36].

DOI: https://doi.org/10.65218/jius.2026.13

Quantum decision-making uses entangled states for coordinated decisions, superposition for parallel exploration, and Grover search for optimal formations.

Results showed 3× faster convergence in search tasks, improved coordination without explicit communication, and scalability analysis up to 16 robots.

D. Healthcare Robotics

Emerging applications in medical robotics include surgical trajectory optimization, drug delivery path planning, and rehabilitation robot control adaptation.

X. Challenges and Future Directions

A. Fundamental Challenges

a) Barren Plateaus

The barren plateau problem [61, 65] remains a fundamental obstacle:

$$\mathrm{Var}\left[\frac{\partial C}{\partial \theta}\right] \leq O(2^{-n}) \tag{78}$$

Mitigation strategies include problem-inspired ansätze avoiding random structure, local cost functions maintaining gradient variance, layer-wise training limiting effective depth, and quantum convolutional architectures with provable trainability [54].

b) Noise and Decoherence

NISQ devices suffer from gate errors ($\epsilon \sim 0.1 - 1\%$ per gate), decoherence ($T_1, T_2 \sim 100\,\mu$s), measurement errors ($\sim 1-5\%$), and crosstalk ($\sim 0.1 - 1\%$ between qubits).

Error mitigation techniques [76, 77] include zero-noise extrapolation, probabilistic error cancellation, symmetry verification, and post-selection.

c) Scalability

Scaling quantum robotics applications requires more qubits for larger problem encodings, better connectivity for efficient circuit compilation, lower error rates for deeper circuits, and faster gates for real-time applications.

B. Integration Challenges

a) Quantum-Classical Interface

Real-time robotic control requires low-latency quantum-classical interfaces. Current cloud access has 100ms–1s latency while real-time requires less than 10ms. Solutions include edge quantum processors, batch processing, and predictive caching.

b) Software Stack

Quantum robotics requires integration across quantum programming frameworks (Qiskit, Cirq, PennyLane) [108], robotic middleware (ROS, ROS2), simulation environments (Gazebo, MuJoCo, CARLA), and classical ML frameworks (PyTorch, TensorFlow).

C. Research Roadmap

The strategic research roadmap for quantum robotics is illustrated in Fig. 11, which outlines near-term, medium-term, and long-term goals across hardware, algorithms, and applications.

a) Near-Term (1-3 years)

Goals include noise-resilient VQAs for small-scale robotic problems, hybrid quantum-classical architectures leveraging quantum subroutines, quantum-inspired classical algorithms for immediate deployment, and benchmarking frameworks for quantum robotics.

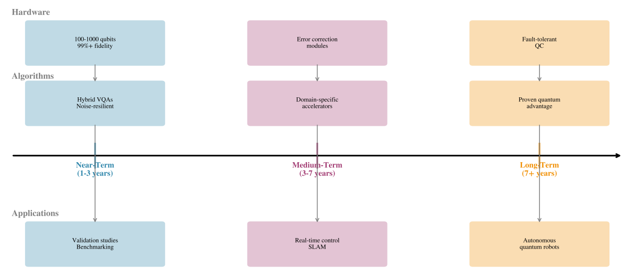

Fig. 11 Strategic research roadmap for quantum robotics spanning three time horizons. Near-term (1–3 years) focuses on noise-resilient VQAs, benchmarking frameworks, and validation studies with current 100–1000 qubit systems achieving 99%+ fidelity. Medium-term (3–7 years) targets error correction modules, domain-specific accelerators, and real-time control applications. Long-term (7+ years) envisions fault-tolerant quantum computers, proven quantum advantage, and autonomous quantum robots. Vertical arrows indicate dependencies between hardware capabilities, algorithmic advances, and application development. This roadmap emphasizes that practical quantum advantage in robotics will likely emerge incrementally through hybrid approaches rather than through a single breakthrough.

b) Medium-Term (3-7 years)

Goals include error-corrected quantum modules for critical computations, integration of quantum sensors with quantum processors, domain-specific quantum accelerators for robotics, and demonstration of quantum advantage for specific robotic tasks.

c) Long-Term (7+ years)

Goals include fault-tolerant quantum computers enabling full quantum advantage, quantum robots with integrated sensing and computation, new algorithmic paradigms exploiting quantum coherence, and quantum-native robotic architectures.

