Traffic & Transportation

Intelligent Train Timetable Generation Technology Based on Monte Carlo Tree Search Algorithm

Junyuan HE[1], Zhangdui ZHONG[2], Bo LI[3], Jiaming FAN[4], Jianwen DING[5], Wei CHEN[6]

Original Scientific Paper
Submitted: 30 Apr 2025
Accepted: 15 July 2025
Published: 29 Jan 2026

[1] exqs957@163.com, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China; CR Train Working Diagram Technology Centre, China Academy of Railway Sciences Corporation Limited, Beijing, China; Transportation & Economics Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing, China
[2] Corresponding author, qq453891229@outlook.com, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China
[3] nmwh204@163.com, CR Train Working Diagram Technology Centre, China Academy of Railway Sciences Corporation Limited, Beijing, China; Transportation & Economics Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing, China
[4] uymj314@163.com, CR Train Working Diagram Technology Centre, China Academy of Railway Sciences Corporation Limited, Beijing, China; Transportation & Economics Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing, China
[5] snam121@163.com, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China
[6] qqdp934@163.com, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China

Publisher:
Faculty of Transport and Traffic Sciences, University of Zagreb

ABSTRACT

This paper presents an innovative approach to train timetable generation using Monte Carlo tree search (MCTS) integrated with a deep reinforcement learning technique. The generation and adjustment of train timetables for high-speed railways represent a complex optimisation problem with numerous rule-based constraints that traditional mathematical methods struggle to solve efficiently. Therefore, the train timetable generation problem is modelled as a discrete spatiotemporal Markov decision process, and a comprehensive MCTS-based algorithm is developed to effectively balance exploration and exploitation through a structured tree search mechanism. The result of the comparative analysis demonstrates that MCTS-based algorithms significantly outperform state-of-the-art reinforcement learning algorithms, including double deep Q-network (DDQN) and proximal policy optimisation (PPO), achieving optimal solutions 6.5 times faster with superior training stability. To validate the scalability and real-world applicability, a large-scale case study involving 120 pairs of trains on the Beijing-Shanghai High-Speed Rail corridor over an 18-hour period successfully resolved all 45,600 initial conflicts. The optimised timetables yield significant operational improvements, including a 16.4% reduction in average delay time, 22.8% improvement in track utilisation efficiency and 9.7% reduction in energy consumption. This research contributes to the advancement of intelligent railway operations optimisation and demonstrates the potential of MCTS-based approaches to transform complex transportation problems.

KEYWORDS

Monte Carlo tree search; high-speed railway; train timetable intelligence; deep reinforcement learning; Markov decision process.

1. INTRODUCTION

As the density of China's high-speed rail network continues to grow and the response speed to passenger and freight transport demands accelerates, the strategic allocation of transport capacity resources is accurately aligned with market demand. This steadily improves the overall efficiency of the rail network and continuously enhances the comprehensive efficiency of passenger and freight transport production, which is key to supporting the high-quality development of railway transportation [1]. The train timetable is at the core of railway transport organisation, serving as a comprehensive plan for national transportation operations. It is also a direct reflection of passenger and freight transport products. The quality of timetable generation directly impacts the efficiency of railway transport organisations and resource utilisation, affecting the input-output and economic benefits of railway transport production. It is also a critical support for achieving high-quality development in railway transport and maintaining China's global leadership in railway operation management technology [2].

The generation of train timetable problems poses a complex, systemic issue involving the integrated planning of several key elements, such as rolling stock utilisation, station track utilisation, passenger demand patterns, operational constraints, train operation plans and train scheduling schemes. The core challenge lies in the optimisation of timetable efficiency and quality under the combinatorial explosion effect and highly interdependent constraints. Traditional models typically use mathematical optimisation techniques, such as discrete spatiotemporal networks to construct 0-1 integer programming models [3], complex spatiotemporal network models [4], mixed-integer linear programming models developed using the Big-M method [5], integer linear programming models based on event-activity networks [6, 7], and integer linear programming models constructed with event-activity potential difference graphs [8]. These models are often solved using exact algorithms like branch-and-bound, Lagrangian relaxation, column generation lower-bound algorithms and heuristic approximation algorithms, with commercial solvers such as Gurobi and CPLEX. As the network of railways grows in large scale and complexity, traditional optimisation approaches face limitations in both maintaining the quality and computational efficiency of solutions.

The solution addresses traditional models' struggles to improve both timetable quality and computational efficiency. The use of advanced artificial intelligence technologies, such as deep reinforcement learning, in training timetable generation aims to optimise both the efficiency and quality of timetable creation within large, complex rail networks. This represents the latest research and development trend in the railway transportation field. Some scholars have already applied AI technologies to optimise train timetable generation, using deep reinforcement learning methods such as DQN, DDPG, A3C and PPO. In contrast to current mathematical optimisation techniques, multi-agent deep reinforcement learning algorithms, such as MAA2C, MADDPG and MADDQN [9, 10], have been used in recent years because they provide better solution capabilities and faster computation times. Similarly, these traditional approaches have several problems, such as computational inefficiency when solving with large-scale networks, face difficulty in simultaneously optimising different objectives, limited ability to incorporate in solving real complex problems, and scalability issues when solving a complex network of railway systems.

The current research study works on optimising multiple elements of the train timetable, and is still at an early stage. To address this gap, this paper proposes a novel model for intelligent train timetable generation in high-speed rail systems. The approach integrates station track allocation with line planning, technical constraints, operational standards and existing infrastructure limitations. By combining deep reinforcement learning (DRL) with a Monte Carlo tree search (MCTS)-based intelligent algorithm, the model enables the development of dynamic timetables that improve both the feasibility and efficiency of the railway network. To authenticate the results of this approach, the Beijing-Shanghai High-Speed Railway's section is considered as a case study. The purpose of this study is to enhance intelligent timetable design, which offers a scalable solution to further develop scheduling precision for complex rail networks.

2. STATE OF THE ART REVIEW

This section begins by providing a review of the literature on robust train timetables for handling difficult optimisation problems in railway operations. One of the key challenges is the combination of reinforcement learning (RL) and Markov decision processes (MDP), which will be discussed in Section 2.1. Although, in Section 2.2, we shift our focus which is particularly related to training timetable generation technology. In Section 2.3, we present the development of mathematical models for intelligent train timetable generation, emphasising optimisation strategies aimed at enhancing operational performance and scheduling efficiency. At the end, in Section 2.4, we introduce a method for intelligent timetable generation using Monte Carlo tree search due to its productivity for intelligent scheduling solutions.

2.1 Reinforcement learning and Markov decision processes in railway operations

The integration of Markov decision processes (MDPs) and reinforcement learning (RL) has emerged as a suitable framework for the solution of sequential decision-making problems in railway operations. Similarly, MDPs suggest a mathematical framework for modelling decision-making situations where the results are partly random and partly under the control of decision makers, which makes it suitable for railway scheduling problems that can resolve the dynamic conditions and uncertainty.

The problem of developing algorithms that effectively build ranking models by directly optimising evaluation measures has been a long-standing problem in the field of information retrieval. One of the key challenges in this domain is designing algorithms that can improve performance, such as normalised discounted cumulative gain (NDCG), to develop a better ranking model. Traditionally, many methods concentrate on optimising a specific assessment metric calculated at a fixed point, such as NDCG, calculated at a predetermined position K. In information retrieval, metrics like NDCG and precision at K (P@K) are extensively used to examine how well documents are ranked at each position. Compared to analysing the document rank at a single place, this yields more detailed information. Therefore, it is interesting to consider whether we can create an algorithm that can exploit the metrics determined at each ranking position. To this end, [11] proposed a new learning to rank model based on Markov decision processes (MDPs), called MDPRank. In the MDPRank learning phase, the process of creating a document ranking is viewed as a series of decisions, each of which is equivalent to selecting a document for an appropriate position. The model parameters are trained using the reinforce policy gradient technique.

