Research on Construction Schedule Optimization Based on Q-Learning Algorithm

Yang Xu*

School of Business Administration, Liaoning Technical University, Huludao, Liaoning, 125105, China
*E-mail: 447001224@qq.com

Abstract. Traditional schedule management methods such as the Critical Path Method (CPM) and Program Evaluation and Review Technique (PERT) have inherent limitations in dynamic construction environments due to their static planning nature and poor adaptability to uncertainties. This paper proposes a schedule optimization model based on the Q-Learning algorithm, which achieves dynamic resource allocation decisions through the construction of a state space, action space, and reward mechanism. Taking a large-scale commercial complex construction project as a case study, the experimental results demonstrate that compared with the traditional CPM method, the Q-Learning model reduces the total project duration by 15.3%, improves resource utilization by 22.1%, and effectively mitigates schedule delay risks. Under random disturbance scenarios including weather impacts and material shortages, the model exhibits strong robustness, with schedule recovery time reduced by 40% compared to manual adjustment. This research provides an intelligent scheduling tool for construction enterprises, promoting the transformation from static planning to dynamic optimization in schedule management.

Keywords: Construction schedule; Q-Learning; Resource optimization; Dynamic scheduling

1 Introduction

Construction schedule management faces severe challenges due to project complexity and environmental uncertainties. Traditional methods like the Critical Path Method (CPM) and Program Evaluation and Review Technique (PERT) are based on deterministic assumptions and cannot effectively handle dynamic factors such as resource conflicts, process coupling, and environmental variations. Statistics indicate that approximately 70% of construction projects in China experience varying degrees of schedule delays, with improper resource allocation being a significant contributing factor. [2]

Reinforcement learning enables intelligent agents to learn optimal strategies through environment interaction and has shown remarkable advantages in dynamic decision-making problems. Q-Learning, as a classical model-free reinforcement learning algorithm with proven convergence properties, is particularly suitable for schedul-

L. W. Koon et al. (eds.), *Proceedings of the 2026 7th International Conference on Management Science and Engineering Management (ICMSEM 2026)*, Advances in Economics, Business and Management Research 399,
https://doi.org/10.2991/978-94-6239-758-3_14

ing optimization problems involving unknown environmental dynamics and complex state spaces. This study proposes a schedule optimization model based on the Q-Learning algorithm to address the limitations of traditional methods and provide intelligent decision support for construction schedule management. [3][4]

Previous research has applied intelligent optimization algorithms such as the Genetic Algorithm (GA) and Ant Colony Optimization (ACO) to resource-constrained project scheduling problems. However, the systematic application of reinforcement learning, particularly the Q-Learning algorithm, in construction schedule management remains underexplored. This study contributes by: (1) systematically applying Q-Learning to construction schedule optimization, enabling autonomous learning and dynamic [1][5]

adjustment of resource allocation strategies; (2) designing a multi-objective optimization reward mechanism that balances schedule, cost, and resource utilization considerations.

2 Research Methods and Model Construction

2.1 Q-Learning Algorithm Principle

Q-Learning is a value-based reinforcement learning algorithm that learns the optimal action-value function Q(s,a) through environment interaction. The algorithm maintains a Q-table storing expected cumulative returns for each state-action pair, updated according to the Bellman equation: [3]

$$Q(s,a) \leftarrow Q(s,a) + \alpha[\quad |r + \gamma \cdot max_{a'} Q(s',a') - Q(s,a)]$$

where s is current state, a is action taken, r is immediate reward, s' is next state, α is learning rate, γ is discount factor.

Action selection employs an ε-greedy strategy that balances exploration and utilization: with probability ε, the agent randomly selects an action (exploration); with probability 1-ε, the agent selects the action with the maximum Q-value (utilization).

2.2 MDP Model Construction

State Space S

The state space is designed to capture all relevant project information and is discretized into the following dimensions:

&Project progress: Completion percentage (discretized into 10% intervals), earliest start/finish times (discretized into weekly intervals)

&Resource status: Labor availability (discretized into 5-person increments), equipment utilization (discretized into 10% intervals), material stock levels (discretized into 10% of total requirements)

&Time status: Current period (discretized into weekly intervals), remaining duration (discretized into monthly intervals), cumulative delays (discretized into daily intervals)

&Environmental conditions: Weather conditions (categorical: normal, light rain, heavy rain, extreme weather)

Each dimension is encoded into discrete states, resulting in a total of 12,600 possible unique states in the Q-table.

