Reinforcement Learning Quantum Neural Network Hybrids for Real-Time Supply-Chain Security: Methods, Threat Models, and a Research Roadmap

Author: Christina Evans, **Affiliation:** Research Associate, Quantum AI Center, University of Edinburgh, South Africa. **Email:** christina.evans@ed.ac.uk

Abstract

Global supply chains are increasingly automated, instrumented, and interconnected creating opportunities for real-time optimization but also novel, rapidly evolving security threats (tampering, insider fraud, diversion, adversarial manipulation of sensors and models). Reinforcement learning (RL) has emerged as a powerful paradigm for sequential decision making in dynamic supply-chain environments, enabling adaptive routing, anomaly response, and recovery actions. Simultaneously, quantum neural networks (QNNs) and other hybrid quantum-classical components promise richer representations and novel algorithmic primitives that may enhance sample efficiency, combinatorial search, and kernel expressivity in data-scarce or adversarial settings. This paper integrates these two frontiers and presents a comprehensive treatment of **RL–QNN hybrid architectures** tailored for *real-time supply-chain security*. We provide (1) formal problem definitions and threat models, (2) theoretical and practical descriptions of hybrid RL–QNN designs (policy/value parameterizations, quantum feature maps, gradient estimation), (3) reproducible training algorithms and pseudocode, (4) evaluation and adversarial robustness frameworks, (5) deployment and MLOps guidance for latency-bound environments, and (6) a detailed research roadmap prioritizing near-term hybrid pilots and longer-term fault-tolerant ambitions. We ground the discussion in recent literature on QNNs, variational quantum algorithms, quantum reinforcement learning, and RL for supply chains (Cerezo et al., 2021; Havlíček et al., 2019; Meyer et al., 2024; Yan, 2022; Correll et al., 2023). Practical recommendations emphasize measurable security outcomes, reproducibility, and interpretable governance.

Keywords: reinforcement learning; quantum neural networks; supply-chain security; adversarial robustness; MLOps; variational quantum circuits; real-time systems

1. Introduction

1.1 Motivation

Modern supply chains are cyber-physical systems: sensors (IoT), telemetry, automated warehouses, and digital marketplaces produce dense temporal signals that can be

exploited for real-time decision making. The same instrumentation, however, creates a broad attack surface ranging from tampered sensors to coordinated fraud rings that can cause financial loss, reputational damage, and operational disruption (Yan, 2022; Ma et al., 2024). Reinforcement learning (RL) provides a principled framework for sequential decision-making under uncertainty and has been successfully applied to inventory control, routing, disruption recovery, and anomaly mitigation (Rolf, 2023; Yan, 2022). Yet RL models are vulnerable to adversarial manipulation (Gleave et al., 2020; Vyas, 2024), and classical function approximators can struggle with highly combinatorial or small-label regimes present in supply-chain security tasks.

Quantum neural networks (QNNs), implemented via parameterized quantum circuits (PQCs), provide alternative inductive biases and access to high-dimensional quantum feature spaces (Havlíček et al., 2019; Cerezo et al., 2021). Hybrid architectures classical RL agents that use QNNs as policy/value approximators or QNNs as representation modules are emerging in the literature (Meyer et al., 2024; Correll et al., 2023). Hybrid RL–QNNs offer the promise of richer features (quantum kernels, amplitude encodings) for sparse-label or adversarial detection tasks, and quantum subroutines (QAOA/annealers) for combinatorial subproblems such as resilient routing under attack.

1.2 Scope and Contributions

This manuscript focuses on research methods and applied design for **RL–QNN hybrid systems** targeted at *real-time supply-chain security*. The primary contributions are:

1. **Formal problem framing** linking supply-chain security objectives to Markov Decision Processes (MDPs) and multi-agent extensions under adversary interactions.
2. **Architectural taxonomy** for hybrid RL–QNN systems (policy parameterizations quantum policy/value networks, quantum embedding layers, quantum-assisted combinatorial subroutines).
3. **Reproducible algorithms and pseudocode** for actor-critic and policy-gradient RL using QNN modules; practical gradient estimation (parameter-shift), shot budgets, and hybrid optimizers.
4. **Adversarial threat modeling and robustness methodology**: attacker goals, capabilities, and metricized defenses (adversarial training, detection thresholds, game-theoretic reserves).
5. **Evaluation plan**: simulation benchmarks, off-policy evaluation with logged data, sensitivity analyses, and security KPIs.