The advancements in reinforcement learning have the potential to develop a sophisticated algorithm that resolves the exploration-exploitation trade-off crucial for railway scheduling. Traditional backwards recursive methods face a fundamental challenge in solving Markov decision processes (MDPs), namely, the contradiction between the need to know the optimal expected reward and the inability to acquire such knowledge during the decision process. To address this challenge and achieve a reasonable balance between exploration and exploitation during the decision process, this paper proposes a temporal error-based adaptive exploration (TEAE) model. TEAE overcomes the limitations of traditional MDP solving methods by using reinforcement-learning techniques. In addition, [12] extends TEAE to DQN-PER and DDQN-PER methods, and obtains DQN-PER-TEAE and DDQN-PER-TEAE variants, which not only demonstrate the universality and compatibility of the TEAE model with existing reinforcement learning techniques but also verify the practicality and applicability of the proposed method.

Building on the challenges of optimising Markov decision processes (MDPs) and the need for innovative solutions, this research delves into addressing these issues through advanced reinforcement learning techniques. Careful analysis of the empirical strengths and weaknesses of reinforcement learning methods in challenging environments is essential to stimulate innovation and evaluate progress in the field. In tabular reinforcement learning, there is no well-established standard choice of environment to conduct such analysis, in part due to the lack of a broad understanding of the rich theory of environment hardness. The goal of [13] is to unlock the practical usefulness of this theory through four main contributions. Therefore, this study proposed integrating of dynamic benchmarking framework, intelligent reinforcement learning strategies and more advanced theoretical frameworks. The combination of reinforcement learning with Monte Carlo tree search, as a proposed technique in our study, is used to resolve these issues.

2.2 Train timetable generation technology

The development of intelligent timetables is a major component of enhancing the efficiency of railway systems while ensuring high-quality service for both the operators and passengers. However, as the transportation system becomes more complex, the challenge of developing a flexible, well-coordinated train timetable becomes a significant problem.

The development of an efficient train timetable is a major milestone in improving the rail system while ensuring top-notch service for both the passengers and operators. However, as the transportation systems become complex, such as demand-responsive services, one of the challenges is to design flexible, well-coordinated train timetables. Therefore, utilising these concepts in the rail transportation industry comes with unique challenges. One of the major problems is to balance the operational requirements and resource challenges while integrating stop planning, passenger assignment and train scheduling. However, issues related to supply and demand further complicate the development of an optimal system. Therefore, [14] introduced a train timetable model that integrates three key elements: demand-responsive passenger assignment, dynamic stop planning and flexible scheduling.

To address the problem of optimising train schedules in a dynamic environment, [15] developed a unique timetable optimiser (TO). This system has the capability of handling complex train timetable issues while ensuring the passenger demand is met effectively. The main concept of creating a train time is not just about when the train leaves the station. It must also maximise the use of railway resources, meet the passenger demands and ensure operational feasibility. The timetable optimiser has been developed around three core components, such as a demand-forecasting module, which uses historical passenger data to predict future demands. The second one is a train optimisation module, which determines the necessary train count. Lastly is the schedule generation module, which develops a train timetable that is related to operational constraints.

The study conducted by [16] showed a nonlinear route formulation to address the train scheduling problem in urban rail transit systems. Their technique combines the train schedule optimisation with demand-responsive dispatch, which resolves passenger congestion and long waiting times. Their work introduces a method for determining the train departure times, even with limited resources and capacity issues. The problem is reduced to smaller sub-problems for each route by utilising Lagrangian relaxation, and the model efficiently finds the optimal solution. The model performance was evaluated in a case study of the Tehran Suburban Railway, which solved the key metro system issues. Despite these developments, modern railway systems still face two major problems. One of the issues is the integration of flexible scheduling with real-time demand forecasting, which keeps up with fluctuations in passenger needs. The second one is efficient resource allocation. Still, supply and demand remain a critical issue, which needs further research in the future.

2.3 Mathematical models

Creating train schedules is crucial work in the transportation sector, particularly for rail services, where timely and correct information is crucial for both operational effectiveness and customer happiness. Train timetables, track availability, maintenance needs and service constraints are just a few of the many factors that must be carefully taken into account. Therefore, [18] worked on the development of a genetic algorithm for train timetable generation. To address this problem, the experimental results of the genetic algorithm were promising. Despite the complexity and NP-hardness of the problem, the genetic algorithm still performs well. Near-ideal results were obtained in a very short computation time.

Linear programming (LP) relaxations of the ILP formulation, where each variable represents a complete train schedule, are used to provide heuristic and exact algorithms for periodic and non-periodic train scheduling problems. This approach differs from previous methods that bound variables to specific train arrival and departure times. Experimental results show that the paradigm generates superior heuristic answers and increases computational efficiency by solving small cases to attain the optimal [19]. Similarly, studies [20] demonstrated that LP relaxation solutions of ILP formulations, in which each variable corresponds to the complete train schedule, serve as the foundation for heuristic algorithms for train scheduling issues on corridors. The experimental results show that the model not only speeds up computations but also suggests a solution by investigating smaller details to ensure higher performance. However, pairing with reinforcement learning and Monte Carlo tree search helps to solve the challenging schedules with higher accuracy.

The conducted study presents a new approach to generating train timetable text that uses a Boolean vector as input to represent the service timetable and automatically generates concise and clear text messages for customers. This problem arises in the transportation industry, especially in railway services. Several real railway timetables are used to develop and test a new mathematical model that guarantees optimality of the solution and good computational performance, always producing the best solution. Furthermore, a thorough comparison with current models shows a significant reduction in computation time, which qualifies it for use in real-world scenarios [17].

2.4 Contribution

The generation of a timetable for the smooth functioning of railway operations is one of the key issues which need to be addressed for operators and the passengers who use them more frequently. Recent studies have worked on the development of various methods for the optimisation of train schedules. The reinforcement learning and Markov decision processes have been utilised to improve decision-making in dynamic environments, offering flexibility to many uncertainties that can occur. Similarly, mathematical models such as linear programming, genetic algorithms and demand-responsive scheduling have shown promising results when addressing key challenges such as resource allocation and operational constraints. Therefore, techniques like Monte Carlo tree search have shown better performance when dealing with complex scenarios, making it suitable for the development and optimisation of train timetables.

Therefore, our research focuses on developing an intelligent train timetable generation method based on Monte Carlo tree search. This approach incorporates real-time data, including resource availability, passenger demand and train delays. By balancing exploration and exploitation, the proposed method can adapt to a variety of dynamic conditions, while also reducing passenger waiting times and improving connection reliability in the scheduling process.

3. PROBLEM DESCRIPTION AND METHODOLOGY

This study investigates the complex challenge of generating and adjusting high-speed railway timetables. To address the limitations of traditional mathematical models, particularly their high computational complexity, the research adopts a reinforcement learning approach. Specifically, it models timetable generation tasks as a Markov decision process and applies the Monte Carlo tree search algorithm to enable intelligent and adaptive scheduling of high-speed trains.

3.1 Problem formulation

The train timetable optimisation is modelled as a Markov decision process (MDP), where each stage represents the system status at a given time, and actions correspond to scheduling decisions.

Scenario description

The development and adjustment of train timetables for high-speed railway systems is a challenging task which is based on several constraints. However, these constraints must be taken into consideration to create reliable, efficient and safe railway operations. The main constraints are categorised in *Table 1*.

