



# Monte Carlo Tree Search and Reinforcement Learning for Early-Stage Construction Scheduling

Aman Yadav

PG Student

Department of Civil Engineering  
MITS DU Gwalior

Prof. Gautam Bhadoriya

Assistant Professor

Department of Civil Engineering  
MITS DU Gwalior

**Abstract**— Early-stage construction scheduling plays a critical role in shaping project feasibility, delivery performance, and strategic decision-making. During the Front-End Loading (FEL) phases, schedules are highly aggregated and must be developed under significant uncertainty, limited design information, and evolving construction strategies. Traditional scheduling approaches are poorly suited to this context, as they assume fixed construction methods and rely on deterministic optimization techniques that cannot efficiently explore the vast number of feasible planning alternatives. This paper proposes a learning-based scheduling framework that combines Monte Carlo Tree Search (MCTS) and reinforcement learning to support early-stage construction scheduling decisions. The framework models scheduling as a sequential decision-making process in which aggregated work packages are progressively planned under alternative construction modes. MCTS enables efficient exploration of the scheduling decision space, while a neural network trained through reinforcement learning guides the search by estimating the value of partial schedules and prioritizing promising actions. The framework is evaluated using synthetically generated construction project instances designed to reflect realistic early-stage planning conditions. Experimental results demonstrate stable learning behavior, convergence toward high-performing scheduling strategies, and clear performance improvements over random scheduling baselines. While challenges remain in outperforming conventional heuristic strategies under certain conditions, the results confirm the feasibility of applying advanced artificial intelligence techniques to early-stage construction scheduling. The study contributes a novel methodological foundation for AI-driven decision support in construction scheduling under uncertainty.

**Keywords**— *Early-stage scheduling; Monte Carlo Tree Search; Reinforcement learning; Construction planning; Artificial intelligence; Modular construction.*

## I. INTRODUCTION

Construction scheduling is a foundational activity in project planning and control, directly influencing cost performance, resource utilization, and delivery reliability. In early project phases, scheduling serves not only as a coordination tool but also as a strategic instrument for evaluating feasibility, comparing alternatives, and supporting investment decisions. However, early-stage scheduling differs fundamentally from detailed execution scheduling due to the limited availability of design information, high uncertainty, and evolving construction strategies [1-5].

During the Front-End Loading (FEL) phases, schedules are typically developed at a high level of aggregation, focusing on major work packages rather than individual activities. These schedules support strategic decisions such as construction method selection, procurement strategy, and risk allocation [6]. Despite their importance, early-stage schedules are often produced using simplified heuristics or adapted versions of detailed scheduling techniques that are not well suited to the characteristics of early planning.

One of the primary limitations of traditional scheduling approaches in early-stage contexts is their assumption of a fixed construction method [7-10]. In practice, alternative construction strategies—such as modular, off-site, or hybrid approaches—may be under consideration. Each strategy fundamentally alters schedule structure, sequencing logic, and opportunities for parallel execution. Yet conventional early-stage schedules rarely capture these differences explicitly, limiting their usefulness for comparative decision-making. From a computational perspective, incorporating alternative construction methods into early-stage scheduling dramatically increases the size of the decision space. Deterministic optimization methods become impractical under these conditions [11-14]. Recent advances in artificial intelligence, particularly reinforcement learning and Monte Carlo Tree Search, offer new opportunities to address these challenges. This paper proposes a scheduling framework that integrates MCTS and reinforcement learning to support early-stage construction scheduling. The objective is to demonstrate that learning-based methods can effectively support early-stage scheduling decisions and provide insights into the planning implications of alternative construction strategies.

## II. BACKGROUND

Traditional construction scheduling methods such as the Critical Path Method and PERT dominate practice but are poorly suited to early-stage scheduling due to their reliance on detailed activity definitions and fixed construction strategies.

Resource-Constrained Project Scheduling Problems and multi-mode variants introduce greater flexibility but require detailed specification and suffer from combinatorial complexity, limiting early-stage applicability.

