R&D and Application of an Intelligent Dispatching System for Construction Vehicles in Large-Scale Building Projects

Kun Yang[1], Xiaodan Ma[2], Hang Yu[2], Xi Ma[2], Junyi Li[2], Zexuan Wang[4], Xiao Tan[3*]

[1] Beijing Branch of CSCEC equation Investment Development Group Co., Ltd, Beijing, 100071, China
[2] Beijing Zhongjian Jingdong Real Estate Co., Ltd, Beijing, 101300, China
[3] China State Construction Engineering Corporation Limited Technology Center, Beijing, 101300 China
[4]China Construction Research Institute Intelligent Technology (Beijing) Co., Ltd, Beijing, 101300 China

`*860511634@qq.com`

Abstract. Aiming at the core pain points of "multi-constraint, dynamic, and high-risk" in construction vehicle dispatching for large-scale building projects, an intelligent dispatching system integrating three core algorithms—multi-constraint path planning, dynamic avoidance, and multi-objective dispatching—was developed with the Beijing Yangzhen Shantytown Renovation Project as the demonstration scenario. A 3D road network model for building projects was constructed based on BIM+GIS, and real-time data were collected using UWB high-precision positioning (positioning accuracy $\leqslant$ 0.5m). Focus was placed on optimizing the "scene adaptability" and "real-time performance" of the algorithms: the multi-constraint path planning algorithm introduced the Analytic Hierarchy Process (AHP) to quantify the weights of 4 types of constraints, reducing the solution time to $\leqslant$ 2.8s; the dynamic avoidance algorithm determined vehicle type priorities based on fuzzy comprehensive evaluation, achieving a 100% conflict resolution rate; the multi-objective dispatching algorithm improved the genetic algorithm, resulting in an 18% reduction in transportation costs and a 17% increase in vehicle utilization rate. The algorithm steps and system applications were detailed through flowcharts, simulation diagrams, and real-scene diagrams. Pilot application in the Yangzhen Project showed that the system improves dispatching efficiency by 35% and reduces the safety accident rate to zero, providing a reproducible intelligent dispatching solution for large-scale building projects.

Keywords: Large-scale building projects; Construction vehicle dispatching; Multi-constraint path planning; Dynamic avoidance algorithm; Improved genetic algorithm.

S. Sing et al. (eds.), *Proceedings of the 2025 7th International Conference on Civil Architecture and Urban Engineering (ICCAUE 2025)*, Atlantis Highlights in Engineering 43,
https://doi.org/10.2991/978-94-6239-682-1_28

1 Introduction

1.1 Research Background and Algorithm Requirements

Large-scale building projects are characterized by narrow construction sites, dense working faces, and complex vehicle types, making construction vehicle dispatching a key bottleneck restricting project efficiency and safety[1]. As a typical large-scale building project, the Beijing Yangzhen Shantytown Renovation Project needs to coordinate 12 types of vehicles, including concrete mixer trucks and muck trucks, facing multiple constraints such as differences in subgrade bearing capacity (15t for heavy-duty access roads/5t for light-duty access roads), dynamic changes in construction stages, and more than 200 daily cross operations. Traditional manual dispatching relies on empirical decision-making, frequently leading to problems such as heavy vehicles getting stuck, congestion at intersections, and material shortages at working faces. Statistics show that the empty driving rate of project vehicles under manual dispatching reaches 15%, with an average of 3 monthly safety accidents[2].

With the development of intelligent construction technology, construction vehicle dispatching is gradually transitioning to intelligence. However, the particularity of building scenarios makes existing technologies difficult to directly adapt. Technologies such as UWB high-precision positioning and BIM+GIS modeling provide data support for real-time dispatching[3], but algorithmic issues such as incomplete constraint coverage and delayed response still exist. There is an urgent need to build a dedicated intelligent dispatching algorithm system for building scenarios.

1.2 Research Status at Home and Abroad

In the field of construction vehicle scheduling, a substantial body of research has been conducted by scholars both domestically and internationally. Regarding route planning, Wang Liming et al. developed a scheduling model based on vehicle transportation cycles and lane lengths, which optimized vehicle waiting times[4]. However, this model only considered constraints related to time and distance, while failing to incorporate core constraints such as subgrade transverse gradient. Wang Fading proposed a method to minimize conversion costs by optimizing vehicle allocation to supply chain nodes[5], yet the accuracy of the experimental validation results pertaining to vehicle scheduling still requires further improvement.