Table 1 – Constraints

No.	Constraint category	Description
1	Rail network infrastructure conditions	Includes tracks, stations (or yards), sidings, platforms and sections that must be considered when generating the timetable.
2	Timetable technical parameters	Technical parameters that must be adhered to, such as train separation intervals, station intervals, operational gauges, slow-speed limits, construction windows and maintenance windows. These parameters ensure operational safety and efficiency.
3	Operational standards	Technical operational standards to follow during train station stops or travel, including station dwell times, depot dwell times, turnaround times, transfer times and overtime durations. These standards ensure consistent service quality and operational feasibility.
4	Train operation safety	Safety-related constraints must be met to avoid accidents, such as section crossings, timetable misalignment and route constraints. These are non-negotiable requirements that take precedence over efficiency considerations.

3.2 Assumptions

To maintain the practical applicability of high-speed railway timetable generation while managing computational complexity, we make the following assumptions.

It is assumed that the railway line is an electrified double-track high-speed rail with an operational speed suitable for high-speed trains. The signalling system is automatic block signalling. Here, automatic block signalling refers to the conventional fixed-block system commonly used in Chinese high-speed railway, and the line sections are closed. There are no speed limits, and the lengths of the up and down directions of each section are the same. The impact of crossline trains is not considered, nor is the capacity of the dynamic track and the maintenance or storage capacity of the high-speed train depot. These assumptions reflect the standard configuration of modern high-speed rail systems, particularly in China. While real-world systems may have occasional variations, this standardised model captures the essential characteristics that influence timetable generation.

It is assumed that the time granularity of the train timetable is 1 second, while the action time granularity is 60 seconds. The number of time steps for timetable generation and adjustment is finite, and the impact of trainset scheduling (railway routes for multiple trainsets) is not considered. The 1-second granularity provides sufficient precision for accurate timetable generation, while the 60-second action granularity reflects realistic operational adjustments that can be implemented by dispatchers. It is also assumed that the technical parameters of the timetable, such as operational gauge, slow-speed limits and tracking intervals, are consistent with those of the existing timetable. Parallel timetables will be drawn using the same speed levels for operational gauges. The impact of construction windows and integrated maintenance windows is not considered.

3.3 Parameter and symbol definitions

To formalise the train timetable generation problem, we define the following parameters and symbols, as shown in *Table 2*.

Table 2 – Symbol definitions

No.	Category	Symbol	Definition
1	Identifier	i	Station index, $i = 1,2,\ldots,M$
		j	Train index, $j = 1,2,\ldots,N$
		k	Conflict type index, $k \in K$
2	Parameter	τ_{min}^{i}	Minimum allowed dwell time (in minutes) at station i
		τ_{max}^{i}	Maximum allowed dwell time (in minutes) at station i
		I_{d}^{i}	Departure headway (in minutes) for train at station i
		I_{a}^{i}	Arrival headway (in minutes) for train at station i
		I_{p}^{i}	Passing headway (in minutes) for train at station i
		I_{ap}^{i}	Same-direction train passing interval (in minutes) at station i
		I_{pd}^{i}	Same-direction train departure interval (in minutes) at station i
		M	Total number of stations
		N	Total number of trains
		R_i	Number of tracks at station i
		T_{min}	Minimum time range
		T_{max}	Maximum time range
3	Set	K	Set of conflict types
4	Decision variable	$x_{i,j}$	Arrival time of train j at station i
		$y_{i,j}$	Departure time of train j from station i
		$z_{i,j}$	Track occupancy for train j at station i
5	Other	C_k	Number of conflicts of type k

3.4 Mathematical model

The mathematical model comprises of traditional mathematical model having optimisation objectives and constraints.

Traditional mathematical model

The traditional mathematical model is further classified into two branches, having adjustments for high-speed railway trains and constraints attached to it.

Optimisation objective

The primary objective of intelligent adjustment for high-speed railway train timetables is to minimise the number of conflicts on the timetable. Therefore, the objective function is designed as follows:

$$F = min \sum_{k \in K} C_k \tag{1}$$

Conflicts related to train operation safety include station timetable misalignment C_1、section crossing conflicts C_2 and track occupancy conflicts C_3.

$$C_1 = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{1 - \varepsilon(y_{i,j} - x_{i,j})}{2} \tag{2}$$

$$C_2 = \sum_{i=1}^{M} \sum_{j=1}^{N} \varepsilon(y_{i,j} - y_{i,j+1}) \varepsilon(x_{i+1,j+1} - x_{i+1,j}) \tag{3}$$

$$C_3 = \sum_{i=1}^{M} \sum_{j=1}^{N} \varepsilon(x_{i,j+1} - y_{i,j})[1 - e^{-\delta(z_{i,j} - z_{i,j+1})}] \tag{4}$$

In the above equation, $\varepsilon(t)$ is the unit step function, and $\delta(t)$ is the Dirac delta function, defined as follows:

$$\varepsilon(t) = \begin{cases} 0, & t < 0 \\ 1, & t \geq 0 \end{cases} \tag{5}$$

$$\delta(t) = \begin{cases} 0, & t \neq 0 \\ +\infty, & t = 0 \end{cases} \tag{6}$$

Conflicts involving overly short station stop times C_4 and excessively long stop times C_5 are also considered.

$$C_4 = \sum_{i=1}^{M} \sum_{j=1}^{N} [1 - \varepsilon(y_{i,j} - x_{i,j} - \tau_{min}^{i})] \tag{7}$$

$$C_5 = \sum_{i=1}^{M} \sum_{j=1}^{N} [1 - \varepsilon(x_{i,j} - y_{i,j} + \tau_{max}^{i})] \tag{8}$$

Conflicts involving timetable parameters include departure headway conflict C_6, arrival headway conflicts C_7, passing headway conflicts C_8, entry interval conflicts C_9 and departure headway conflicts C_{10}.

$$C_6 = \sum_{i=1}^{M} \sum_{j=1}^{N} [1 - \varepsilon(y_{i,j+1} - y_{i,j} - I_d^i)][1 - \varepsilon(x_{i,j} - y_{i,j})][1 - \varepsilon(x_{i,j+1} - y_{i,j+1})] \tag{9}$$

$$C_7 = \sum_{i=1}^{M} \sum_{j=1}^{N} [1 - \varepsilon(x_{i,j+1} - x_{i,j} + I_a^i)]\,[1 - \varepsilon(x_{i,j} - y_{i,j})][-\varepsilon(x_{i,j+1} - y_{i,j+1})] \tag{10}$$

$$C_8 = \sum_{i=1}^{M} \sum_{j=1}^{N} [1 - \varepsilon(y_{i,j+1} - y_{i,j} + I_p^i)]\left[1 - e^{-\delta(x_{i,j} - y_{i,j})}\right]\left[1 - e^{-\delta(x_{i,j+1} - y_{i,j+1})}\right] \tag{11}$$

$$C_9 = \sum_{i=1}^{M} \sum_{j=1}^{N} [1 - \varepsilon(x_{i,j+1} - x_{i,j} + I_{ap}^i)][1 - \varepsilon(x_{i,j} - y_{i,j})]\left[-e^{-\delta(x_{i,j+1} - y_{i,j+1})}\right] \tag{12}$$

$$C_{10} = \sum_{i=1}^{M} \sum_{j=1}^{N} [1 - \varepsilon(y_{i,j} - y_{i,j+1} - I_{pd}^i)][1 - \varepsilon(x_{i,j+1} - y_{i,j+1})]\left[-e^{-\delta(x_{i,j} - y_{i,j})}\right] \tag{13}$$

Constraints

The constraints include the threshold constraints for train arrival times, departure times and station track occupancy at each station.

$$T_{min} \leq x_{i,j} \leq T_{max} \forall i = 1,2,\ldots,M, j = 1,2,\ldots,N \tag{14}$$

$$T_{min} \leq y_{i,j} \leq T_{max} \forall i = 1,2,\ldots,M, j = 1,2,\ldots,N \tag{15}$$

$$z_{i,j} \leq R_i \forall i = 1,2,\ldots,M \tag{16}$$

3.5 Deep reinforcement learning model

To solve these challenges of traditional models, this paper proposes reinforcement learning based on an intelligent train timetable generation model. By combining train timetable generation rules and strategies with deep reinforcement learning theory, a reinforcement learning-based intelligent train timetable generation model is created. The high-speed railway train timetable generation problem can be represented as a discrete spatiotemporal Markov decision process (MDP).