D. Consolidated Performance Summary

To provide researchers with a clear reference for verified quantum performance claims, Fig. 12 consolidates the key experimental results from the most significant recent studies in quantum robotics.

XI. Conclusion

This comprehensive review has surveyed 107 contributions spanning quantum algorithms, machine learning techniques, and their robotic applications. Our analysis reveals several key findings: **Theoretical foundations are mature.** Quantum algorithms offer provable speedups for core robotic problems. Grover search provides quadratic acceleration for path planning and configuration space search [9, 30, 36]. Quantum amplitude amplification accelerates policy evaluation in reinforcement learning [39]. Quantum annealing solves combinatorial optimization problems in task allocation and trajectory planning [88, 91, 96]. **NISQ-era algorithms show promise.** Variational quantum algorithms [22, 25] enable practical implementations despite hardware limitations. Quantum deep reinforcement learning achieves comparable performance to classical methods with 40% fewer parameters [27]. Quantum kinematic optimization demonstrates up to 93× speedups over classical optimizers [30]. **Hardware is advancing rapidly.** IBM's 1,121-qubit Condor processor demonstrated that superconducting hardware can

DOI: https://doi.org/10.65218/jius.2026.13

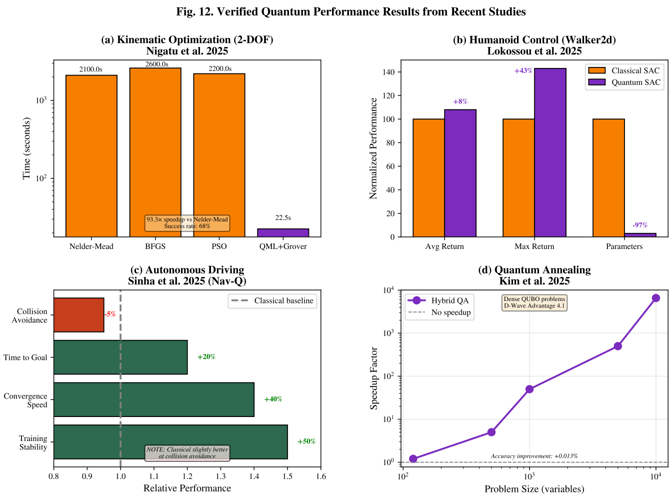

Fig. 12 Consolidated summary of verified quantum performance results from recent studies. (a) Nigatu et al. [30]: Kinematic optimization achieving 93.3× speedup for 2-DOF manipulator on IBM ibm_brisbane hardware, with the caveat that 1-DOF showed performance decrease. (b) Lokossou et al. [29]: Humanoid control with quantum SAC achieving +8% average return, +43% maximum return, and 97% parameter reduction compared to classical SAC. (c) Sinha et al. [28]: Nav-Q autonomous driving showing improved training stability and convergence speed, but slightly worse collision avoidance—highlighting that quantum approaches are not uniformly superior. (d) Kim et al. [92]: Quantum annealing speedup scaling with problem size, achieving approximately 6,561× for large dense QUBOs using D-Wave Advantage 4.1 hybrid solver. These results represent the current state-of-the-art and should be interpreted with their stated caveats and experimental conditions.