Artificial intelligence approaches, including genetic algorithms and other metaheuristics, have been applied to construction scheduling but typically operate as black-box optimizers and provide limited insight into sequential planning decisions [15-20].

Monte Carlo Tree Search provides an efficient mechanism for navigating large sequential decision spaces, while reinforcement learning enables policy learning without labeled datasets. Their integration has demonstrated success in complex planning domains.

The literature reveals a gap in learning-based scheduling frameworks tailored to early-stage construction planning. This study addresses that gap.

#### A. Research Methodology

This research adopts a computational and methodological approach to evaluate the feasibility of applying MCTS and reinforcement learning to early-stage construction scheduling. The study is designed as a proof-of-concept investigation focusing on algorithmic behavior and learning dynamics.

Scheduling is formulated as a sequential decision-making problem at an aggregated level consistent with Class 3 and Class 4 schedules [21-24]. Synthetic scheduling instances are used to enable controlled experimentation and reproducibility.

MCTS is employed to explore the scheduling decision space, while a neural network trained through reinforcement learning guides search and improves decision quality over successive iterations.

### III. PROPOSED LEARNING-BASED SCHEDULING FRAMEWORK

Naïve The framework treats early-stage scheduling as a sequential process in which decisions are made incrementally. Each state represents a partial schedule, and actions correspond to selecting work packages and construction modes.

MCTS builds a search tree to explore alternative scheduling sequences, while a neural network estimates state values and action probabilities to guide exploration. A reinforcement learning loop enables iterative improvement of scheduling policies based on simulated outcomes [25].