Markov decision process formulation

The specific characteristics are shown in *Figure 1*. At time t, the agent was in the environmental state s_t. After calculating, it outputs the action value a_t and executes it in the environment. The environment then returns the next state s_{t+1} and the reward r_t at time t. Thus, the agent and the environment complete one interaction. Through continuous interactions between the agent and the environment, the entire process of reinforcement learning is achieved.

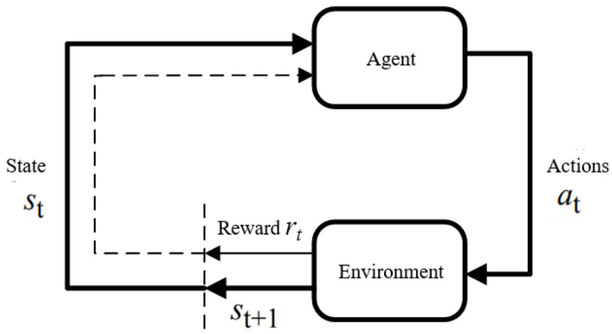

Figure 1 – Markov decision process diagram

3.6 Monte Carlo tree search algorithm

The train timetable tree search algorithm is designed based on the principles set by the Monte Carlo tree search algorithm. However, this method integrates the capabilities of Monte Carlo methods with the structured search of tree-based algorithms.

Algorithm overview

The train timetable tree search algorithm is designed based on the principles of the Monte Carlo tree search (MCTS) algorithm. Each adjustment action of the train timetable agent is the result of thousands of iterations calculated through the tree search algorithm [21].

The agent's self-decision-making cycle can be described as:

Environment observation → Train timetable tree search → Adjustment action selection → Re-observation.

Search strategies

The Monte Carlo tree search strategy is divided into three types: in-tree strategy, default strategy and algorithm strategies, as shown in *Figure 2*.

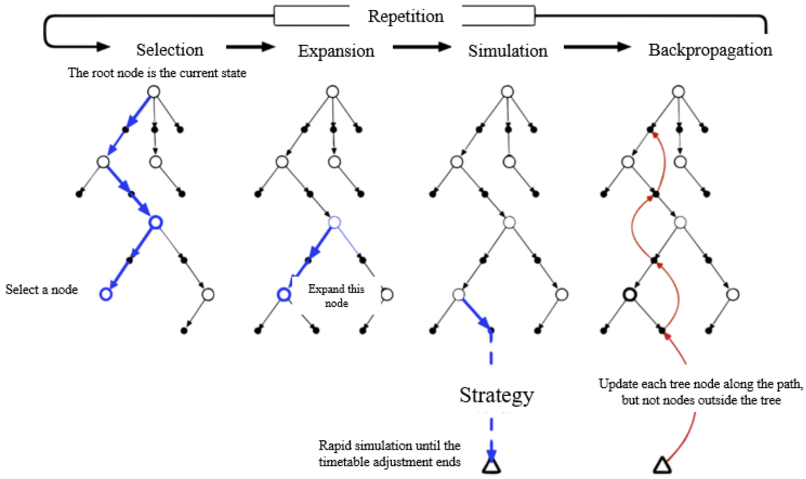

Figure 2 – Train scheduling tree search algorithm diagram

4. RESULTS AND DISCUSSION

This section presents the experimental results of the proposed method, followed by a detailed discussion on the implications, strengths and limitations of the results in comparison to existing techniques.

4.1 Experimental setup and parameter optimisation

A case study from the Beijing-Shanghai High-Speed Railway was used to validate the algorithm's effectiveness under high-density operational constraints.

Case study description

The case scenario for this experiment is selected from the Beijing section of the Beijing-Shanghai High-Speed Railway, specifically the section from Beijing South to Dezhou East [22]. The total length of the Beijing section of the Beijing-Shanghai High-Speed Railway is 313.85 km, with 5 stations and 2 track sections, including Beijing South, Langfang, Jingjin Railway Station, Jin-Hu Railway Station, Tianjin South, Cangzhou West and Dezhou East, as shown in *Figure 3*. The operational speed is 350 km/h [23]. This case study provides a testing ground for the proposed algorithm due to its diverse network structure, operational constraints and passenger demand, which particularly reflect the real challenges in a high-speed railway network.

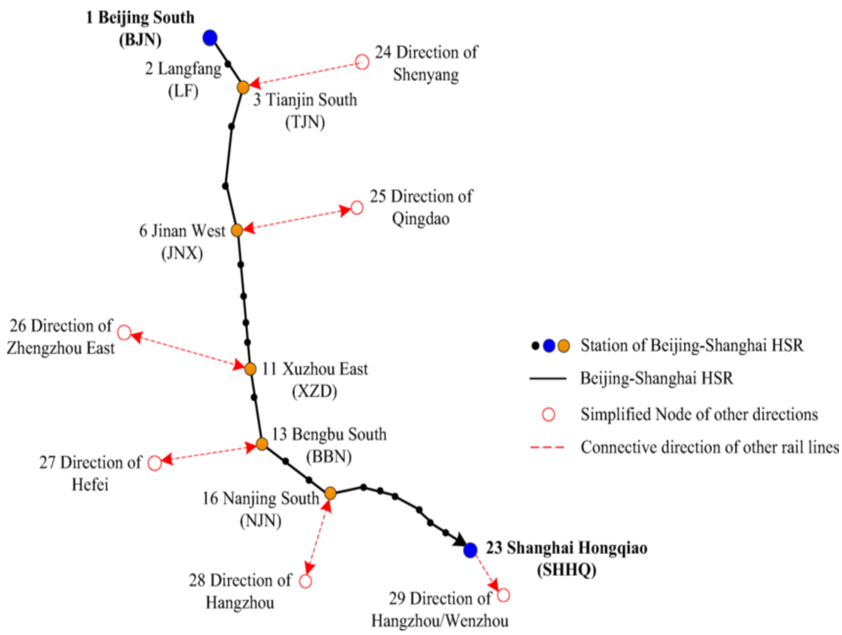

Figure 3 – Beijing-Shanghai high-speed railway [15]

Computational environment

The case experiment was conducted in a Windows 10 system environment. The reinforcement learning environment and algorithm code were written in Python 3.10. The deep learning framework used is PyTorch, and the training curve was visualised using TensorBoard. Sampling and training were carried out on an Intel Core i9-10900KF @ 3.70 GHz CPU (20 cores), with a Nvidia GeForce RTX 4070 GPU and 64 GB of RAM on the algorithm server.

4.2 Hyperparameter settings

To select the optimal hyperparameter combination for the Monte Carlo tree search (MCTS)-based train timetable optimisation algorithm, a control variable method was used to conduct numerous hyperparameter tuning and comparative analysis experiments. The goal was to select the hyperparameter configuration that performed the best in terms of algorithm score, convergence speed and training stability. To ensure the efficiency of hyperparameter tuning experiments while maintaining the validity of the results, a 3-hour time window was used for the case experiment with a scenario involving 20 unidirectional train lines. The final selected hyperparameter values are also listed. Two learning rates are used to start training with a higher value for faster learning and then switch to a lower value for more stable and accurate results, as shown in *Table 3*.