6. **Deployment & MLOps blueprint** for low-latency inference and resilient training, including model governance, logging, and human-in-the-loop escalation.
7. **Roadmap and prioritized research agenda** bridging NISQ-era pilots and long-term fault-tolerant goals.

We integrate and cite the most relevant literature through 2024 (Biamonte et al., 2017; Havlíček et al., 2019; Cerezo et al., 2021; Meyer et al., 2024; Correll et al., 2023) and include practitioner resources on RL for supply chains (Yan, 2022; Rolf, 2023; Ma et al., 2024), as well as domain security work (Samuel, 2021, 2023).

2. Background: Reinforcement Learning, QNNs, and Supply-Chain Security

This section gives compact background: MDPs and RL basics, parameterized quantum circuits and QNN properties, and supply-chain security characteristics that affect modeling choices.

2.1 Reinforcement Learning for Sequential Decision-Making

An MDP is defined as ((\mathcal{S}, \mathcal{A}, P, r, \gamma)) where (\mathcal{S}) is the state space, (\mathcal{A}) the action space, (P(s'|s,a)) transition probabilities, (r(s,a)) reward, and (\gamma) the discount factor. The RL agent seeks policy (\pi_\theta(a|s)) parameterized by (\theta) to maximize expected return (J(\theta)=\mathbb{E}[\sum_{t=0}^\infty \gamma^t r_t]) (Sutton & Barto, canonical text). In supply chains, states often include inventory levels, lead times, shipments, telemetry, and anomaly indicators; actions include rerouting, hold/release orders, isolation of nodes, and investigative/forensic triggers. Multi-agent extensions model different organizations (manufacturer, carrier, retailer) or distributed controllers (Littman, 1994; Foerster et al., 2016).

2.2 Threat Landscape & Security Objectives in Supply Chains

Real-time supply-chain security problems have several characteristic properties:

- **Heterogeneous observations:** sensor measurements (noisy), transactional records, third-party feeds (weather, port status).
- **Partial observability and delays:** delayed confirmations, censored signals.
- **Adversarial actors:** tamperers, insiders, fraud rings able to manipulate observations or transactions.
- **Cost asymmetry:** false positives (unnecessary holds) have economic costs; false negatives (missed attacks) can cause severe downstream disruption.

Key security objectives are to **detect**, **mitigate**, and **recover** from security incidents with bounded operational cost and latency. RL lends itself to policies that trade off economic loss and mitigation overhead.

2.3 Quantum Neural Networks (QNNs) & Variational Quantum Circuits

QNNs are instantiated via parameterized quantum circuits (PQCs) sequences of parameterized single- and two-qubit gates that map classical inputs encoded into quantum states to expectation-value outputs (Cerezo et al., 2021; Havlíček et al., 2019). Important QNN design choices:

- **Encoding/feature maps:** angle encoding, amplitude encoding, basis encoding (Havlíček et al., 2019).
- **Ansatz/variational layers:** hardware-efficient ansatz vs problem-inspired ansatz (Cerezo et al., 2021).
- **Measurements and readout:** expectation values of observables (e.g., Pauli Z) often produce scalar outputs that are then postprocessed by classical layers.

QNNs can act as expressive feature transformers (quantum kernels) or direct function approximators for policies/values in RL (Meyer et al., 2024). Practicalities for NISQ-era devices include circuit depth restrictions, noise, shot/noise tradeoffs, and barren plateau phenomena (Cerezo et al., 2021; Zhang, 2024).

3. Problem Formulation: Real-Time Supply-Chain Security as an RL Task

We formalize the supply-chain security problem suitable for RL–QNN hybrids.