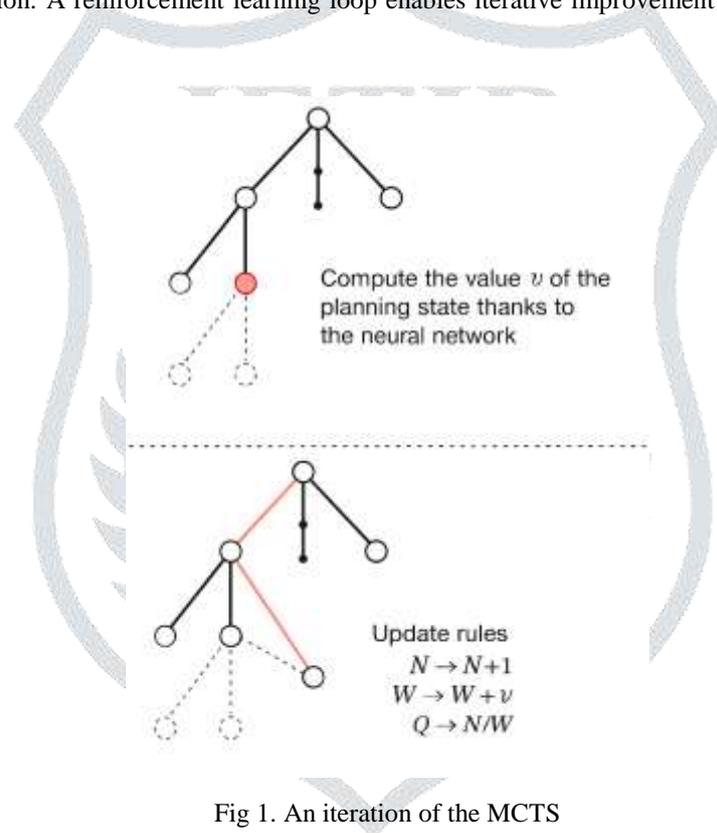


Fig 1. An iteration of the MCTS

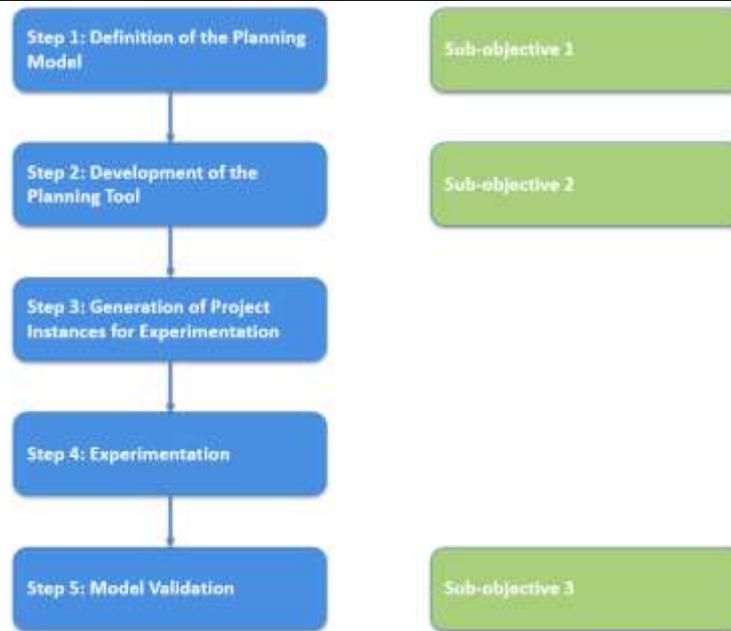


Figure 2. Steps in the research method

#### A. *Experimental Design and Evaluation*

Synthetic scheduling instances are generated to evaluate learning behavior, scalability, and robustness. Instance complexity is varied by adjusting the number of work packages and construction modes.

The training protocol involves repeated MCTS simulations and periodic neural network updates. Baseline strategies include random scheduling and heuristic approaches favoring conventional construction.

Performance is evaluated using aggregated schedule duration and convergence behavior.

## IV. RESULTS AND ANALYSIS

### 4.1 Discussion and Practical Implications

This section presents the results obtained from applying the proposed Monte Carlo Tree Search and reinforcement learning-based framework to the synthetic early-stage scheduling instances. The analysis focuses on learning behavior, convergence characteristics, sensitivity to algorithmic parameters, and comparative performance against baseline scheduling strategies. The intent is to evaluate whether the framework can reliably learn effective scheduling policies under the abstraction and uncertainty inherent to early-stage planning.

### 4.2 Learning Behavior and Convergence Characteristics

A primary objective of the experimental evaluation is to assess whether the proposed framework demonstrates consistent learning behavior across training iterations. Learning performance is evaluated by tracking aggregated schedule duration as a function of iteration count.

Across all tested instances, the framework exhibits a clear improvement in scheduling performance during the early training iterations. Initial iterations are characterized by relatively poor performance, reflecting the uninformed nature of the neural network and the reliance on exploratory MCTS simulations. As training progresses, performance improves steadily, indicating that the framework is learning to prioritize scheduling actions that lead to shorter and more coherent aggregated schedules.

After a sufficient number of iterations, performance gains begin to plateau, signaling convergence toward stable scheduling policies. This behavior is consistent with reinforcement learning applied to finite decision spaces, where incremental improvements diminish as the policy approaches a locally optimal strategy. Importantly, convergence is observed without oscillatory or unstable behavior, suggesting that the interaction between MCTS and the neural network is well balanced.

### 4.3 Effect of Monte Carlo Tree Search Simulation Depth

The number of MCTS simulations per decision step has a significant influence on scheduling quality. Increasing the simulation depth allows the search algorithm to explore the decision