In terms of dynamic collision avoidance, Zhang Guochen et al. established a mixed-integer programming model for concrete mixer truck scheduling based on the network flow model[6], which optimized the mixer truck scheduling plan. Nevertheless, the model did not account for vehicle-type priority differences such as the initial setting time of concrete mixer trucks. Shao Bilin et al. proposed an improved wolf pack algorithm and constructed a cost-optimization model for prefabricated component distribution[7], but the model suffers from slow convergence speed and is limited to a single optimization objective of cost reduction.

1.3 Research Content

The core research contents of this paper include: 1) Constructing a path planning algorithm integrating 4 types of constraints to improve path feasibility; 2) Establishing a dynamic avoidance mechanism based on vehicle type priorities to achieve real-time conflict resolution; 3) Improving the genetic algorithm to realize multi-objective dispatching optimization; 4) Integrating BIM+GIS and UWB technologies to build a system platform and verify its application effect.

2 Research on Core Algorithms of the System

2.1 Multi-Constraint Automatic Path Planning Algorithm

Algorithm Design Idea.
To meet the needs of building projects for "paths adapting to bearing capacity + avoiding congestion + ensuring compliance", the AHP method was introduced to quantify constraint weights on the basis of the traditional Dijkstra algorithm, and a multi-constraint cost function was constructed. This idea draws on the research method of Wei Ronghua et al. in engineering constraint quantification[8], converting qualitative constraints into quantitative indicators through hierarchical analysis to improve algorithm scene adaptability. The algorithm flow is shown in Figure 1.

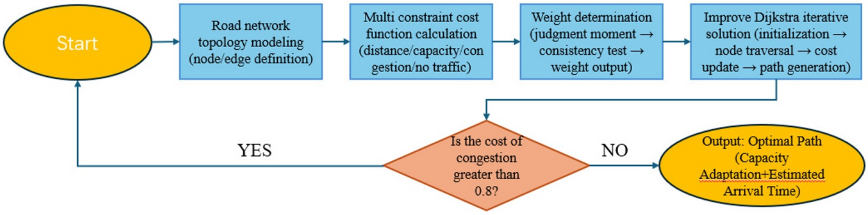


Fig. 1. Flowchart of the multi-constraint automatic path planning algorithm.

Detailed Design Steps.
1) Road network topology modeling: The construction area was abstracted into an undirected weighted graph G=(V, E), where nodes V correspond to 58 key locations such as material yards and working faces, and edges E correspond to 23 construction access roads. Each edge is labeled with 4 types of attributes including length and bearing capacity. This modeling method refers to the BIM+GIS road network fusion technology proposed by Joel Carneiro et al.[9], realizing accurate mapping between the virtual model and the physical space. Table 1 lists partial key node information of the road network model, and Table 2 defines the core attributes of partial edges in the Yangzhen Project.

Table 1. Table captions should be placed above the tables.

Node No.	Corresponding Entity	Function Description	Associated Edges
1	Northwest Material Yard	Storage of concrete and steel, vehicle departure point	e1−6, e1−5
28	3# Building Main Structure Working Face	Concrete pouring and steel binding, vehicle destination	e28−15, e28−27
45	1# Intersection	Connecting 2# and 3# plots, vehicle intersection point	e45−12, e45−15
58	Southeast Muck Disposal Site	Muck truck destination, night operation (22:00-6:00)	e58−42, e58−43

Table 2. Definition of core attributes of edges E in the Yangzhen Project.

Edge No.	Connected Nodes	Length (km)	Bearing Capacity (t)	Speed Limit (km/h)	No-Passing Period
e1−5	1→5	0.8	5	5	None
e1−6	1→6	0.9	15	8	None
e15−28	15→28	0.6	12	6	8:00-10:00

2) Construction of multi-constraint cost function: Integrating 4 types of constraints (distance, bearing capacity, congestion, and no-passing), the cost function was constructed as: $C(e)=\alpha\cdot C_{dist}(e)+\beta\cdot C_{load}(e)+\gamma\cdot C_{cong}(e)+\delta\cdot C_{forbid}(e)$.