3.1 State and Observation Spaces

Define the agent's observation o_t as an aggregation:

$$o_t = \{ I_t, T_t, S_t, F_t, E_t \}$$

- I_t: Inventory vectors per node
- T_t: Telemetry (sensor streams, device health)
- S_t: Shipment/visibility (GPS, ETAs)
- F_t: Fraud/fingerprint features (transaction anomalies)
- E_t: Exogenous context (weather, port status)

Observation dimensionality is often large and heterogeneous; QNNs are used as compact representation modules by encoding suitably preprocessed classical vectors into qubit states.

3.2 Action Space

Typical action primitives include:

- **Mitigation actions:** quarantine shipment, reroute, hold, rerank supplier.
- **Investigative actions:** request forensic inspection, escalate to human operator.
- **Proactive adjustments:** adjust reorder quantity, preemptive shipments.

Action selection must respect latency budgets: high-frequency decisions (e.g., immediate hold) require millisecond to second inference.

3.3 Reward Design

Design a reward that balances security outcomes and operational costs:

[
r_t = -\alpha \cdot C_{breach}(t) - \beta \cdot C_{false_positives}(t) - \gamma \cdot C_{delay}(t) + \delta \cdot \text{RecoveryBenefit}(t)
]

Weights (\alpha,\beta,\gamma,\delta) reflect business priorities. For detection tasks, reward may be sparse, motivating shaped rewards, auxiliary objectives (prediction of risk scores), or constrained RL formulations.

3.4 Adversary and Game Model

We model adversary (A) as an agent with capabilities to interfere with observations (sensor spoofing), manipulate transactions, and adapt strategies. The environment becomes a (partially observable) stochastic game; we consider **worst-case adversary** formulations (minimax) and stochastic adversaries (best-response learning). Adversary modeling is central to robust training and evaluation (Gleave et al., 2020; Vyas, 2024).

4. Architectural Patterns for RL–QNN Hybrids

This section catalogs candidate hybrid architectures and the rationale for each.

4.1 QNN as Representation Learner (State Encoder)

Pattern: Preprocess raw features (\mathbf{x}) with a classical encoder (E_{c}) producing compact vector (z\in\mathbb{R}^d) (d small). Encode (z) into a quantum state (|\phi(z)\rangle) (e.g., angle or amplitude encoding). Apply PQC (U(\theta)) and measure

expectation values to yield transformed features (q=z' \in\mathbb{R}^k). Then feed (q) into classical policy/value heads.

Rationale: QNN feature maps can separate classes in Hilbert space and may improve small-label generalization for anomaly detection or loss-sensitive decisions (Havlíček et al., 2019).

Caveats: Encoding costs and measurement noise; requires tight design to keep latency acceptable.

4.2 QNN as Policy Network (Direct Action Parameterization)

Pattern: Parameterize policy (\pi_\theta(a|s)) by a QNN: (\pi_\theta(a|s) = \mathrm{softmax}(f_\theta(s))) where (f_\theta) results from QNN measurements. Use policy gradient / actor-critic updates with gradient estimation via parameter-shift or stochastic estimators.

Rationale: QNNs may produce richer nonlinear mappings for complex action mappings (Meyer et al., 2024). Works have implemented VQCs for deep RL (Chen et al., 2020; Chen et al., 2024).

Caveats: Policy gradient variance amplified by shot noise; need careful estimator budget and baselines.

4.3 Hybrid Pipeline: Classical Candidate + QNN Re-ranking (Low-Latency)

Pattern: For actions requiring millisecond response (e.g., immediate hold or release), perform classical fast candidate generation and then use QNN re-ranking in an asynchronous cached manner or for re-scoring top K candidates.

Rationale: Reduces quantum inference calls, keeps real-time path classical while leveraging quantum enhancement for high-value decisions.

4.4 Quantum-Assisted Combinatorial Subroutines

Pattern: Use QAOA or quantum annealing for discrete combinatorial subproblems (bundling, routing under compromised nodes). Use hybrid solver to propose candidate solutions that are validated and refined by classical heuristics.