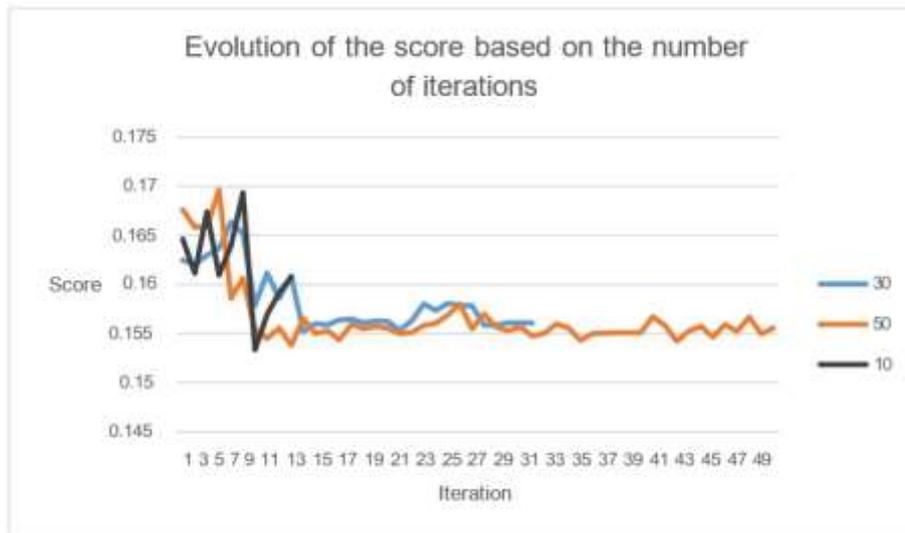


Figure 3 Evolution of the score as a function of the number of iterations

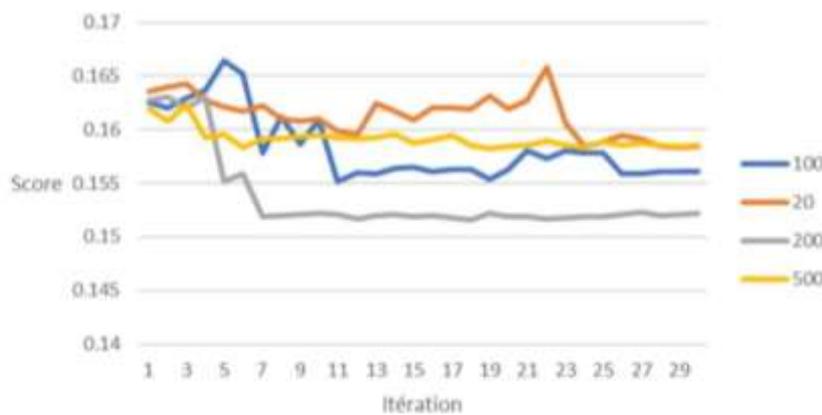


Figure 5 Evolution of the score as a function of the variation in numEps

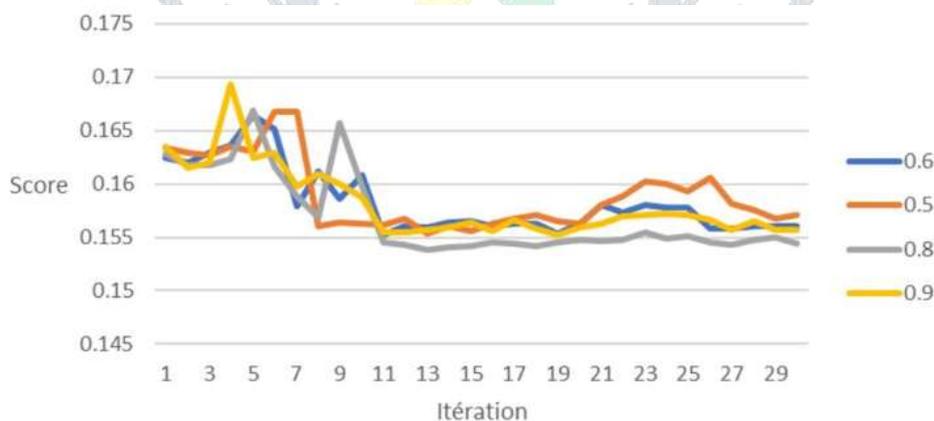


Figure 6 Evolution of the score according to the variation of the update threshold

Space more thoroughly, leading to more accurate estimates of action value.

Experimental results show that moderate increases in simulation depth yield substantial improvements in scheduling performance, particularly during early training iterations. However, beyond a certain threshold, additional simulations produce diminishing returns. While deeper search improves decision quality marginally, the associated computational cost increases disproportionately.