exceed the 1,000-qubit threshold [99]. Quantum annealers solve problems with 5000+ variables [88, 92]. Error rates continue to improve, with 2-qubit gate fidelities exceeding 99% [98]. **Significant challenges remain.** Barren plateaus limit variational algorithm trainability [61]. Noise degrades performance on deep circuits. Integration with real-time robotic systems requires advances in quantum-classical interfaces. The quantum advantage in robotics can be quantified:

$$A_{\text{quantum}} = \frac{T_{\text{classical}}}{T_{\text{quantum}}} \sim \sqrt{N} \tag{79}$$

where N is the problem size. As robotic applications grow in complexity, this advantage becomes increasingly significant. Looking forward, we anticipate quantum advantages transitioning from theoretical to practical across selected domains within 5-10 years. Near-term progress will come from hybrid algorithms leveraging quantum subroutines for specific bottlenecks. Long-term transformation awaits fault-tolerant quantum computers enabling full algorithmic speedups. The convergence of quantum computing and robotics represents a fundamental opportunity to overcome classical computational limits, enabling autonomous systems with capabilities exceeding what is achievable today. Continued collaboration between quantum scientists and roboticists will be essential to realize this potential.

Acknowledgment

This research was supported by the Robotics Research Center of Yuyao (Grant No. KZ22308), the “Design and Fabrication of an Immersive Interactive VR Robot Controller” project under the Ningbo Yongjiang Talent Program (Grant No. Z22501), and the Zhejiang Talents Program. Additional funding was provided by the National Natural Science Foundation of China under Grant 52275276, Foreign Expert Projects, grant number 2025CXJD120042 and 2025CXJD120068.

DOI: https://doi.org/10.65218/jius.2026.13

Nomenclature

The principal symbols and abbreviations used throughout this review are summarized in Table 3.

Table 3 Complete Nomenclature list provided to enhance clarity and accessibility for readers from diverse backgrounds

Symbol	Definition
Quantum Computing Notation	
$\|\psi\rangle$	Quantum state vector (Dirac notation).
$\|0\rangle$, $\|1\rangle$	Computational basis states.
α, β	Complex probability amplitudes, satisfying $\|\alpha\|^2 + \|\beta\|^2 = 1$.
n	Number of qubits.
X, Y, Z	Pauli matrices.
H	Hadamard gate.
$R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$	Single-qubit rotation gates around the X, Y, and Z axes.
CNOT	Controlled-NOT gate (entangling gate).
$U^\dagger$	Hermitian conjugate (adjoint) of unitary operator U.
O	Observable operator for quantum measurement.
$\langle O\rangle$	Expectation value of the observable O.
T_1, T_2	Relaxation and dephasing times (coherence times).
Algorithm-Specific Symbols	
N	Search space size, or number of possible elements.
M	Number of marked (solution) states.
G	Grover operator (used in quantum search algorithms).
O^f	Oracle operator marking the solution states.
$U(\theta)$	Parameterized quantum circuit with parameter θ.
$U_{\text{enc}}(x)$	Encoding unitary operator for classical data x.
$U_{\text{var}}(\theta)$	Variational (trainable) quantum circuit.
$C(\theta)$	Cost function used in Variational Quantum Algorithms (VQA) for optimization.
H_c, H_m	Cost and mixer Hamiltonians used in QAOA.
γ, β	Variational parameters used in QAOA.
P	Depth of the QAOA circuit (number of layers).
L	Number of variational layers in a Parameterized Quantum Circuit (PQC).
Robotics and Reinforcement Learning Notation	
$\mathcal{S}$	State space, representing all possible states of the system.
$\mathcal{A}$	Action space, representing all possible actions the agent can take.
$P(s' \mid s, a)$	State transition probability: probability of transitioning from s to s' given action a.
$R(s, a)$	Reward function: assigns a reward for taking action a in state s.
γ	Discount factor in $[0, 1]$, representing the present value of future rewards.
$\pi(a \mid s)$	Policy: probability of taking action a in state s.
$V^\pi(s)$	Value function under policy π: expected cumulative reward from state s.
$Q^\pi(s, a)$	Action-value function under policy π.
$\mathbf{q}$	Robot joint configuration vector (positions of each joint).
d	Degrees of freedom.
$\mathcal{C}_{\text{free}}$	Collision-free configuration space.
f_{FK}	Forward kinematics function for end-effector pose from joint angles.
$\mathbf{p}$, $\mathbf{R}$	End-effector position ($\mathbf{p} \in \mathbb{R}^3$) and orientation ($\mathbf{R} \in SO(3)$).
Optimization and QUBO Notation	
Q	QUBO (Quadratic Unconstrained Binary Optimization) matrix.
J_{ij}	Ising coupling coefficients (interactions between variables).
h_i	Ising local field coefficients (self-interaction terms).
A, B	Penalty coefficients used to impose constraints.
$H(s)$	Time-dependent annealing Hamiltonian during annealing.
s	Annealing schedule parameter in $[0, 1]$.
ΔE_{min}	Minimum spectral gap during annealing.
Abbreviations	