Rationale: When discrete combinatorics dominate runtime (routing under constraints), quantum annealing/QAOA can provide candidate sets for RL to evaluate (Correll et al., 2023; Weinberg et al., 2022).

5. Learning Algorithms and Practical Training Procedures

We now present concrete algorithms and practical considerations for training RL agents whose function approximators include QNNs.

5.1 Optimization & Gradient Estimation for QNN Parameters

The classical optimizer updates (\theta) based on gradients estimated by the **parameter-shift rule** when gates are single-parameter rotations:

[
\frac{\partial \langle \hat{O} \rangle}{\partial \theta_i} = \frac{1}{2} \Big(\langle \hat{O} \rangle_{\theta_i+\frac{\pi}{2}} - \langle \hat{O} \rangle_{\theta_i-\frac{\pi}{2}} \Big)
]

(Works when parameterized ansatz gates satisfy certain properties see Cerezo et al., 2021; Cornelissen, 2018; Meyer et al., 2024). For noisy devices, gradient variance increases with shot noise; gradient-free optimizers (SPSA, COBYLA) are viable alternatives.

5.2 Actor-Critic with QNN Critic (Algorithm)

Below is a reproducible high-level pseudocode for a hybrid actor-critic algorithm where the *actor* is classical and the *critic* is a QNN (alternating pattern is also common).

Algorithm 1: Hybrid Actor-Critic (Classical Actor, QNN Critic)

Inputs: Env E, classical actor π_φ(a|s), QNN critic Q_θ(s)

Hyperparams: episodes N, steps T, batch size B, shots S

Initialize φ (actor parameters), θ (QNN parameters)

for episode = 1..N:

  s0 <- E.reset()

  trajectory = []

  for t = 0..T-1:

    a_t ~ π_φ(.|s_t)

    s_{t+1}, r_t, done <- E.step(a_t)

    trajectory.append((s_t, a_t, r_t, s_{t+1}))

```
    if done: break
  # Compute returns and/or GAE advantages
  for minibatch in sample_batches(trajectory, B):
    # Critic update (QNN)
    for s in minibatch.states:
      z = classical_encoder(s)
      prepare quantum state |φ(z)□ with S shots
      Q_val = measure_QNN(θ, |φ(z)□, S)
    loss_Q = MSE(Q_val, bootstrap_targets)
    θ <- θ - η_Q * grad_estimate(loss_Q, θ)  # parameter-shift or SPSA

    # Actor update (classical)
    advantages = bootstrap_targets - Q_val.detach()
    loss_actor = -E[ log π_φ(a|s) * advantages ]
    φ <- φ - η_actor * ∇_φ loss_actor
```

Notes: measurement shot count S controls estimator variance; QNN backprop is via parameter-shift; detach prevents actor gradients flowing into QNN.

5.3 Policy Gradient with Quantum Policy (Algorithm)

If the **policy itself** is quantum-parameterized, modify the actor update to use quantum gradients estimated via parameter shift. The policy gradient becomes:

[
\nabla_\theta J(\theta) = \mathbb{E} \Big[\nabla_\theta \log \pi_\theta(a|s) \cdot A(s,a) \Big]
]

Compute (\nabla_\theta \log \pi_\theta) by differentiating measured outputs (with parameter-shift). Use baselines to reduce variance.

5.4 Measurement & Shot Budgeting

Measurement cost is a central engineering parameter. Use the following guidelines:

- **Training:** allow larger shot budgets for critic updates (stability). Consider progressive shot schedules (fewer shots early, increasing later).
- **Inference:** keep shots minimal; use cached re-scoring for top K candidates.
- **Hybrid variance control:** combine multiple measurement estimators and classical surrogates.

Citations: approaches for shot-scheduling, surrogate models, and optimization heuristics appear in the quantum ML literature (Cerezo et al., 2021; Chen et al., 2020).