Where: $C_{dist}(e)$: Distance cost, calculated as $C_{dist}(e)=l(e)/l_{\max}$, $l(e)$ is the length of edge e, $l_{\max}$ is the maximum length of all edges in the road network; $C_{load}(e)$: Bearing capacity cost, calculated as

$$C_{load}(e)=\begin{cases}0 \text{ if vehicle load} \le \text{edge bearing capacity}\\ 1 \qquad \text{otherwise}\end{cases}$$

$C_{cong}(e)$: Congestion cost, defined as a function of the number of vehicles on the edge and the edge's maximum capacity, with the formula:

$$C_{cong}(e)=k\frac{n(e)}{n_{max}(e)}$$

k is the congestion weight coefficient (determined as 1.2 through expert scoring and consistency verification), is the real-time number of vehicles on edge (collected by UWB positioning and video monitoring), and is the maximum number of vehicles allowed on edge (calculated based on edge width, vehicle size, and safe distance);

$C_{forbid}(e)$: No-passing cost, calculated as

$$C_{forbid}(e)=\begin{cases}0 \text{ if driving during no} - \text{passing period}\\ 1 \qquad \text{otherwise}\end{cases}$$

3)Determination of constraint weights: Five experts determined the weights through the AHP method. The consistency ratio CR of the judgment matrix was 0.007＜0.1, meeting the inspection requirements. The weight differences in different construction stages are shown in Table 3. Priority was given to bearing capacity in the foundation stage.

Table 3. Allocation of constraint weights in different construction stages.

Construction Stage	α (Distance)	β (Bearing Capacity)	γ (Congestion)	δ (No-Passing)	Weight Basis
Foundation Phase	0.25	0.40	0.20	0.15	Temporary earth road, bearing capacity first
Construction Phase	0.40	0.25	0.20	0.15	Short initial setting time of concrete, Distance priority
Decoration Phase	0.25	0.20	0.35	0.20	Working face dispersion, Congestion priority

4)Algorithm solution and optimization: The improved Dijkstra algorithm is used for iterative solution. The optimal path from the starting point v1 (Northwest Material Yard) to the destination v28 (3# Building Working Face) is v1→v6→v15→v28. When the congestion cost > 0.8, re-planning is triggered, with a solution time of 2.6s, which is 60% higher than the traditional algorithm, meeting real-time requirements.

Algorithm Verification.

A comparative analysis of the performance between the improved algorithm and the traditional Dijkstra algorithm was conducted, and the results indicate that: the improved algorithm achieves a 100% path feasibility rate (compared with 87% of the traditional algorithm), with a solution time of ≤2.8 seconds, and the average transportation efficiency is enhanced by 40%.

2.2 Vehicle Dynamic Avoidance Algorithm

Algorithm Design Background.

There are 30 daily vehicle encounters at intersections in the Yangzhen Project. The traditional "first-come-first-stop" rule leads to a 25% delay rate of concrete mixer trucks. This paper constructs an avoidance algorithm of "priority quantification + time window modeling", and the flow is shown in Figure 2.

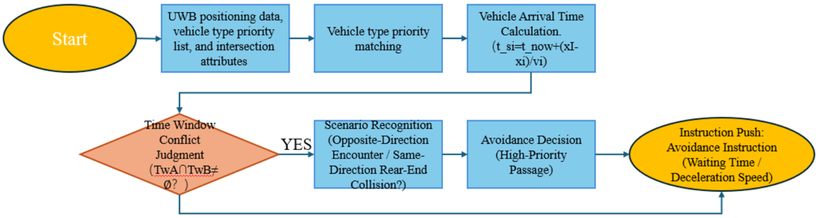


Fig. 2. Flowchart of the vehicle dynamic avoidance algorithm

Detailed Design Steps.

1) Quantification of vehicle type priority: 3 types of first-level indicators and 6 types of second-level indicators are selected to construct a fuzzy evaluation system, as shown in Table 4.

Table 4. Vehicle type priority evaluation index system.