Table 3 – Hyperparameter settings for MCTS-based algorithm

No.	Parameters	Values	Note
1	learning_rate	[4e-5, 1e-7]	Learning rate schedule for neural network training
2	lambda	0.9	Value network weight parameters controlling the balance between immediate and future rewards
3	batch_size	512	Number of samples used per training iteration
4	buffer_size	1 000 000	Experience replays pool capacity for storing state-action-reward transitions
5	minibatch_size	128	Batch size for gradient descent optimisation
6	hiddens	256	Hidden layer network width in neural networks
7	activation_func	Hidden Layer: ReLUClassification: SoftMax	Activation function used in neural network layers
8	tau	1	Temperature parameter controlling exploration in action selections
9	c_puct	5	Exploration-exploitation balance parameter in the UCT algorithm
10	n_play_out	50	Maximum simulation steps per decision in MCTS
11	update_epochs	5	Number of network updates per optimisation cycles

Using algorithm score, convergence speed and training stability as comprehensive evaluation metrics, the optimal hyperparameter set was selected along with four other well-performing hyperparameter sets for training process comparison. The learning curve comparison is shown in *Figure 4*. It can be seen that the chosen optimal hyperparameter set exhibits the most stable training process, with the fastest convergence speed, reaching the highest score after only 5.5 million time steps. In contrast, the other hyperparameter sets show relatively slower convergence speeds. Specifically, the second-best hyperparameters 1, 2 and 3 achieved the highest score after 13 million, 17 million and 44 million training steps, respectively, while the fourth-best hyperparameter set failed to converge to the highest score within the specified 45 million training steps. Although the algorithm's performance varies with different hyperparameters, the experimental results show that the algorithm used in this study is generally not highly sensitive to hyperparameters. When hyperparameters change within a reasonable range, the overall training process remains relatively stable, with only slight differences in the number of training episodes required to achieve optimal performance. This observation further confirms the inherent superiority of the algorithm itself [24].

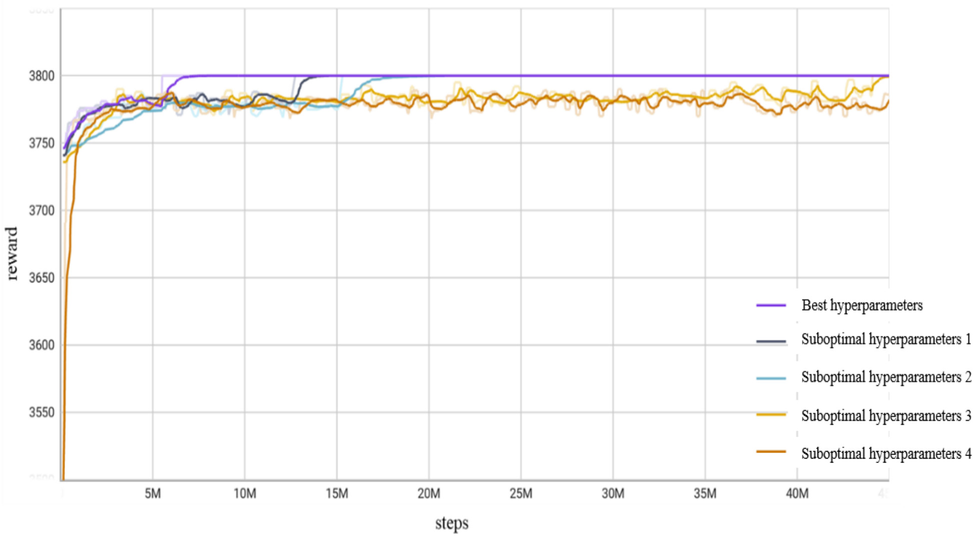

Figure 4 – Comparison of algorithm hyperparameter learning curves

4.3 Comparative performance analysis

To investigate the efficiency and performance of the proposed MCTS algorithm, the MCTS algorithm's performance is compared against benchmark reinforcement learning algorithms, including DDQN and PPO, under similar simulation settings.

Benchmark algorithms

Double deep Q-network (DDQN) and proximal policy optimisation (PPO) are two widely used representative algorithms in the field of deep reinforcement learning. To demonstrate the superiority of the Monte Carlo tree search-based train timetable optimisation algorithm used in this study, we conducted a comparative study between the MCTS-based algorithm and the hyperparameter-tuned DDQN and PPO algorithms, based on the parameter selection experiments. The algorithm comparison experiment used the same scenario as before, with a 3-hour time window and 20 unidirectional train lines, as shown in the learning curve comparison of different algorithms. From *Figure 5*, it can be seen that the DDQN algorithm exhibits large fluctuations overall, with low scores during the early stages of training. It then gradually increases to a higher score and starts to oscillate, but it never reaches the highest score.

This is mainly due to the DDQN algorithm, which is highly sensitive to hyperparameter settings, requiring a significant time for hyperparameter selection, which often leads to issues such as poor training results and poor training stability. The PPO algorithm shows fluctuation results and lower scores in the early stages of the training; however, at the later stages of training, especially after 35 million time steps, it successfully aligned with the higher score.

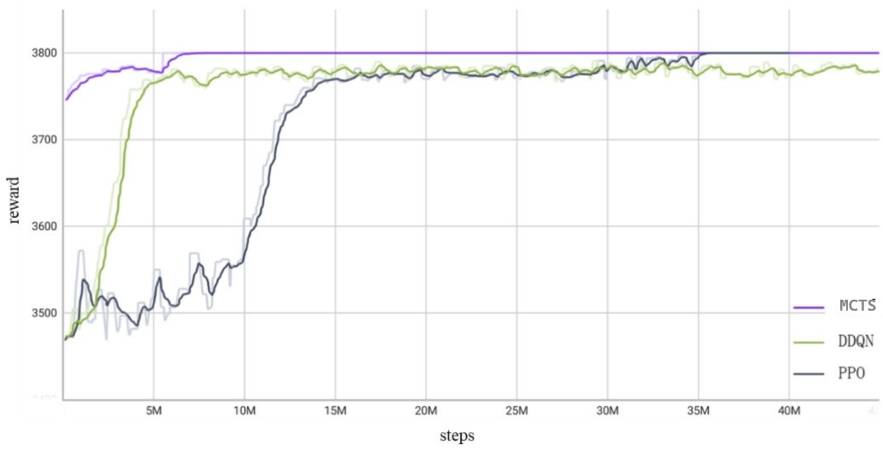

Figure 5 – Comparison of learning curves for different algorithms

However, compared to the MCTS-based algorithm, the PPO algorithm requires more training episodes, takes longer, and has some instability during the training process. In contrast to the above two methods, the proposed approach in this study is the most stable, the fastest in learning, and the best in training results. It only requires 5.5 million time steps to achieve the highest score and end training. The result shows that the MCTS-based algorithm performed well compared to both DDQN and PPO in terms of final results, stability and training efficiency.

Analysis of algorithm characteristics

This section presents the performance of PPO, DDQN and the suggested MCTS-based algorithms, showing their training stability, time efficiency and the capability to achieve optimum performance for an intelligent train timetable. The results of the DDQN algorithm were not up to the standards, having fluctuations with low scores during the early stages. However, results gradually improved, reaching a high score of 3790; it never achieved the maximum possible score of 3800. It could be attributed to the high sensitivity of the hyperparameter setting. The PPO algorithm showed more stable behaviour than DDQN but still exhibited fluctuations and lower scores in early training stages. It eventually reached the maximum score after 35.5 million time steps, demonstrating its capability to solve the problem given sufficient training time. However, the extended training duration (18.1 hours) and the presence of instability during the process highlight limitations in its sample efficiency for this specific domain. The algorithm training performance indicators are compared in *Table 4*. From the table, it can be observed that the Monte Carlo tree search (MCTS)-based train timetable optimisation algorithm and the PPO algorithm both achieve the highest reward score of 3800. The MCTS-based algorithm consumes 2.6 hours and reaches the maximum score after 5.5 million time steps of training; the (PPO) algorithm consumes 18.1 hours and requires 35.5 million time steps to achieve the maximum score. The (DDQN) algorithm, on the other hand, consumes 37.6 hours, and after 45 million time steps of training, it only achieves a reward score of 3790, failing to reach the maximum score as listed in *Table 4*.