6. Adversarial Threats & Robustness for RL–QNN Systems

Security is the central purpose; here we map potential attacks to defense strategies.

6.1 Threat Taxonomy

1. **Observation Spoofing:** attacker modifies telemetry to mask diversion or mislead the agent.
2. **Data Poisoning:** contaminated training data (logs) bias agent behavior; especially dangerous for online RL.
3. **Adversarial Policies:** adversarial agents that manipulate environment dynamics to create misleading trajectories (Gleave et al., 2020).
4. **Model Extraction & Inference Attacks:** attackers probe policy APIs to infer sensitive patterns or extract model parameters.
5. **Supply-chain compromises of compute fabric:** manipulation or denial of quantum/cloud resources.

6.2 Defense Approaches

- **Adversarial Training:** simulate attack policies and include them in training as opponents (Gleave et al., 2020; Gross, 2023).
- **Robust RL formulations:** distributionally robust RL or constrained optimization where worst-case losses bounded (Duchi et al., Literature).
- **Data provenance and secure ingestion:** tamper-evident logs (blockchain/zero-trust) for audit trails (Ma et al., 2024; Samuel, 2021).
- **Differential privacy:** limit information leakage from policy updates and prevent model inversion (McMahan et al., 2017; Cummings on DP).

- **Ensemble & detection:** monitor ensemble disagreement (classical and QNN) as a signal of anomalous inputs.

6.3 Quantum-Specific Threats

- **Hardware level attacks:** supply-chain attacks on QPU firmware; side channels from quantum cloud providers (Beaudoin et al., 2022).
- **Measurement tampering:** adversary influencing measurement results in a shared cloud QPU context mitigated by authenticated channels and cryptographic attestations.

7. Experimental Design, Benchmarks, and Evaluation

This section prescribes rigorous evaluation methodology for RL–QNN hybrids.

7.1 Synthetic & Industry-Scale Simulation Environments

- **Simulators:** build high-fidelity supply-chain simulators modeling stochastic demand, transit delays, and sensor noise. Correll et al. (2023) and Weinberg et al. (2022) demonstrate simulation-in-the-loop evaluations for routing tasks.
- **Benchmarks:** propose a benchmark suite with scenarios:
  - *Normal operations*: no adversary.
  - *Sensor spoofing*: time-windowed spoofing attacks.
  - *Coordinated fraud*: multi-node collusion to divert shipments.
  - *Supply shocks*: sudden disruption to upstream nodes.
- **Data fidelity:** use anonymized enterprise logs for curriculum learning and domain transfer.

7.2 Baselines

- **Classical RL baselines:** DQN, PPO, SAC, and multi-agent RL (MADDPG/COMA).
- **Hybrid baselines:** classical encoders with classical policy but quantum-assisted re-ranking.
- **Combinatorial baselines:** Mixed Integer Programming (Gurobi), classical annealing heuristics.

7.3 Metrics

- **Security metrics:**
  - Detection Rate (True Positive Rate for compromises).
  - Time-to-Mitigation (seconds/minutes).
  - False Positive Rate (operational cost).
  - Economic Impact Averted (monetary).
- **RL metrics:**
  - Cumulative reward, sample efficiency (episodes to x% performance).
- **Robustness metrics:**
  - Worst-case regret under adversarial policies (Gleave et al., 2020).
  - Attack success rate against learned policies.
- **Quantum resource metrics:**
  - QPU time, shots per update, qubit count, circuit depth.

7.4 Off-Policy / Counterfactual Evaluation

For logged historical data, use off-policy estimators (Inverse Propensity Scoring, Doubly Robust estimators) to estimate policy value and mitigate evaluation bias (Joachims et al., 2017).

8. Case Study: Simulated RL–QNN Pipeline for Sensor Spoofing Detection and Response

We present a prototypical experimental case (simulation) illustrating training and evaluation flow.

8.1 Scenario

- A two-tier supply chain (warehouse + carrier network). Sensor spoofing adversary intermittently manipulates GPS and temperature sensors to hide diversion and tamper events.
- The agent must decide per-shipment whether to continue routing, perform remote verification, or require human inspection.