This finding highlights an important practical consideration: early-stage scheduling tools must balance decision quality with computational efficiency. The results indicate that near-optimal performance can be achieved with a limited number of simulations, making the framework viable for practical planning contexts where rapid scenario evaluation is required.

### 4.4 Impact of Training Dataset Size

The size of the training dataset generated during each iteration also affects learning behavior. Smaller datasets lead to slower convergence and higher variability in performance, as the neural network struggles to generalize from limited examples. In

contrast, larger datasets improve stability and accelerate learning by providing richer information about the scheduling environment.

However, similar to MCTS simulation depth, increasing dataset size beyond a certain point yields limited additional benefit. Excessively large datasets increase training time without proportionate performance gains. These results suggest that effective learning can be achieved with carefully selected dataset sizes that balance information richness and computational effort.

Table 1 Parameters for comparison with other models

Setting	Reference value
$\alpha = [ \dots ]$	[2,3,3]
<i>numIters</i>	30
<i>numEps</i>	200
<i>updateThreshold</i>	0.6
<i>numMCTSSims</i>	80
<i>MaxlenofQueue</i>	200,000
<i>arenaCompare</i>	200
F	$\sum a_i d_i + c_i$ Where $d_i$ and $c_i$ are the durations and costs of planned work packages in the schedule
Constant representing the cost of a day of work	800

#### 4.5 Comparative Performance Against Baseline Strategies

Comparative evaluation against baseline scheduling strategies provides critical context for interpreting the framework's effectiveness. The random scheduling baseline consistently produces the poorest performance, as expected, confirming that learning-based improvements are not attributable to chance.

When compared with the heuristic baseline reflecting conventional early-stage practice, results are more nuanced. In scenarios where alternative construction modes offer limited scheduling flexibility, heuristic strategies perform competitively, reflecting the embedded domain knowledge in conventional planning rules. However, in instances where alternative modes enable parallel execution or altered sequencing, the learning-based framework demonstrates superior performance.

These results indicate that the proposed framework is particularly valuable in contexts characterized by strategic uncertainty and multiple viable scheduling alternatives. Rather than replacing conventional heuristics, the framework complements them by revealing opportunities that are not captured by fixed decision rules.

#### 4.6 Robustness and Generalization

Robustness is evaluated by examining performance variability across multiple scheduling instances. The framework demonstrates relatively low variability after convergence, indicating that learned scheduling policies generalize across instances with similar characteristics. This generalization capability is essential for early-stage scheduling, where planners seek transferable insights rather than project-specific solutions.

Nevertheless, some variability remains, particularly in highly constrained instances where scheduling flexibility is limited. In such cases, learning-based approaches offer limited advantage over heuristics, underscoring the importance of contextual applicability.

### V. INTERPRETATION OF SCHEDULING POLICIES

Qualitative analysis of selected scheduling trajectories provides insight into the policies learned by the framework. The model tends to prioritize scheduling decisions that maximize early overlap between compatible work packages while deferring decisions that constrain future flexibility. This behavior aligns with established early-stage planning principles but is discovered autonomously through learning rather than encoded explicitly.

The emergence of such behavior reinforces the potential of learning-based scheduling frameworks to capture complex planning logic that may be difficult to formalize using deterministic rules.

## VI. CONCLUSION

This paper presented a learning-based framework for early-stage construction scheduling that integrates Monte Carlo Tree Search and reinforcement learning to address the uncertainty and abstraction characteristic of the Front-End Loading phases. By formulating scheduling as a sequential decision-making problem, the framework enables systematic exploration of alternative scheduling strategies without requiring exhaustive enumeration or detailed problem specification. Experimental results demonstrate stable learning behavior, convergence toward high-performing scheduling policies, and meaningful performance improvements over uninformed baseline strategies. While conventional heuristics remain competitive in constrained scenarios, the learning-based framework provides clear advantages in contexts characterized by multiple viable scheduling alternatives and strategic uncertainty. The primary contribution of this study lies in demonstrating the feasibility and value of applying advanced artificial intelligence techniques to early-stage construction scheduling. Rather than replacing expert judgment, the proposed framework enhances decision-making by revealing scheduling trade-offs and strategic opportunities that are difficult to identify using traditional methods. As construction projects continue to increase in complexity and as alternative construction strategies become more prevalent, the need for intelligent early-stage scheduling tools will grow. This research provides a methodological foundation for future development of AI-driven decision-support systems in construction planning and contributes to the broader advancement of construction technology and management.