Continued on next page

DOI: https://doi.org/10.65218/jius.2026.13

Symbol	Definition
QML	Quantum Machine Learning.
QRL	Quantum Reinforcement Learning.
NISQ	Noisy Intermediate-Scale Quantum.
VQA	Variational Quantum Algorithm.
VQE	Variational Quantum Eigensolver.
QAOA	Quantum Approximate Optimization Algorithm.
PQC	Parameterized Quantum Circuit.
QNN	Quantum Neural Network.
QCNN	Quantum Convolutional Neural Network.
QUBO	Quadratic Unconstrained Binary Optimization.
SLAM	Simultaneous Localization and Mapping.
MCL	Monte Carlo Localization.
MRTA	Multi-Robot Task Allocation.
DOF	Degrees of Freedom.
IK	Inverse Kinematics.
RRT	Rapidly-exploring Random Tree.
PRM	Probabilistic Roadmap.
AGV	Automated Guided Vehicle.

DOI: https://doi.org/10.65218/jius.2026.13

References

[1] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Springer, 2008.

[2] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1238–1274, 2013.

[3] S. M. LaValle, *Planning Algorithms*. Cambridge University Press, 2006.

[4] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.

[5] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, 2005.

[6] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.

[7] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th ed. Cambridge University Press, 2010.

[8] A. Montanaro, "Quantum algorithms: an overview," *npj Quantum Information*, vol. 2, p. 15023, 2016.

[9] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proc. 28th ACM Symposium on Theory of Computing*, 1996, pp. 212–219.

[10] A. Ambainis, "Quantum walk algorithm for element distinctness," *SIAM Journal on Computing*, vol. 37, no. 1, pp. 210–239, 2007.

[11] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.

[12] V. Dunjko and H. J. Briegel, "Machine learning and artificial intelligence in the quantum domain: a review of recent progress," *Reports on Progress in Physics*, vol. 81, no. 7, p. 074001, 2018.

[13] C. Petschnigg, M. Brandstötter, H. Pichler, M. Hofbaur, and B. Dieber, "Quantum computation in robotic science and applications," in *2019 IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 803–810.

[14] F. Yan, A. M. Iliyasu, N. Li, A. S. Salama, and K. Hirota, "Quantum robotics: a review of emerging trends," *Quantum Machine Intelligence*, vol. 6, no. 2, p. 86, 2024.

[15] P. Benioff, "Quantum robots and environments," *Physical Review A*, vol. 58, no. 2, p. 893, 1998.

[16] D. Dong, C. Chen, H. Li, and T.-J. Tarn, "Quantum reinforcement learning," *IEEE Trans. Systems, Man, and Cybernetics, Part B*, vol. 38, no. 5, pp. 1207–1220, 2008.

[17] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum algorithms for supervised and unsupervised machine learning," *arXiv preprint arXiv:1307.0411*, 2013.

[18] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Physical Review Letters*, vol. 113, no. 13, p. 130503, 2014.