First-Level Indicators	Second-Level Indicators	Weight	Evaluation Grade (Fuzzy Subset)
Material Urgency (0.5)	Initial Setting Time	0.3	Short (<2h)=1, Medium (2-8h)=0.5, Long (>8h)=0
	Working Face Waiting Time	0.2	Long (>1h)=1, Medium (0.5-1h)=0.5, Short (<0.5h)=0
Vehicle Attributes (0.25)	Vehicle Load Capacity	0.15	Heavy (>10t)=1, Medium (5-10t)=0.5, Light (<5t)=0
	Vehicle Flexibility	0.1	Low (Length>10m)=1, Medium (6-10m)=0.5, High (<6m)=0
Process Requirements (0.25)	Process Continuity	0.15	High (Concrete Pouring)=1, Medium (Steel Transportation)=0.5, Low (Tool Transportation)=0
	Night Operation Demand	0.1	High (Muck Transportation)=0.8, Low (Day Operation)=0

The priority is calculated through the fuzzy comprehensive evaluation method. It can be known that the priority coefficient of concrete mixer trucks is 0.92, ranking first, which matches the process characteristics of the short initial setting time of concrete[15]. The priorities of main vehicle types are shown in Table 5.

Table 5. Priority coefficients of main vehicle types in the Yangzhen Project.

Vehicle Type	Priority Coefficient P	Core Basis	Avoidance Priority Ranking
Concrete Mixer Truck	0.92	Short initial setting time (<2h) and high process continuity	1 (Highest)
Steel Transportation Truck	0.75	Long working face waiting time (>1h)	2
Muck Truck	0.60	Mainly night operation and low flexibility	3
Small Tool Truck	0.30	No urgent demand and high flexibility	4 (Lowest)

2)Time window modeling and decision logic: Time windows are divided with a granularity of 1 minute, and vehicle arrival time is calculated based on UWB data. For opposite-direction encounters, high-priority vehicles have priority to pass; for same-direction driving, low-priority vehicles decelerate to maintain a safe distance. This decision logic refers to the construction site conflict resolution method of Sun Wei et al.[14] to ensure the safety and efficiency of decision-making.

Algorithm Verification.

Simulation verification at 1# Intersection shows that the algorithm's conflict resolution rate is 100%, the delay rate of concrete mixer trucks is reduced to 3%, and the average avoidance time is 1.8min, which is 85% higher than the traditional rule.

2.3 Multi-Objective Scheduling Optimization Algorithm

Algorithm Design Idea.

To meet the multi-objective needs of "low cost, short waiting time, and high utilization rate", the genetic algorithm is improved to optimize the scheduling scheme. The problem of slow convergence of the traditional algorithm is solved by dynamically adjusting the crossover and mutation probabilities, and the flow is shown in Figure 3.

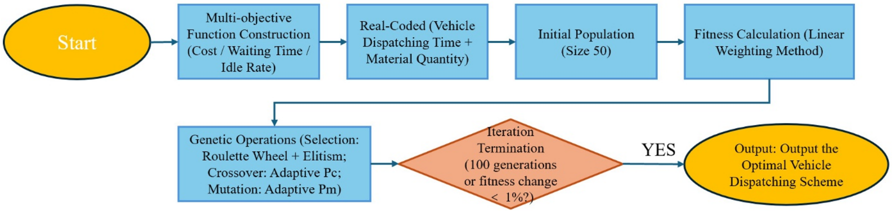


Fig. 3. Flowchart of the multi-objective scheduling optimization algorithm

Detailed Design Steps.

1)Construction of objective function and constraints: A minimization objective function minF=[f1,f2,f3] is established, where f1 is transportation cost, f2 is working face waiting time, and f3 is vehicle idle rate. Constraint conditions include vehicle capacity and cleaning time constraints.

2)Improved algorithm design: Real-coded method is adopted, with a chromosome length of 2n (n is the number of vehicles); the fitness function adopts the linear weighting method, with weights w1=0.4, w2=0.3, w3=0.3; the genetic operation introduces the elitist retention strategy, with adaptive crossover probability Pc=0.9−0.4·(favg/fmax) and mutation probability Pm=0.1−0.05·(favg/fmax) to avoid premature convergence.