Action Space A

The action space consists of discrete resource allocation decisions, including:

&Labor adjustment: Increase/decrease labor by 5 workers, reallocate labor between processes

&Equipment allocation: Assign/reallocate specific equipment types, adjust equipment usage hours

&Process priority: Adjust process precedence relationships within allowable constraints

&Emergency measures: Approve overtime (up to 2 hours/day), mobilize additional teams, or modify construction methods

Each action is encoded as a discrete choice in the Q-table, resulting in 24 possible actions.

Reward Function R

The reward function uses a multi-objective weighted form to guide the learning direction:

$$R = -w_1 \cdot \Delta T - w_2 \cdot \Delta C + w_3 \cdot U - w_4 \cdot P$$

where ΔT is the schedule change (days), ΔC is the cost change (yuan), U is the resource utilization rate, P is the penalty for constraint violations, and w_1-w_4 are weight coefficients set to 0.4, 0.3, 0.2, and 0.1 respectively.

2.3 Algorithm Implementation

The Q-Learning schedule optimization model implements the following steps:

Step 1: Initialize the environment, define the project network plan, resource constraints, and learning parameters;

Step 2: Initialize the Q-table, setting all Q(s,a) = 0; Step 3: For each episode, initialize the project state s;

Step 4: For each decision point: (1) Select action a via the ε-greedy strategy; (2) Execute action a, obtain the next state s' and reward r; (3) Update the Q-table according to the Bellman equation; (4) Update the state $s \leftarrow s'$;

Step 5: Check the termination condition (schedule completion or maximum iterations), return to Step 4 if not met;

Step 6: Output the optimal scheduling strategy corresponding to the maximum cumulative reward.

Parameter settings: learning rate $\alpha = 0.1$, discount factor $\gamma = 0.9$, exploration rate ε decays exponentially from 0.2 to 0.01 over 5000 episodes, reward weights $w_1=0.4$, $w_2=0.3$, $w_3=0.2$, $w_4=0.1$.

3 Case Study

3.1 Project Background

A large-scale commercial complex project in Shanghai with a total construction area of 180,000 m², 3 underground floors, 32 above-ground floors, and a planned duration of 28 months. The project involves 24 main construction processes including earthwork, foundation, main structure, secondary structure, decoration, and installation. Key characteristics include multiple resource constraints, complex process relationships with parallel execution, and high environmental uncertainty. The project basic information is summarized in Table 1.

Table 1. Project basic information

Indicator	Value
Total construction area	180,000 m²
Total duration	28 months
Number of main processes	24
Resource types	Labor (8 skill levels), Equipment (12 types), Materials (15 categories)
Historical delay rate	18.5%
Historical resource utilization	67.3%

3.2 Model Application

The network plan model was constructed defining process logical relationships. Training settings: 5000 episodes, maximum 1000 steps per episode, batch size 32. Training results: model converged after approximately 3500 episodes with optimal cumulative reward of 1245.6, training time 45 minutes. The optimization results are presented in Table 2.

Table 2. Optimization results comparison

Indicator	CPM method	Q-Learning optimized	Improvement
Total duration (days)	840	711	15.3%
Resource utilization	67.3%	89.4%	+22.1%
Labor cost (million yuan)	12.6	11.8	-6.3%
Schedule delay risk	18.5%	5.2%	-71.9%

3.3 Dynamic Performance Test

Random disturbance scenarios were introduced to verify dynamic adjustment capability: The dynamic adjustment performance is shown in Table 3.

Scenario 1: 5-day continuous rain, outdoor process suspension;

Scenario 2: Material shortage, 7-day concrete supply delay;

Scenario 3: Equipment failure, 2 tower cranes out of service for 3 days.

Table 3. Dynamic adjustment performance

Scenario	Method	Schedule extension (days)	Recovery time (days)
Rainy weather	Manual	8	12
	Q-Learning	3	6
Material shortage	Manual	10	15
	Q-Learning	4	8
Equipment failure	Manual	6	9
	Q-Learning	2	5

Results show Q-Learning model significantly outperforms manual adjustment, with average 58.3% reduction in schedule extension and 44.4% shorter recovery time. (see Table 3).

3.4 Algorithm Comparison

Comparative experiments with Genetic Algorithm (GA) and Ant Colony Optimization (ACO) were conducted: The comparative results are summarized in Table 4.

Table 4. Algorithm comparison

Algorithm	Optimal duration (days)	Convergence time (s)	Robustness
CPM	840	-	Low
GA	735	320	Medium
ACO	748	285	Medium
Q-Learning	711	2700	High

Q-Learning achieves best optimization effect with highest robustness in dynamic environments, though training time is longer. [1][5] (as detailed in Table 4).