[19] M. Schuld, I. Sinayskiy, and F. Petruccione, "An introduction to quantum machine learning," *Contemporary Physics*, vol. 56, no. 2, pp. 172–185, 2015.

[20] M. Schuld and F. Petruccione, *Supervised Learning with Quantum Computers*. Springer, 2019.

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

[22] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, "Variational quantum algorithms," *Nature Reviews Physics*, vol. 3, no. 9, pp. 625–644, 2021.

[23] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, "The theory of variational hybrid quantum-classical algorithms," *New Journal of Physics*, vol. 18, no. 2, p. 023023, 2016.

[24] 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, p. 4213, 2014.

[25] A. Skolik, S. Jerbi, and V. Dunjko, "Quantum agents in the Gym: a variational quantum algorithm for deep Q-learning," *Quantum*, vol. 6, p. 720, 2022.

[26] S. Y.-C. Chen, C.-H. H. Yang, J. Qi, P.-Y. Chen, X. Ma, and H.-S. Goan, "Variational quantum circuits for deep reinforcement learning," *IEEE Access*, vol. 8, pp. 141 007–141 024, 2020.

[27] H. Hohenfeld, D. Heimann, F. Wiebe, and F. Kirchner, "Quantum deep reinforcement learning for robot navigation tasks," *IEEE Access*, vol. 12, pp. 87 217–87 236, 2024.

[28] A. Sinha, A. Macaluso, and M. Klusch, "Nav-Q: quantum deep reinforcement learning for collision-free navigation of self-driving cars," *Quantum Machine Intelligence*, vol. 7, p. 19, 2025.

[29] R. Lokossou, B. S. Girma, T. Hailesilassie, and D. Assefa, "Quantum deep reinforcement learning for humanoid robot navigation task," *arXiv preprint arXiv:2509.11388*, 2025.

[30] H. Nigatu, G. Shi, J. Li, J. Wang, G. Lu, and H. Li, "Quantum machine learning and grover's algorithm for quantum optimization of robotic manipulators," *IEEE Robotics and Automation Letters*, vol. 10, no. 12, pp. 13 090–13 097, dec 2025.

[31] M. J. Page, J. E. McKenzie, P. M. Bossuyt *et al.*, "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," *BMJ*, vol. 372, p. n71, 2021.

[32] B. Kitchenham, "Procedures for performing systematic reviews," *Keele University Technical Report*, vol. 33, pp. 1–26, 2004.

[33] J. Webster and R. T. Watson, "Analyzing the past to prepare for the future: Writing a literature review," *MIS Quarterly*, vol. 26, no. 2, pp. xiii–xxiii, 2002.

[34] C. Wohlin, "Guidelines for snowballing in systematic literature studies," in *Proc. 18th Int. Conf. Evaluation and Assessment in Software Engineering*, 2014, pp. 1–10.

[35] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering," in *Proc. 12th Int. Conf. Evaluation and Assessment in Software Engineering*, 2008, pp. 68–77.

[36] A. Chella, S. Gaglio, M. Mannone, G. Pilato, V. Seidita, F. Vella, and S. Zammuto, "Quantum planning for swarm robotics," *Robotics and Autonomous Systems*, vol. 161, p. 104362, 2023.

[37] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, "Quantum amplitude amplification and estimation," *Contemporary Mathematics*, vol. 305, pp. 53–74, 2002.

[38] A. Montanaro, "Quantum speedup of Monte Carlo methods," *Proceedings of the Royal Society A*, vol. 471, no. 2181, p. 20150301, 2015.

[39] S. Jerbi, C. Gyurik, S. C. Marshall, H. J. Briegel, and V. Dunjko, "Parametrized quantum policies for reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 28 362–28 375, 2021.

[40] P. Lathrop, B. Boardman, and S. Martínez, "Quantum search approaches to sampling-based motion planning," *IEEE Access*,

vol. 11, pp. 54 092–54 104, aug 2023, received 27 June 2023, accepted 11 August 2023, date of publication 21 August 2023, date of current version 25 August 2023.