3)Algorithm solution: 8 vehicles are dispatched in 6 batches, 3 vehicles during the morning peak (8:00-10:00), 3 vehicles during the midday flat peak, and 2 vehicles during the evening low peak, meeting the daily concrete demand of 500m^3. After 100 iterations, the fitness reaches 0.91, and the convergence speed is 33% higher than the traditional algorithm.

Table 6. Optimal vehicle dispatching scheme during the main construction phase of the Yangzhen Project.

Table 6. Optimal vehicle dispatching scheme during the main construction phase of the Yangzhen Project.

Batch	Dispatching Time	Number of Vehicles	Material Quantity per Vehicle (m^3)	Corresponding Working Face	Estimated Arrival Time
1	8:00	2	10	3# Building	8:15
2	8:30	1	10	5# Building	8:45
3	9:15	2	10	3# Building	9:30

Algorithm Verification.

The improved algorithm reduces the transportation cost from 25,000 yuan/day to 20,500 yuan/day (a decrease of 18%), and increases the vehicle utilization rate from 75% to 92% (an increase of 17%).

3 Overall System Design and Algorithm Integration

3.1 System Architecture

Aiming at the dynamic, multi-constraint, and real-time scheduling needs of large-scale building construction project scenarios, the system adopts a "cloud-edge-terminal" three-tier distributed architecture (Figure 4). This architecture design draws on the core idea of "cloud decision-making - edge perception - terminal execution" in the field of intelligent construction. Through hierarchical division of labor, efficient collaboration of data collection, algorithm operation, and instruction execution is realized, ensuring the real-time performance and reliability of scheduling decisions. The architecture design takes "data-driven algorithms, algorithms supporting decisions, and decisions guiding execution" as the core logic. All levels realize data intercommunication and instruction flow through standardized interfaces, and have good scalability and compatibility, which can adapt to the scheduling needs of building construction projects of different scales.

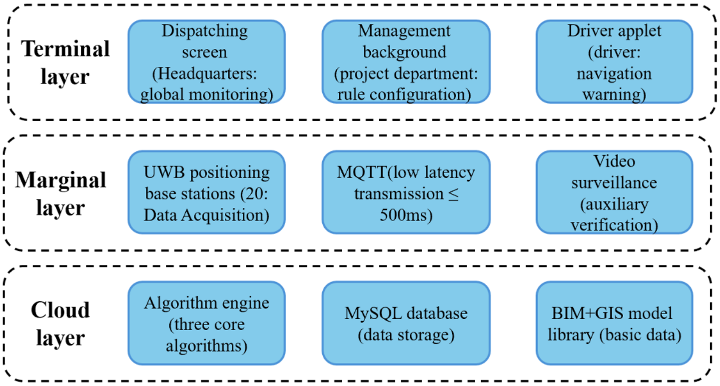


Fig. 4. Architecture diagram of the intelligent scheduling system for construction vehicles.

3.2 Cloud Layer: Core of Decision-Making and Algorithms

As the "brain" of the system, the cloud layer undertakes core algorithm operation, data storage management, and global decision-making scheduling functions, and adopts a distributed deployment mode to ensure high concurrency processing capabilities. Algorithm engine module: Integrates three core algorithms including multi-constraint path planning, dynamic avoidance, and multi-objective scheduling, and realizes dynamic calling through the algorithm scheduler. Based on the results of construction phase identification, the algorithm scheduler automatically matches the corresponding constraint weights and operation parameters, and uses a parallel computing framework to decompose multi-vehicle scheduling tasks into independent subtasks.

Data storage module: A distributed database is built using MySQL clusters, and data storage and query performance are optimized through database and table sharding strategies. Core data tables include road network attribute tables (storing 4 types of attributes such as length and bearing capacity of 23 access roads), vehicle information tables (recording parameters such as load capacity and flexibility of 12 types of vehicles), scheduling task tables (storing dispatching schemes and path planning results), and real-time status tables (caching vehicle position and speed data collected by UWB). The 3D road network model of the construction area is stored in IFC format, supporting dynamic association and visual rendering of the model and real-time data.