8.2 Model

- **State encoder:** classical CNN/MLP for sensor sequences → 32-dim vector.

- **QNN module (representation):** angle encoding into 6 qubits, 4 variational layers, readout yields 4 features.
- **Actor:** classical MLP mapping [classical features + QNN features] to action logits.
- **Critic:** classical MLP.

8.3 Training

- Hybrid actor-critic with QNN encoder trained end-to-end. Shot schedule: S=512 early, S=2048 late for critic stability. Optimizers: Adam for classical, SPSA for QNN (alternatively parameter-shift when available).

8.4 Results (Hypothetical / Suggested Reporting)

- **Sample efficiency:** hybrid model reaches baseline reward in ~15% fewer episodes on small-label spoofing regimes (consistent with small-sample benefits reported by quantum kernel literature) report statistically across seeds.
- **Robustness:** under adversarial policy A*, hybrid agent maintains lower false-negative rate than classical baseline ($p<0.05$).
- **Quantum resources:** average per-update wall time dominated by QPU latency; demonstrate use of simulated QNNs for algorithmic development and cloud QPU for small, targeted validation runs (Correll et al., 2023).

(Actual empirical numbers should be produced by implementing the described pipeline and running controlled experiments.)

9. MLOps, Governance, and Deployment Considerations

9.1 Latency, Orchestration, and Hybrid Inference

- **Real-time constraints:** use QNN inference sparingly in the critical path. Preferred patterns include asynchronous re-scoring, top-K batch re-scoring, and cached QNN results.
- **Orchestration:** integrate QPU calls via cloud providers (IBM, IonQ, Rigetti) or on-prem quantum accelerators using a unified middleware (PennyLane/Qiskit) shielding the classical stack.
- **Edge vs Cloud:** perform lightweight prefiltering at edge; heavy compute offloaded to cloud/quantum backends. Enforce authenticated channels and attestations.

9.2 Model Registry and Versioning

- Track QNN circuit definitions, parameter values, shot budgets, and hardware backends.
- Produce **model cards** and **quantum model cards** (Mitchell et al., 2019; adapted) capturing intended use, limitations, fairness tests, and security constraints.

9.3 Logging and Post-mortem Analysis

- Log raw inputs, QNN readouts, classical features, and final actions for every decision. To preserve privacy, log aggregates and use authenticated storage with tamper evidence (Samuel, 2021).

9.4 Human-in-the-Loop & Escalation

- Define automation tiers: fully automated for low-cost mitigation; human-in-loop for medium/high impact actions. Establish clear SLAs for operator review.

10. Evaluation of Practicality: When to Pilot RL–QNN Hybrids

We propose heuristics to select pilot problems:

1. **Small-label or high feature-complexity regimes** where classical models require prohibitive labeled data and quantum kernels may improve reachability (Havlíček et al., 2019).
2. **Moderate combinatorial subproblems** (vehicle routing, regional assortment under constraints) where hybrid annealing/QAOA can propose candidate sets (Correll et al., 2023; Weinberg et al., 2022).
3. **High-value security decisions** where small accuracy gains produce outsized economic benefit.
4. **Availability of realistic simulators** and enterprise willingness to run online randomized experiments or shadow trials.

11. Challenges, Limitations, and Open Problems

11.1 Hardware Limitations & Noisy Devices

NISQ devices have limited qubit counts, limited connectivity, and noise, constraining circuit depth and expressivity (Preskill, 2018; Cerezo et al., 2021).

11.2 Barren Plateaus & Optimization Complexity

Parameter landscape issues such as barren plateaus complicate training and may demand problem-aware ansatz and initialization (Cerezo et al., 2021; Zhang, 2024).

11.3 Explainability & Regulatory Compliance

QNN internals are less interpretable than classical models. For regulated decisions in supply chains (e.g., delaying shipments for inspection), design surrogate explanations and human-readable decision artifacts (Mitchell et al., 2019).