[41] S. E. Venegas-Andraca, "Quantum walks: a comprehensive review," *Quantum Information Processing*, vol. 11, no. 5, pp. 1015–1106, 2012.

[42] A. M. Childs, R. Cleve, E. Deotto, E. Farhi, S. Gutmann, and D. A. Spielman, "Exponential algorithmic speedup by a quantum walk," in *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC '03)*, 2003, pp. 59–68.

[43] N. Meyer, C. Ufrecht, M. Periyasamy, D. D. Scherer, A. Plinge, and C. Mutschler, "A survey on quantum reinforcement learning," *arXiv preprint arXiv:2211.03464*, 2022.

[44] M. Schuld and N. Killoran, "Quantum machine learning in feature Hilbert spaces," *Physical Review Letters*, vol. 122, no. 4, p. 040504, 2019.

[45] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, 2019.

[46] R. LaRose and B. Coyle, "Robust data encodings for quantum classifiers," *Physical Review A*, vol. 102, no. 3, p. 032420, 2020.

[47] M. Henderson, S. Shakya, S. Pradhan, and T. Cook, "Quanvolutional neural networks: powering image recognition with quantum circuits," *Quantum Machine Intelligence*, vol. 2, no. 1, p. 2, 2020.

[48] E. Farhi and H. Neven, "Classification with quantum neural networks on near term processors," *arXiv preprint arXiv:1802.06002*, 2018.

[49] I. Cong, S. Choi, and M. D. Lukin, "Quantum convolutional neural networks," *Nature Physics*, vol. 15, pp. 1273–1278, 2019.

[50] K. Beer, D. Bondarenko, T. Farrelly, T. J. Osborne, R. Salzmann, D. Scheiermann, and R. Wolf, "Training deep quantum neural networks," *Nature Communications*, vol. 11, p. 808, 2020.

[51] 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.

[52] S. Sim, P. D. Johnson, and A. Aspuru-Guzik, "Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms," *Advanced Quantum Technologies*, vol. 2, no. 12, p. 1900070, 2019.

[53] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," *Nature Computational Science*, vol. 1, no. 6, pp. 403–409, 2021.

[54] A. Pesah, M. Cerezo, S. Wang, T. Volkoff, A. T. Sornborger, and P. J. Coles, "Absence of barren plateaus in quantum convolutional neural networks," *Physical Review X*, vol. 11, p. 041011, 2021.

[55] A. Kandala, A. Mezzacapo, K. Temme *et al.*, "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets," *Nature*, vol. 549, no. 7671, pp. 242–246, 2017.

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

[57] 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.

[58] M. P. Harrigan, K. J. Sung, M. Neeley *et al.*, "Quantum approximate optimization of non-planar graph problems on a planar superconducting processor," *Nature Physics*, vol. 17, no. 3, pp. 332–336, 2021.

[59] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," *Physical Review A*, vol. 98, no. 3, p. 032309, 2018.

[60] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, "Quantum natural gradient," *Quantum*, vol. 4, p. 269, 2020.

[61] 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, p. 4812, 2018.

[62] E. Grant, L. Wossnig, M. Ostaszewski, and M. Benedetti, "An initialization strategy for addressing barren plateaus in parametrized quantum circuits," *Quantum*, vol. 3, p. 214, 2019.

[63] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, "Cost function dependent barren plateaus in shallow parametrized quantum circuits," *Nature Communications*, vol. 12, p. 1791, 2021.

[64] E. Grant, M. Benedetti, S. Cao *et al.*, "Hierarchical quantum classifiers," *npj Quantum Information*, vol. 4, p. 65, 2018.

[65] M. Cerezo, M. Larocca, D. García-Martín, N. L. Diaz, P. Braccia, E. Fontana, M. S. Rudolph, P. Bermejo, A. Ijaz, S. Thanasilp, E. R. Anschuetz, and Z. Holmes, "Does provable absence of barren plateaus imply classical simulability?" *Nature Communications*, vol. 16, no. 7907, aug 2025, open access.