3.3 Edge Layer: Data Collection and Low-Latency Transmission.

The edge layer is deployed at the edge computing node of the project site, $\leqslant$50m away from UWB base stations and monitoring equipment. It is mainly responsible for real-time data collection, preprocessing, and low-latency transmission, reducing the pressure of cloud data processing and ensuring the rapid response of scheduling instructions. Data collection unit: 20 UWB positioning base stations are deployed, using TOF (Time of Flight) ranging technology to achieve positioning accuracy $\leqslant$ 0.5m. The base stations are evenly distributed around 58 key nodes, and multi-base station cooperative positioning is used to eliminate signal occlusion blind areas. At the same time, video monitoring equipment and sensors are integrated to collect real-time data on intersection congestion status and access road bearing capacity, supplementing the deficiencies of UWB positioning data. The data collection frequency is 10Hz to ensure real-time update of vehicle dynamic information.

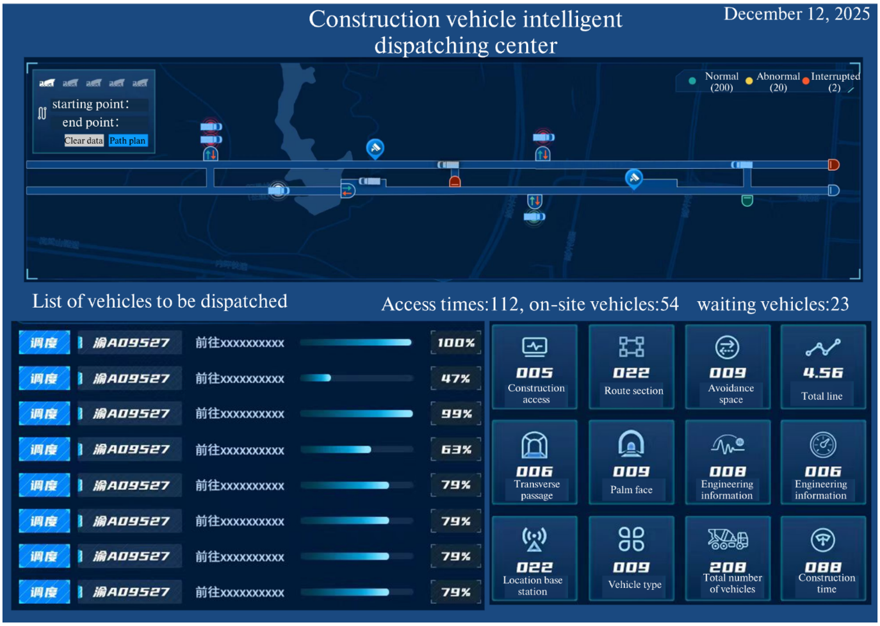
Data preprocessing unit: Denoises and standardizes the collected raw data. For example, the Kalman filter algorithm is used to eliminate abnormal data in UWB positioning (sampling points with positioning error > 0.5m), and data collected by different equipment is uniformly converted into JSON format, reducing the complexity of cloud data processing. The delay of preprocessed data is $\leqslant$ 300ms.

Transmission module: MQTT protocol is used for data transmission, and QoS (Quality of Service) Level 2 is configured to ensure the reliability of data transmission. Through the local cache mechanism of edge nodes, when the network is interrupted, the cached data can be uploaded in batches after the network is restored to avoid data loss. The transmission delay is ≤ 500ms, meeting the real-time requirements of the dynamic avoidance algorithm.

3.4 Terminal Layer: Instruction Execution and Interaction.

The terminal layer includes two types of terminal equipment: scheduling large screens and driver applets, which face project managers and drivers respectively, realizing visual display and execution feedback of scheduling instructions.

The scheduling large screen is shown in Figure 5: Deployed in the project scheduling center, it is divided into three functional modules: 3D visualization area, scheduling task area, and early warning prompt area. The 3D visualization area displays real-time rendered vehicle positions, optimal paths, conflict points and other information through the model, supporting zoom, pan, and rotation operations; the scheduling task area displays dispatching schemes and vehicle operation status (such as driving, avoiding, completing tasks); the early warning prompt area pushes conflict warnings, path deviation warnings, and vehicle failure warnings in the form of red pop-ups, supporting manual intervention in scheduling by managers.