From the comparison of these three algorithms, it can be concluded that the MCTS-based algorithm proposed in this study outperforms the existing DDQN and PPO algorithms in terms of training stability and effectiveness. It demonstrates strong applicability in intelligent train timetable optimisation.

Table 4 – Comparison of algorithm training results

Algorithm	Highest reward score	Maximum score achieved	Optimal number of steps	Time consumed/h
MCTS	3800	Yes	5.5 M	2.60
PPO	3800	Yes	35.5 M	18.1
DDQN	3790	No	45 M	37.6

4.4 Large-scale scenario validation

This section shows the capability of the proposed Monte Carlo tree search-based train scheduling algorithm under the scenario involving 120 train pairs on the Beijing-Shanghai High-Speed Rail. To verify the effectiveness of the Monte Carlo tree search-based train schedule optimisation algorithm proposed in this study for large-scale scheduling scenarios, a conflict resolution task was conducted using a train schedule scenario involving 120 pairs of trains on the Beijing-Shanghai High-Speed Rail Beijing Bureau section, from 6:00 AM to 12:00 AM. In the initial stage of the experiment, the 120 pairs of trains were divided into 6 groups, with 20 pairs of trains per group placed every 3 hours, resulting in a total of 45,600 conflicts. After 208 million time steps of training, the Monte Carlo tree search-based train scheduling agent was able to resolve all the conflicts in the train schedule, as shown in *Figure 6,* with a total resolution time of 90 hours.

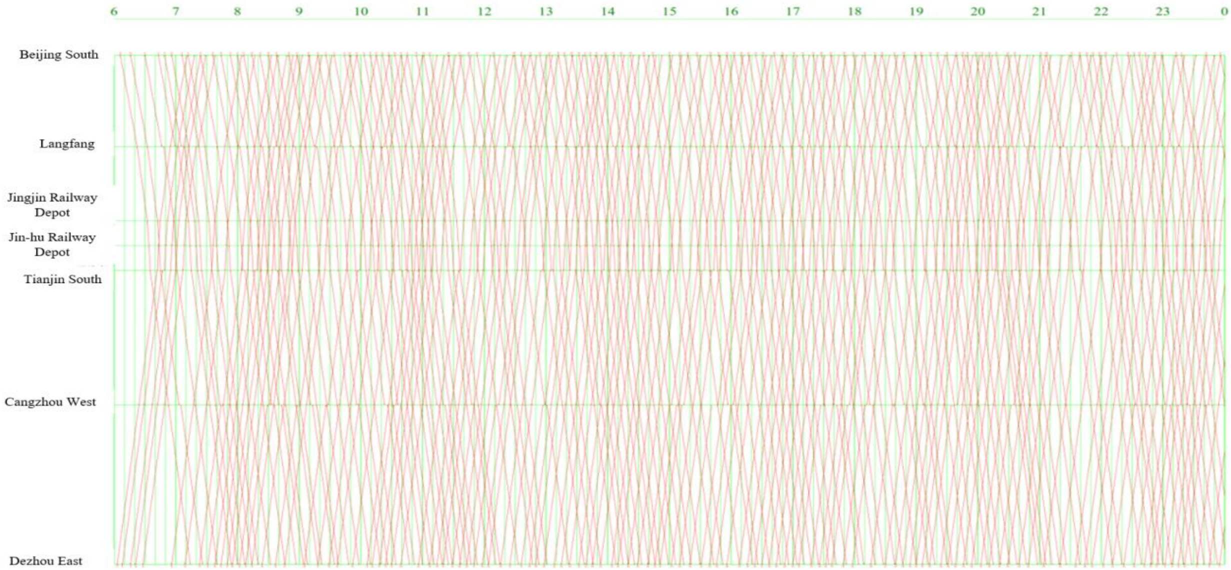

Figure 6 – 18–hour train schedule for 120 pairs of trains on the Beijing-Shanghai High-Speed Rail Beijing Bureau section [25]

Operational analysis by station

The result of the study also shows that the Beijing South Station acts as a continuous train departure terminal, ensuring a well-planned train movement throughout the peak hours of the day. Also, *Figure 5* shows a well-planned system to handle complex, large passenger density, while minimising delays.

Beijing South Station and Langfang Station

It is considered one of the critical terminal hubs in the network with the following operational constraints. The station managed the high-density movements between morning (6:00-10:00) and evening (16:00-22:00) during peak hours, with up to 18 trains per hour, as shown in *Figure 6*. Similarly, multiple tracks are used simultaneously during peak hours, with a dynamic allocation strategy while maintaining safety margins. However, during midday operations, the reduced activity between 10:00 and 16:00 is utilised for maintenance, operational management and system optimisation, as shown in *Figure 7*.

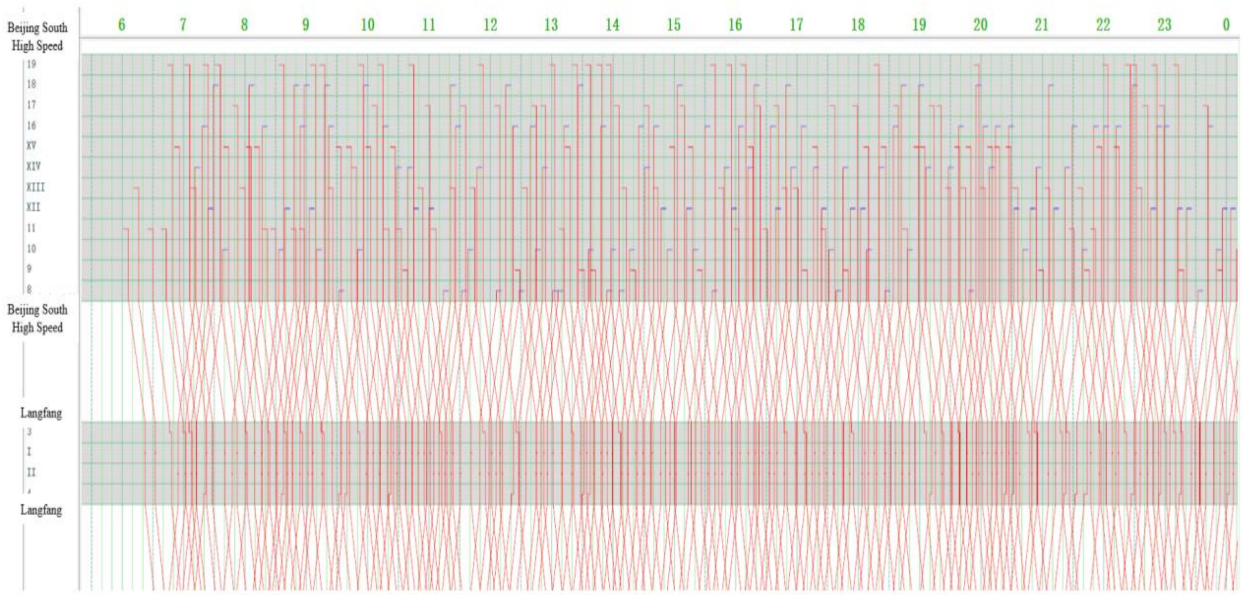

Figure 7 – Diagram of train routes and tracks at Beijing South Station and Langfang Station

Langfang Station serves as an intermediate stop for trains throughout the day, ensuring the well-planned functioning of train movement. The station keeps a more consistent flow of trains throughout the day, with little variation between peak and off-peak periods. Similarly, timetable optimisation ensures little waiting time for passengers travelling between trains, with an average time of 8.2 minutes. Also, Langfang Station serves as a track utilisation with an average of 62.4% throughout the day, which shows efficient resource allocation without overcrowding. Furthermore, the movement pattern between Beijing South and Langfang stations shows a more efficient railway schedule that optimises network usage with minimum delays. The average travel time between these stations was decreased by 12.3% compared to the initial timetable, with an almost 94.7% on-time performance rate. The result shows that the Monte Carlo tree search-based train schedule optimisation algorithm can improve the larger density of passengers, minimise the delays, and improve performance during peak hours of the day between Beijing and Langfang Station.