11.4 Security & Adversarial Robustness Remain Open

Adversarial policies that target RL agents in complex supply chains require ongoing research in robust RL, game-theoretic defenses, and economic modeling of false positive/negative costs (Gleave et al., 2020; Gross, 2023).

12. Roadmap and Prioritized Research Agenda (Near, Mid, Long Term)

12.1 Near Term (0–24 months)

- **Benchmark suite creation**: simulated supply-chain security tasks with adversarial scenarios.
- **Hybrid prototypes**: QNN feature encoders with classical RL policies in simulation; proof-of-concept deployments in shadow mode (Correll et al., 2023).
- **Tooling**: standardized interfaces (PennyLane, PennyLane-RL wrappers), shot scheduling utilities, and reproducible experiment repositories.

12.2 Mid Term (2–5 years)

- **Federated hybrid learning**: privacy-preserving cross-platform collaborations for fraud detection (McMahan et al., 2017; Samuel, 2021).
- **Adversarial RL defenses**: combine robust RL and quantum encodings to resist adaptive attackers.
- **Hardware-in-the-loop studies**: increasing use of cloud QPUs for targeted subroutines with cost/benefit analysis.

12.3 Long Term (5+ years)

- **Fault-tolerant quantum RL components** for combinatorial optimization at scale.
- **Standardized governance** for quantum-augmented automated decision systems.
- **Economic integration**: evaluating TCO and return on quantum adoption for enterprise supply chains.

13. Conclusion

Hybrid RL–QNN architectures provide a promising direction for enhancing real-time supply-chain security by combining adaptive sequential decision making with novel representation and combinatorial optimization primitives available in quantum computing. However, benefits are conditional on problem selection, simulator fidelity, careful experimental validation, and robust governance. Near-term value will most likely arise from **targeted hybrid pilots** that use QNNs as representation modules or quantum annealers for discrete subroutines, supported by classical RL agents for the control loop. Rigorous adversarial evaluation, measurement budgeting, and explainability practices are essential to ensure operational safety and regulatory compliance. The research roadmap proposed here prioritizes reproducible benchmarks, hybrid tool chains, and federated privacy-preserving experiments as the logical next steps towards responsible industrial adoption.

References

1. Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., & Lloyd, S. (2017). Quantum machine learning. *Nature*, 549(7671), 195–202. https://doi.org/10.1038/nature23474

2. Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K., ... & Coles, P. J. (2021). Variational quantum algorithms. *Nature Reviews Physics*, 3(9), 625–644. https://doi.org/10.1038/s42254-021-00348-9

3. Chen, S. Y.-C., Yang, C.-H. H., Qi, J., Chen, P.-Y., Ma, X., & Goan, H.-S. (2020). Variational quantum circuits for deep reinforcement learning. *IEEE Access*, 8, 141007–141024. https://doi.org/10.1109/ACCESS.2020.3017379

4. Chen, H.-Y., et al. (2024). Deep Q-learning with hybrid quantum neural network on [application]. *Quantum Machine Intelligence/Journal* (see related implementations of hybrid QNN RL). [Use this as a literature pointer implementers should consult the 2024 hybrid QNN RL literature for concrete examples.] (Note: consult the cited 2024 implementations such as Chen et al., 2024 for applied templates.)

5. Correll, R., Weinberg, S. J., Sanches, F., Ide, T., & Suzuki, T. (2023). Quantum Neural Networks for a Supply Chain Logistics Application. *Advanced Quantum Technologies*, 6(7), 2200183. https://doi.org/10.1002/qute.202200183

6. Dunjko, V., & Briegel, H. J. (2018). Machine learning & artificial intelligence in the quantum domain: Recent progress and outlook. *Applied Physics Reviews* / referenced via QRL surveys. (Background references on potential quantum enhancements for learning.)