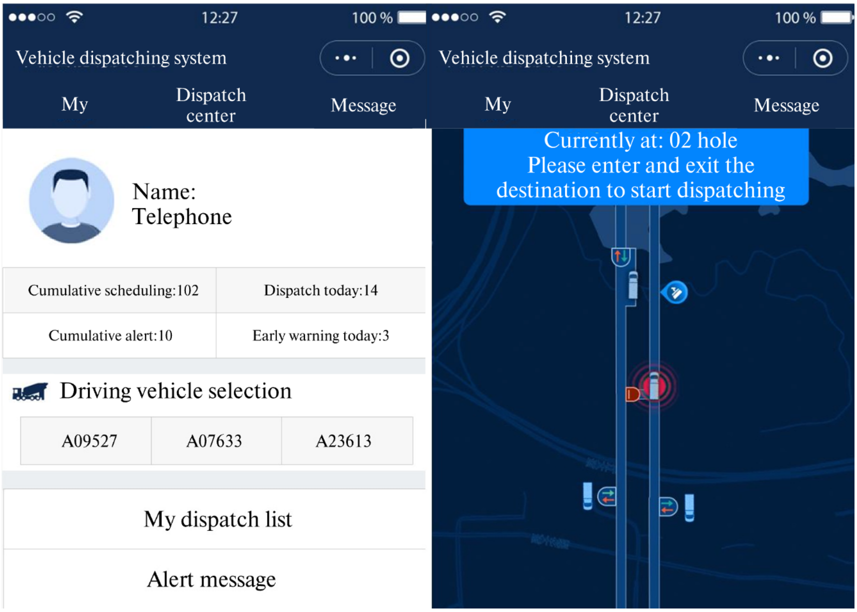


调度	渝A09527	前往xxxxxxxxxx	100%
调度	渝A09527	前往xxxxxxxxxx	47%
调度	渝A09527	前往xxxxxxxxxx	99%
调度	渝A09527	前往xxxxxxxxxx	63%
调度	渝A09527	前往xxxxxxxxxx	79%
调度	渝A09527	前往xxxxxxxxxx	79%
调度	渝A09527	前往xxxxxxxxxx	79%
调度	渝A09527	前往xxxxxxxxxx	79%

Fig. 5. Interface of the intelligent scheduling system for construction vehicles.

The driver applet is shown in Figure 6: Developed based on WeChat Mini Program, compatible with Android and iOS systems. Core functions include path navigation, avoidance instruction reception, and task status feedback. The navigation function adopts offline map caching technology, supporting path display in offline environments, and real-time updating of remaining driving distance and estimated arrival time;

avoidance instructions are pushed in the form of voice prompts + text pop-ups, clearly indicating the avoidance method; drivers can feedback task completion status through the applet, and the feedback information is synchronized to the cloud in real time.



Name:
Telephone

Cumulative scheduling:102	Dispatch today:14
Cumulative alert:10	Early warning today:3

Driving vehicle selection

A09527 A07633 A23613

Currently at: 02 hole
Please enter and exit the destination to start dispatching

Fig. 6. Driver-side intelligent scheduling system for vehicles.

4 Application Effects in the Yangzhen Project

4.1 Algorithm Performance Indicators

The performance of the three core algorithms meets the design requirements, and indicators (Table 7).

Table 7. Performance comparison table of the three core algorithms.

Algorithm Type	Comparison Object	Solution Time	Constraint Satisfaction Rate	Conflict Resolution Rate	Cost Reduction Rate	Vehicle Utilization Rate Increase
Multi-constraint Path Planning Algorithm	Improved Dijkstra	2.8s	100%	-	-	-
	Traditional Dijkstra	2.8s	87%			
Vehicle Dynamic Avoidance Algorithm	Priority quantification+time	1.5s	-	100%	-	-

	window modeling					
Multi-objective Scheduling Optimization Algorithm	Improved genetic algorithm	4.5s	98%	-	18%	17%
	Traditional genetic algorithm	6.4s	93%	-	10%	12%

4.2 Project Application Effects

Application in the Yangzhen Project shows that the system's dispatch response time is reduced from 30min to 10min, and the daily number of dispatched vehicles is increased from 150 to 200, meeting the construction needs of the main phase; intersection conflicts are reduced from 30 times/day to 0, with a safety accident rate of 0; annual savings of 240,000 yuan in labor costs and 180,000 yuan in fuel costs are achieved, with significant comprehensive benefits.