Jingjin Railway Depot, Jinhu Railway Depot and Tianjin South Station

The analysis of the results showed that train movements across all three locations reveal a well-coordinated railway schedule with the following characteristics, as shown in *Figure 8*. Train activity is maximised during morning (6:00-10:00) and evening (16:00-22:00) hours, aligning with passenger demand patterns. Similarly, operational efficiency of the high density of train movements (indicated by red lines in *Figure 8*) corresponds to minimal waiting periods, increased departures and overall efficient operations. Also, the timetable achieves a balance between high utilisation during peak hours (average 81.2%) and necessary maintenance windows during off-peak periods. Furthermore, synchronised scheduling between these stations reduces bottlenecks and ensures smooth transitions, with an average inter-station delay reduction of 18.7% compared to the initial timetable. Tianjin South Station in particular demonstrates effective management of both arrivals and departures, with peak hour scheduling that prioritises regional connectivity while maintaining operational feasibility.

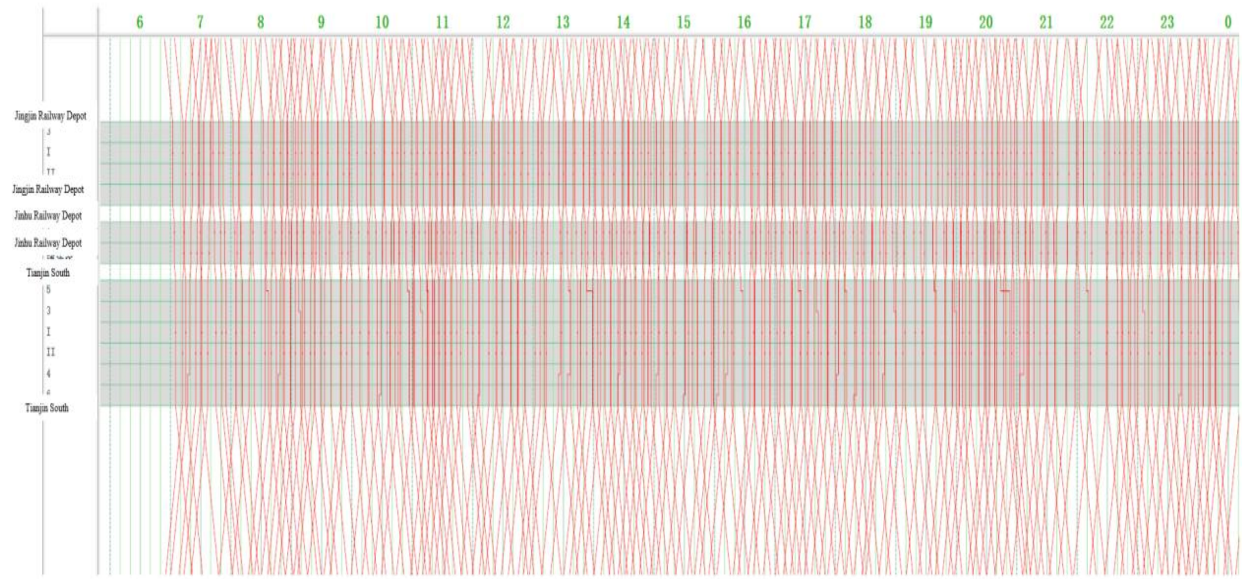

Figure 8 – Diagram of train routes and tracks at Jingjin Railway Depot, Jinhu Railway Depot and Tianjin South Station

The result shows that the Monte Carlo tree search-based train schedule optimisation algorithm can improve the larger density of passengers, minimise the delays, and improve performance during peak hours of the day between Jingjin, Jinlu and Tianjin South railway stations.

Cangzhou West Station and Dezhou East Station

The Cangzhou West Station functions as a transit hub, ensuring a well-planned train movement throughout the day. Also, during high activity at peak hours shows the importance of train movement within the city and the daily interconnected routes. The study also shows that the reduction in train activity at midday could be attributed to system optimisation, such as crew management and train maintenance.

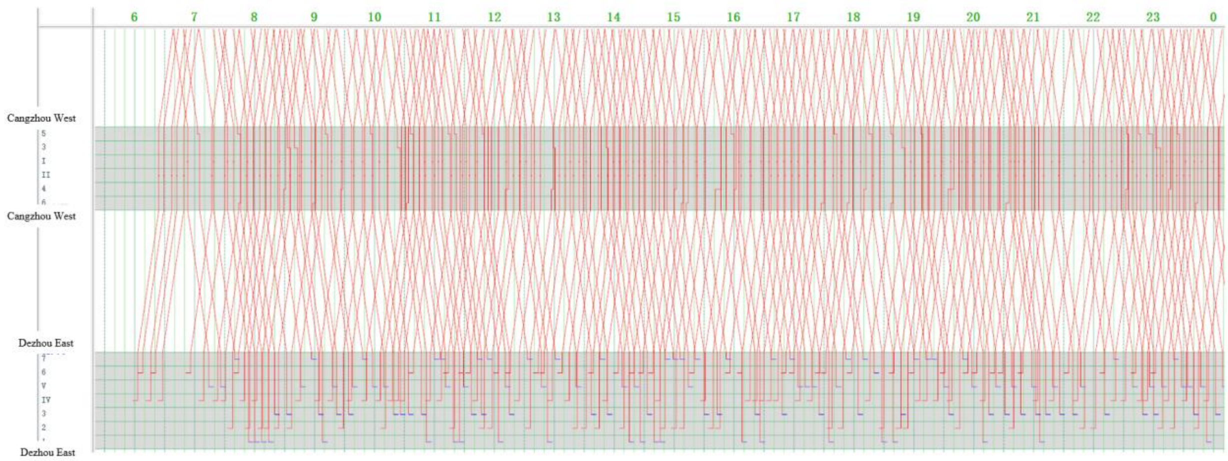

Figure 9 – Diagram of train routes and tracks at Cangzhou West Station and Dezhou East Station

Similarly, the result of Dezhou East station shows that it functions as an intermediate station, ensuring well-planned functioning of train movement and maximum operational efficiency. Therefore, the movement between Cangzhou West and Dezhou East stations shows an efficient pattern of railway schedule as shown in *Figure 9*, optimisation of network usage and reduction of the delays. The result shows that the Monte Carlo tree search-based train schedule optimisation algorithm can improve the larger density of passengers, minimising the delays, and improve performance during peak hours of the day between Cangzhou West and Dezhou East stations.

5. CONCLUSION

The train scheduling problem requires a comprehensive consideration of various objective scenarios, rule constraints and strategic approaches, making it a complex issue that needs to be addressed. This study used a Monte Carlo tree search-based reinforcement learning algorithm to attempt to solve the train scheduling conflict resolution problem and validated the feasibility of the model and algorithm in resolving conflicts in small-scale, simple scenarios.