7. Fatunmbi, T. O. (2023). Revolutionizing multimodal healthcare diagnosis, treatment pathways, and prognostic analytics through quantum neural networks. World Journal of Advanced Research and Reviews, 17(01), 1319-1338. https://doi.org/10.30574/wjarr.2023.17.1.0017

8. Fatunmbi, T. O. (2022). Quantum-Accelerated Intelligence in E-Commerce: The Role of AI, Machine Learning, and Blockchain for Scalable, Secure Digital Trade. International Journal of Artificial Intelligence & Machine Learning, 1(1), 136-151. https://doi.org/10.34218/IJAIML_01_01_014

9. Gleave, A., Dennis, M., Wild, C., & others (2020). Adversarial Policies: Attacking Deep Reinforcement Learning. *OpenReview.* https://openreview.net/forum?id=HJgEMpVFwB

10. Gross, D., et al. (2023). Targeted Adversarial Attacks on Deep Reinforcement Learning. *Conference/ArXiv* (examples of adversarial attacks in RL). [See Gross (2023) for targeted attack techniques and defense evaluation.]

11. Havlíček, V., Córcoles, A. D., Temme, K., et al. (2019). Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747), 209–212. https://doi.org/10.1038/s41586-019-0980-2

12. Joachims, T., Swaminathan, A., & Schnabel, T. (2017). Deep learning with logged bandit feedback. *Conference literature / counterfactual evaluation methods.* (See classical RL offline evaluation techniques.)

13. Ma, Z., Chen, X., Sun, T., Wang, X., Wu, Y. C., & Zhou, M. (2024). Blockchain-Based Zero-Trust Supply Chain Security Integrated with Deep Reinforcement Learning for Inventory Optimization. *Future Internet*, 16(5), 163. https://doi.org/10.3390/fi16050163

14. McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & Aguera y Arcas, B. (2017). Communication-efficient learning of deep networks from decentralized data. *Proceedings of AISTATS 2017.* (Federated learning foundational paper.)

15. Meyer, N., Ufrecht, C., Periyasamy, M., Scherer, D. D., Plinge, A., & Mutschler, C. (2024). A Survey on Quantum Reinforcement Learning. *arXiv:2211.03464v2.* https://arxiv.org/abs/2211.03464

16. Preskill, J. (2018). Quantum computing in the NISQ era and beyond. *Quantum*, 2, 79. https://doi.org/10.22331/q-2018-08-06-79

17. Rolf, B. (2023). A review on reinforcement learning algorithms and applications in supply chain management. *[Journal Review Article]*. (Comprehensive survey of RL for SCM see for domain adaptation and benchmarking.) consult for RL for supply chains.

18. Samuel, A. J. (2021). Cloud-Native AI solutions for predictive maintenance in the energy sector: A security perspective. *World Journal of Advanced Research and Reviews*, 9(03), 409–428. https://doi.org/10.30574/wjarr.2021.9.3.0052

19. Samuel, A. J. (2023). A Comprehensive Frameworks for Fraud Crime Detection and Security: Leveraging Neural Networks and AI. *Journal of Science, Technology and Engineering Research*, 1(4), 15–45. https://doi.org/10.64206/m3jxre09

20. V. Zhang, Y. et al. (2024). Reliability Research on Quantum Neural Networks. *Electronics*, 13(8), 1514. (Analyses on QNN reliability and training issues in near-term devices.)

21. Weinberg, S. J., Sanches, F., Ide, T., Kamiya, K., & Correll, R. (2022). Supply Chain Logistics with Quantum and Classical Annealing Algorithms. *arXiv:2205.04435.* https://arxiv.org/abs/2205.04435

22. Yan, Y. (2022). Reinforcement learning for logistics and supply chain management: Methodologies, state of the art, and future opportunities. *Journal/Review* see survey for domain-specific RL challenges in SCM.

23. Zhang, Y., et al. (2024). Reliability Research on Quantum Neural Networks. *Electronics*, 13(8), 1514. (Discusses training reliability, noise resilience, and experiment reproducibility for QNNs.)

24. Zhou, M. G., et al. (2023). Quantum Neural Network for Quantum Neural Computing. *Research (article)*. https://doi.org/10.34133/research.0134