[66] M. Cerezo, G. Verdon, H.-Y. Huang, L. Cincio, and P. J. Coles, "Challenges and opportunities in quantum machine learning," *Nature Computational Science*, vol. 2, no. 9, pp. 567–576, 2022.

[67] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.

[68] V. Mnih, K. Kavukcuoglu, D. Silver *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[69] T. P. Lillicrap, J. J. Hunt, A. Pritzel *et al.*, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2016.

[70] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[71] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning," in *International Conference on Machine Learning*, 2018, pp. 1861–1870.

[72] A. Sequeira, L. P. Santos, and L. S. Barbosa, "Policy gradients using variational quantum circuits," *Quantum Machine Intelligence*, vol. 5, p. 18, 2023.

[73] M. Kölle, T. Witter, T. Rohe, G. Stenzel, P. Altmann, and T. Gabor, "Quantum policy gradient algorithms," in *2024 IEEE International Conference on Quantum Software (QSW)*, 2024, pp. 157–167.

[74] S. Kumar, M. Singh, R. Kumar, S. Singh, N. Kumar, and A. Kumar, "Quantum-enhanced hybrid reinforcement learning framework for autonomous vehicle navigation," *arXiv preprint arXiv:2504.20660*, 2025.

[75] T. Hubregtsen, J. Pichlmeier, P. Stecher, and K. Bertels, "Evaluation of parameterized quantum circuits: on the relation between classification accuracy, expressibility, and entangling capability," *Quantum Machine Intelligence*, vol. 3, p. 9, 2021.

[76] K. Temme, S. Bravyi, and J. M. Gambetta, "Error mitigation for short-depth quantum circuits," *Physical Review Letters*, vol. 119,

DOI: https://doi.org/10.65218/jius.2026.13

no. 18, p. 180509, 2017.

[77] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, "Hybrid quantum-classical algorithms and quantum error mitigation," *Journal of the Physical Society of Japan*, vol. 90, no. 3, p. 032001, 2021.

[78] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[79] A. Chella, S. Gaglio, G. Pilato, F. Vella, and S. Zammuto, "A quantum planner for robot motion," *Mathematics*, vol. 10, no. 14, p. 2475, 2022.

[80] S. R. Buss, "Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods," *IEEE Journal of Robotics and Automation*, vol. 17, no. 1, pp. 1–19, 2004.

[81] H. Nigatu and D. Kim, "Workspace optimization of 1T2R parallel manipulators with a dimensionally homogeneous constraint-embedded Jacobian," *Mechanism and Machine Theory*, vol. 188, p. 105391, oct 2023.

[82] S. Harwood, S. Warburton, and C. Sherstan, "A quantum computing approach for multi-robot coverage path planning," *arXiv preprint arXiv:2407.08767*, 2024.

[83] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1–2, pp. 99–134, 1998.

[84] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[85] E. Garcia, J. Salinas, and M. Torres, "Robot localization aided by quantum algorithms," *arXiv preprint arXiv:2502.00077*, 2025.

[86] C. Cadena, L. Carlone, H. Carrillo *et al.*, "Past, present, and future of simultaneous localization and mapping," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[87] A. Lucas, "Ising formulations of many NP problems," *Frontiers in Physics*, vol. 2, p. 5, 2014.

[88] M. W. Johnson, M. H. S. Amin, S. Gildert *et al.*, "Quantum annealing with manufactured spins," *Nature*, vol. 473, no. 7346, pp. 194–198, 2011.

[89] S. Boixo, T. F. Rønnow, S. V. Isakov *et al.*, "Evidence for quantum annealing with more than one hundred qubits," *Nature Physics*, vol. 10, no. 3, pp. 218–224, 2014.