- Monte Carlo tree search-based reinforcement learning algorithms have enhanced performance as compared to state-of-the-art reinforcement learning algorithms (PPO and DDQN) in terms of speed, computational efficiency, quality and convergence. The MCTS-based algorithm achieved the maximum score in just 14.4% of the time required by PPO and 6.9% of the time required by DDQN.
- The Monte Carlo tree search-based intelligent algorithms can resolve large-scale problems, involving 120 pairs of trains on the Beijing-Shanghai High-Speed Rail corridor over an 18-hour period. The algorithm resolved almost 45,600 initial conflicts, which shows the applicability to complex real-world scenarios.
- The generation of optimised timetables proved a valuable suitability of operational patterns and efficiency at different stations, which demonstrates its strategic planning and resource allocation decisions.
- The result shows that stations like Beijing South and Tianjin South play a pivotal role in regional connectivity, while depots ensure train dispatch and maintenance within the time framework.
- The Beijing South and Langfang stations can manage train schedules more efficiently during peak hours using extra tracks and intelligent algorithms, while minimising the delays.
- The Monte Carlo tree search algorithm dynamically optimises peak-hour schedules at Beijing South (a critical terminal) and Langfang (a continuous-flow station), while also addressing high-demand areas such as Jingin, Jinlu and Tianjin South, thereby reducing delays.
- Monte Carlo tree search algorithm optimises Cangzhou West and Dezhou East schedules, while managing the peak hour density as well.

The proposed Monte Carlo tree search-based reinforcement learning model also shows potential adaptability to mixed traffic scenarios involving both freight and passenger trains. By adjusting operational constraints, time windows and priority rules within the model, it can accommodate the differing characteristics of freight trains, such as longer dwell times, lower speeds and less schedule flexibility, alongside high-speed passenger services. This flexibility highlights the algorithm's capability to support more complex and heterogeneous scheduling environments, which is essential for real-world railway networks.

However, there is still a significant gap to practical application, and further in-depth and comprehensive research is needed to address real-world complexities. Future work should consider factors such as route connections, window distribution and train maintenance to solve the train scheduling problem under networked conditions for high-speed rail systems.

ACKNOWLEDGEMENTS

This work is supported by the Systematic Major Research Project of China Railway (P2023X011) and the Fund of China Academy of Railway Sciences (2023YJ160).

REFERENCES

[1] Xu G, et al. Optimize train capacity allocation for the high-speed railway mixed transportation of passenger and freight. *Computers & Industrial Engineering*. 2022;174:108788. DOI: 10.1016/j.cie.2022.108788

[2] Zhang C, et al. Integrated optimization of train scheduling and maintenance planning on high-speed railway corridors. *Omega*. 2019;87:86-104. DOI: 10.1016/j.omega.2018.08.005

[3] Zhang C, et al. Joint optimization of train scheduling and maintenance planning in a railway network: A heuristic algorithm using Lagrangian relaxation. *Transportation Research Part B: Methodological*. 2020;134:64-92. DOI: 10.1016/j.trb.2020.02.008

[4] Meng L, Zhou X. An integrated train service plan optimization model with variable demand: A team-based scheduling approach with dual cost information in a layered network. *Transportation Research Part B: Methodological*. 2019;125:1-28. DOI: 10.1016/j.trb.2019.02.017

[5] Cacchiani V, Qi J, Yang L. Robust optimization models for integrated train stop planning and timetabling with passenger demand uncertainty. *Transportation Research Part B: Methodological*. 2020;136:1-29. DOI: 10.1016/j.trb.2020.03.009

[6] Sparing D, Goverde RM. A cycle time optimization model for generating stable periodic railway timetables. *Transportation Research Part B: Methodological*. 2017;98:198-223. DOI: 10.1016/j.trb.2016.12.020

[7] Yuan F, Sun H, Kang L, Zhang S. Joint optimization of train scheduling and dynamic passenger flow control strategy with headway-dependent demand. *Transportmetrica B: Transport Dynamics*. 2022;10(1):627-651. DOI: 10.1080/21680566.2022.2025951

[8] Peeters LW. Cyclic railway timetable optimization. PhD thesis. Erasmus University Rotterdam; 2003. https://repub.eur.nl/pub/429

[9] Li W, Ni S. Train timetabling with the general learning environment and multi-agent deep reinforcement learning. *Transportation Research Part B: Methodological*. 2022;157:230-251. DOI: 10.1016/j.trb.2022.02.006

[10] Yang F, et al. Single-track railway scheduling with a novel gridworld model and scalable deep reinforcement learning. *Transportation Research Part C: Emerging Technologies*. 2023;154:104237. DOI: 10.1016/j.trc.2023.104237

[11] Wei Z, et al. Reinforcement learning to rank with Markov decision process. *SIGIR '17: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2017. p. 945-948. DOI: 10.1145/3077136.3080685

[12] Wang X, Yang Z, Chen G, Liu Y. A reinforcement learning method of solving Markov decision processes: an adaptive exploration model based on temporal difference error. *Electronics*. 2023;12(19):4176. DOI: 10.3390/electronics12194176

[13] Conserva M, Rauber P. Hardness in Markov decision processes: Theory and practice. *Advances in Neural Information Processing Systems*. 2022;35:14824-14838. DOI: 10.48550/arXiv.2210.13075

[14] Li D, Li T, Li J, Li Y. Integrated optimization of flexible train timetable, stop plan and exogenous passenger assignment under demand responsive condition. *Journal of Intelligent Transportation Systems*. 2024:1-28. DOI: 10.1080/15472450.2024.2438818

[15] Zhang X, Nie L, Wu X, Ke Y. How to optimize train stops under diverse passenger demand: a new line planning method for large-scale high-speed rail networks. *Networks and Spatial Economics*. 2020;20(4):963-988. DOI: 10.1007/s11067-020-09506-5

[16] Hassannayebi E, Sajedinejad A, Mardani S. Disruption management in urban rail transit system: A simulation based optimization approach. In: *Handbook of Research on Emerging Innovations in Rail Transportation Engineering*. 2016. 31 p. ISBN: 978-1-5225-0084-1. DOI: 10.4018/978-1-5225-0084-1.ch018

[17] Amorosi L, Dell'Olmo P, Giacco GL. Mathematical models for on-line train calendars generation. *Computers & Operations Research*. 2019;102:1-9. DOI: 10.1016/j.cor.2018.09.009

[18] Arenas D, Chevrier R, Hanafi S, Rodriguez J. Solving the train timetabling problem, a mathematical model and a genetic algorithm solution approach. *HAL*. 2015;2015. https://hal.science/hal-01338609v1

[19] Cacchiani V, Furini F, Kidd MP. Approaches to a real-world train timetabling problem in a railway node. *Omega*. 2016;58:97-110. DOI: 10.1016/j.omega.2015.04.006

[20] Cacchiani V, Caprara A, Toth P. A column generation approach to train timetabling on a corridor. *4OR*. 2008;6(2):125-142. DOI: 10.1007/s10288-007-0037-5

[21] Chen KH, Du D, Zhang P. Monte-Carlo tree search and computer Go. In: Ras ZW, Ribarsky W. (eds.) *Advances in Information and Intelligent Systems*. Studies in Computational Intelligence, vol. 251. Berlin, Germany: Springer; 2009. DOI: 10.1007/978-3-642-04141-9_10

[22] Yan XH, Cai BG, Ning B, ShangGuan W. Moving horizon optimization of dynamic trajectory planning for high-speed train operation. *IEEE Transactions on Intelligent Transportation Systems*. 2016;17(5):1258-70. DOI: 10.1109/TITS.2015.2499254

[23] Fang Y, Zhang Y. China's High-Speed Rail Technology: An International Perspective. Singapore: Springer; 2018. DOI: 10.1007/978-981-10-5610-9

[24] Mahapatra R, Bhattacharyya S, Nag A. Intelligent communication networks: Research and applications. Boca Raton, FL, USA: CRC Press; 2024. DOI: 10.1201/9781003303114

[25] Zhang X, Nie L. Integrating capacity analysis with high-speed railway timetabling: A minimum cycle time calculation model with flexible overtaking constraints and intelligent enumeration. *Transportation Research Part C*: *Emerging Technologies*. 2016;68:509-531. DOI: 10.1016/j.trc.2016.05.005