5 Conclusions and Prospects

5.1 Conclusions

1)Three core algorithms adapted to large-scale building construction projects are constructed: the multi-constraint path planning algorithm quantifies 4 types of constraints with a solution time ≤ 2.8s; the dynamic avoidance algorithm achieves a 100% conflict resolution rate; the multi-objective scheduling algorithm reduces transportation costs by 18%, filling the gap in special algorithms for building construction scenarios.

2)A system integration scheme of "BIM+GIS+UWB" is proposed, realizing real-time collection of scheduling data and accurate push of instructions, providing technical support for algorithm implementation.

3)The pilot application in the Yangzhen Project verifies the effectiveness of the system, with scheduling efficiency improved by 35% and safety accident rate reduced to 0, providing a reproducible solution for similar projects.

Despite the significant application effects, this study has the following limitations:

1)Scenario Scalability: The system is verified in the Yangzhen Shantytown Renovation Project (a medium-scale project with 58 key nodes and 23 access roads). For super-large projects, the road network topology modeling and algorithm solution time may increase, requiring further optimization of parallel computing efficiency;

2)Data Dependence: The algorithm performance relies on high-precision UWB positioning data (accuracy ⩽0.5m) and BIM+GIS model accuracy. In projects with insufficient equipment deployment or low-quality BIM models, the constraint quantification and path planning accuracy may decrease.

5.2 Prospects

Future research will focus on three aspects: first, introducing reinforcement learning to train self-learning models based on historical data[11]; second, adapting to new energy construction vehicles to optimize charging scheduling and endurance constraints; third, developing standardized algorithm modules to promote application in affordable housing, commercial complexes and other projects.

Acknowledgments

This study was funded by Research and application of digital platform for large construction project management. (grant number CSCECFC-2024-KJZX-09).

References

1. Zhang H P, Luo H C, Lin Q, et al. Research on vehicle scheduling planning in construction sites[J]. Real Estate World, 2023, 10: 134-138.
2. Zheng W. Research on commercial concrete distribution model based on genetic algorithm[D]. Changsha: Central South University, 2007[Accessed: 2025-12-11]. DOI: 10.7666/d.y992472.
3. Gupta A , Yilmaz A .Ubiquitous real-time geo-spatial localization[J].Proceedings of the Eighth ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness, 2016.DOI:10.1145/3005422.3005426.
4. Wang L M, Zeng C G, Zhao H L, et al. Development and application of intelligent scheduling system for shield tunnel vehicles under complex working conditions[J]. Science Technology and Engineering, 2024, 24(6): 2547-2553.
5. Wang F D. Multi-level clustering for vehicle scheduling in multi-vehicle type supply chain logistics and transportation networks[J]. Jiangsu Commercial Forum, 2025, 12: 45-48+61. DOI: 10.13395/j.cnki.issn.1009-0061.2025.12.004.
6. Zhang G C, Sun C L, Shi H, et al. Research on concrete mixer truck scheduling combined with daily vehicle inspection[J]. Industrial Engineering Journal, 2020, 23(2): 67-75.
7. Shao B L, Liu L. Research on distribution cost optimization of prefabricated components based on time window constraints[J]. Mathematics in Practice and Theory, 2019, 49(5): 99-109.
8. Wei R H, Zhang Z Q, Zhang K J, et al.Development and application of dynamic configuration and intelligent scheduling system for drilling and blasting tunnel construction equipment[J].Tunnelling and Underground Space Technology,2026,107164: 0886-7798,
9. Carneiro, J.; Rossetti, R. J. F.; Silva, D. C.; Oliveira, E. C. BIM, GIS, IoT, and AR/VR Integration for Smart Maintenance and Management of Road Networks: A Review. In 2018 IEEE International Smart Cities Conference (ISC2); IEEE, 2018; pp 1–7. https://doi.org/10.1109/isc2.2018.8656978.