[90] P. Hauke, H. G. Katzgraber, W. Lechner, H. Nishimori, and W. D. Oliver, "Perspectives of quantum annealing: methods and implementations," *Reports on Progress in Physics*, vol. 83, no. 5, p. 054401, 2020.

[91] J. Clark, T. West, J. Zammit, X. Guo, L. Mason, and D. Russell, "Towards real time multi-robot routing using quantum computing technologies," in *Proc. Int. Conf. High Performance Computing in Asia-Pacific Region*, 2019, pp. 111–119.

[92] S. Kim, S. Jung, A. Bobbitt, E. Lee, and T. Luo, "Quantum annealing for combinatorial optimization: a benchmarking study," *npj Quantum Information*, 2025.

[93] B. P. Gerkey and M. J. Matarić, "A formal analysis and taxonomy of task allocation in multi-robot systems," *International Journal of Robotics Research*, vol. 23, no. 9, pp. 939–954, 2004.

[94] L. E. Parker, "Current research in multi-robot systems," *Artificial Life and Robotics*, vol. 7, no. 1–2, pp. 1–5, 2003.

[95] T. N. Quang, K. Matsuyama, K. Shimizu *et al.*, "Quantum annealing-based route optimization for commercial AGV operating systems," *Scientific Reports*, 2025.

[96] M. Willmann, M. Albus, F. Capozzi, D. Russell, and S. Feld, "Application of quantum annealing for scalable robotic assembly line optimization," *arXiv preprint arXiv:2412.09239*, 2024.

[97] M. J. Schuetz, J. K. Brubaker, H. Montagu, Y. van Dijk, J. Klepsch, P. Ross, A. Luckow, M. G. Resende, and H. G. Katzgraber, "Optimization of robot-trajectory planning with nature-inspired and hybrid quantum algorithms," *Physical Review Applied*, vol. 18, p. 054045, nov 2022.

[98] F. Arute, K. Arya, R. Babbush *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, pp. 505–510, 2019.

[99] J. M. Gambetta, J. M. Chow, and M. Steffen, "Building logical qubits in a superconducting quantum computing system," *npj Quantum Information*, vol. 3, p. 2, 2017.

[100] T. Monz, P. Schindler, J. T. Barreiro *et al.*, "14-qubit entanglement: creation and coherence," *Physical Review Letters*, vol. 106, no. 13, p. 130506, 2011.

[101] H.-S. Zhong, H. Wang, Y.-H. Deng *et al.*, "Quantum computational advantage using photons," *Science*, vol. 370, no. 6523, pp. 1460–1463, 2020.

[102] J. L. O'Brien, "Optical quantum computing," *Science*, vol. 318, no. 5856, pp. 1567–1570, 2007.

[103] C. L. Degen, F. Reinhard, and P. Cappellaro, "Quantum sensing," *Reviews of Modern Physics*, vol. 89, no. 3, p. 035002, 2017.

[104] L. Pezzè, A. Smerzi, M. K. Oberthaler, R. Schmied, and P. Treutlein, "Quantum metrology with nonclassical states of atomic ensembles," *Reviews of Modern Physics*, vol. 90, no. 3, p. 035005, 2018.

[105] S. Pirandola, B. R. Bardhan, T. Gehring, C. Weedbrook, and S. Lloyd, "Advances in photonic quantum sensing," *Nature Photonics*, vol. 12, no. 12, pp. 724–733, 2018.

[106] A. Luckow, M. Chen, Z. Zhang *et al.*, "Quantum computing for automotive applications," *arXiv preprint arXiv:2409.14183*, 2024.

[107] M. Jensen, U. Aseginolaza *et al.*, "Quantum-assisted automatic path-planning for robotic quality inspection," *arXiv preprint arXiv:2507.01462*, 2025.

[108] V. Bergholm, J. Izaac, M. Schuld *et al.*, "PennyLane: Automatic differentiation of hybrid quantum-classical computations," *arXiv preprint arXiv:1811.04968*, 2018.