## REFERENCES

1. AACE International. (2010). ACE International recommended practice No. 27R-03 Schedule classification system. :
2. Alcaraz, J., Maroto, C., & Ruiz, R. (2003). Solving the multi-mode resource-constrained project scheduling problem with genetic algorithms. *Journal of the Operational Research Society*, 54(6), 614-626.
3. Alpaydin, E. (2020). *Introduction to machine learning*: MIT press.
4. Bertram, N., Fuchs, S., Mischke, J., Palter, R., Strube, G., & Woetzel, J. (2019). *Modular construction: From projects to products*. McKinsey & Company: Capital Projects & Infrastructure, 1-34.
5. Bikitsha, L., & Haupt, T. (2011). Impact of prefabrication on construction site health and safety: Perceptions of designers and contractors.
6. Bouleimen, K., & Lecocq, H. (2003). A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version. *European journal of operational research*, 149(2), 268-281.
7. Burkov, A. (2019). *The hundred-page machine learning book (vol. 1)*: Andriy Burkov Canada. CCQ. (2017). *Annual Statistics*.
8. CEFRIO. (2011). Improving the efficiency and productivity of the construction sector through information technologies.
9. Cherkaoui, K. (2017). *Tactical planning of large engineering and construction projects*. (École Polytechnique de Montréal).
10. Choi, JO, Chen, XB, & Kim, TW (2019). Opportunities and challenges of modular methods in dense urban environments.
11. *International Journal of Construction Management*. doi:10.1080/15623599.2017.1382093
12. Cottrell, W.D. (1999). Simplified program evaluation and review technique (PERT). *Journal of Construction Engineering and Management*, 125(1), 16-22.
13. Coulom, R. (2006). Efficient selectivity and backup operators in Monte-Carlo tree search.
14. Paper presented at the International Conference on Computers and Games (pp. 72-83).
15. De Boer, R. (1998). *Resource-constrained multi-project management*. (PhD thesis, University of Twente, The Netherlands).
16. Elmaghraby, S.E. (1964). An algebra for the analysis of generalized activity networks.
17. *Management Science*, 10(3), 494-514.
18. Gans, D., Corbusier, L., & Plattus, A. (2006). *The Le Corbusier Guide*: Princeton Architectural Press.
19. Gibb, A.G. (1999). *Off-site manufacturing: prefabrication, pre-assembly and modularization*: John Wiley & Sons.
20. Gibson Jr, G., Kaczmarowski, J., & Lore Jr, H. (1995). Preproject-planning process for capital facilities. *Journal of construction engineering and management*, 121(3), 312-318.
21. Guo, J.-X., Hu, C.-M., & Bao, R. (2019). Predicting the duration of a general contracting industrial project based on the residual modified model. *KSCE Journal of Civil Engineering*, 23(8), 3275-3284.
22. Hammad, A.W.A., Akbarnezhad, A., Rey, D., & Grzybowska, H. (2017). Decision support system for modular construction scheduling.
23. Hwang, B.G., Shan, M., & Looi, K.Y. (2018). Key constraints and mitigation strategies for prefabricated prefinished volumetric construction. *Journal of Cleaner Production*. doi:10.1016/j.jclepro.2018.02.136
24. Leon, GP d. (2011). *Scheduling a Project at Different Levels*.
25. Lepelley, D., Paul, M., & Smaoui, H. (2013). *Introduction to game theory (1): non-games cooperatives*.
26. Li, L., Li, Z., Wu, G., & Li, X. (2018). Critical success factors for project planning and control in prefabrication housing production: A China study. *Sustainability (Switzerland)*. doi:10.3390/su1003083.