4 Conclusions and Future Work

4.1 Conclusions

This study successfully applies Q-Learning algorithm to construction schedule optimization, demonstrating that:

1.Q-Learning effectively solves dynamic resource allocation problems in schedule management through autonomous learning of optimal strategies via environment interaction;

2.Compared with traditional CPM method, Q-Learning reduces total duration by 15.3%, improves resource utilization by 22.1%, and decreases schedule delay risk by 71.9%; (as shown in Table 2).

3.The model shows superior performance in dynamic adjustment under random disturbances with faster response and better outcomes than manual adjustment;

4.The proposed method provides an intelligent scheduling tool enabling dynamic optimization of schedule management, improving project delivery reliability and resource efficiency.

4.2 Limitations and Future Research

Current limitations include: (1) state space dimensionality increases exponentially for projects with over 50 processes; (2) dynamic adjustment capability depends on accurate real-time data; (3) model focuses on single-project optimization without multi-project coordination.

Future research directions: (1) Introduce Deep Q-Network (DQN) to handle high-dimensional state spaces for large-scale complex projects; (2) Extend to multi-project scheduling via multi-agent reinforcement learning; (3) Integrate with BIM and IoT for real-time data acquisition and digital twin-based management; (4) Apply transfer learning to transfer strategies between similar projects. [3][4]

4.3 Summary

This study realizes dynamic optimization of construction schedule management through Q-Learning algorithm, providing an intelligent and adaptive tool for the construction industry. Future integration of advanced reinforcement learning algorithms and real-time data technologies will promote transformation from static planning to intelligent decision-making, enhancing overall industry efficiency and competitiveness.

5 Discussion

The experimental results provide compelling evidence that the Q-Learning algorithm offers significant advantages over conventional scheduling methods in dynamic construction environments. Several key aspects merit further discussion regarding convergence behavior, reward function design, and practical applicability.

5.1 Convergence and Reward Function Analysis

The model achieved stable convergence after approximately 3,500 episodes (α=0.1, γ=0.9, ε decaying from 0.2 to 0.01 over 5,000 episodes). The exponential ε-decay strategy proved critical in balancing exploration and exploitation: early episodes with high ε values enabled thorough state-action space exploration, while gradual reduction to 0.01 ensured convergence to near-optimal deterministic policies. Sensitivity analysis varying α in {0.01,0.05,0.1,0.2,0.5} and γ in {0.8,0.85,0.9,0.95,0.99} confirmed the chosen parameter combination produced the fastest convergence and highest cumulative reward. The multi-objective reward function with weights (0.4,0.3,0.2,0.1) proved robust to moderate variations within 20%, though extreme configurations prioritizing cost over schedule reduced duration improvement to only 6-8%, compared to 15.3% with balanced weights. This underscores the importance of calibrating reward functions to align with project-specific priorities.

5.2 Practical and Managerial Implications

The 15.3% reduction in project duration translates to approximately 4.2 months of schedule savings with associated overhead cost reductions and earlier revenue generation. The 22.1% improvement in resource utilization enables more efficient allocation across concurrent projects, while the 71.9% reduction in delay risk provides substantially greater confidence in meeting delivery commitments. The offline-training/online-deployment paradigm is well-suited to construction planning: the one-time 45-minute training investment yields instantaneous optimized schedules, and the trained Q-table can serve as a transfer-learning starting point for similar projects. However, the exponential growth of state space for projects exceeding 50 processes constitutes a practical limitation, motivating the adoption of Deep Q-Network (DQN) approaches in future applications.

Reference

1. Liu, J., Chen, J., Tian, J., & Li, Y. (2025). An Efficient Random Key–Based Genetic Algorithm for Solving Multimode Resource-Constrained Project Scheduling Problem. Journal of Construction Engineering and Management, https://doi.org/10.1061/JCEMD4.COENG-16722.
2. China Construction Industry Association. (2023). Analysis of Schedule Delay Factors in Large-Scale Construction Projects. Beijing: China Construction Industry Press.
3. Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction (Second Edition). MIT Press.
4. Hsu, S.-C. (2023). Reinforcement Learning in Construction Engineering and Management: A Review. Journal of Construction Engineering and Management, 148(11), https://doi.org/10.1061/(ASCE)CO.1943-7862.0002386.
5. Lin, C.-W. J., Lv, Q., Yu, D., & Chen, C.-H. (2022). Optimized scheduling of resource-constraints in projects for smart construction. Information Processing & Management, 59(5), 103005. https://doi.org/10.1016/j.ipm.2022.103005.