UDC 004.89:656.13:69.05
DOI: 10.15587/1729-4061.2026.361917

DEVELOPMENT OF AN APPROACH TO MANAGING TRANSPORT LOGISTICS AT A CONSTRUCTION SITE

Pavlo Pasieka
Corresponding author
PhD Student*
E-mail: paaaa.paul@gmail.com
ORCID: https://orcid.org/0000-0001-8675-7300
Oleksii Panko
Candidate of Technical Sciences, Associate Professor
Department of Architectural Structures**
ORCID: https://orcid.org/0000-0002-3886-8476
Nataliia Poltorachenko
Candidate of Technical Sciences, Associate Professor*
ORCID: https://orcid.org/0000-0002-2238-6130
Svitlana Terenchuk
Candidate of Physical and Mathematical Sciences, Professor*
ORCID: https://orcid.org/0000-0001-6527-4123
Bohdan Yeremenko
Candidate of Technical Sciences, Associate Professor
Department of Management Technologies
Taras Shevchenko National University of Kyiv
Volodymyrska str., 60, Kyiv, Ukraine, 01033
ORCID: https://orcid.org/0000-0002-3734-0813
*Department of Information Technologies of Design and Applied Mathematics**
**Kyiv National University of Construction and Architecture
Povitrianykh Syl ave., 31, Kyiv, Ukraine, 03037

This study investigates the process that forms a vehicle route in a complex logistics environment. The task addressed relates to the lack of a full-fledged logic in autonomous decision-making regarding operational routing in specialized logistics systems.

An analysis of modern transport logistics management systems revealed their limitations in terms of the efficiency and explainability of the logic of autonomous decision-making regarding route replanning. An approach to route formation has been devised that combines the modified A* algorithm, multi-agent reinforcement learning, and spatio-temporal graph convolutional networks (STGCNs).

Distinctive features of this approach are the combination of spatial-temporal forecasting, multi-agent decision-making, and modular system architecture, which provides adaptability, scalability, resistance to communication degradation, as well as explainability of transport routing decisions in closed reconfigurable networks.

Experimental comparison of the devised approach and analogs based on A* and STGCN, carried out on a simulation model of the logistics environment at a construction site in residential area, showed its statistical superiority in terms of the route travel time criterion. With a threshold of this criterion of 300 s, the proposed approach achieved this indicator in 90% of cases, and the approaches based on STGCN and A* – in 79% and 57% of cases. In this case, the value of the 95th percentile was 310 s, 360 s, and 410 s for the proposed approach, STGCN, and A*. These results are explained by the combination of network state prediction and adaptive route replanning.

The findings could be used in the design of intelligent transport logistics management systems at other facilities with dynamic transport infrastructure

Keywords: A*, construction logistics, hexagonal environment model, machine learning, multi-agent, STGNN

Received 24.03.2026
Received in revised form 26.05.2026
Accepted 11.06.2026
Published 30.06.2026

How to Cite: *Pasieka, P., Panko, O., Poltorachenko, N., Terenchuk, S., Yeremenko, B. (2026). Development of an approach to managing transport logistics at a construction site. Eastern-European Journal of Enterprise Technologies, 3 (3 (141)), 41–53. https://doi.org/10.15587/1729-4061.2026.361917*

1. Introduction

Given the large-scale destruction caused by the military actions of the Russian Federation on the territory of Ukraine, construction projects in urban residential areas are becoming particularly urgent.

The industrial process at a construction site puts forward certain requirements on personnel safety, reasonable use of equipment, planning, and logistics.

First, the transport network of the construction site is closed and reconfigured in accordance with the work schedule.

Second, a heterogeneous fleet of construction equipment includes a large number of different machines and automated guided vehicles (AGVs). All this equipment with different dimensions, kinematic characteristics, and braking paths operates in the same confined space. The operation of different machines in a confined space requires coordination; this issue is still open both at the theoretical and practical levels [1].

Third, most urban intelligent transport systems for local positioning usually have acceptable accuracy in terms of GNSS (Global Navigation Satellite System) [2]. However, the accuracy of local positioning based on RTK-GNSS (GNSS with real-time kinematic positioning) in open space and UWB (Ultra-Wideband) wireless positioning technology near metal structures goes beyond these limits.

Under the specified conditions:

– a construction site is transformed into a complex dynamic stochastic decentralized partially observable logistics environment with unstable communication in which different vehicles operate simultaneously;

– vehicle control in such a logistics environment shifts from static route optimization to adaptive control over the environment, which differs fundamentally from an urban road network [3, 4].

In addition to the conditions under which the system being designed must operate effectively, the specific nature of construction requires the interpretability of system outputs to ensure the safety of the production process. Therefore, another requirement for a transport logistics management system at a construction site is a high level of explainability with the capability to reproduce the logic that determines the choice of the final route. Moreover, the development of adaptive logistics systems is exacerbated by the requirements for scalability, adaptability to topology changes, as well as suitability for deployment under unstable communication conditions.

Thus, the development of transport logistics management systems adapted to specific environmental conditions remains a relevant area of research.

2. Literature review and problem statement

Modern transport logistics management systems at a construction site are divided into the following groups:

– focused on BIM (Building Information Modeling) and IoT (Internet of Things) streaming platforms;

– digital twins;

– automated guided vehicle coordination systems;

– transport dispatching systems.

Each of these groups solves separate tasks of modeling, monitoring, planning, or coordinating transport processes. However, the capabilities of these systems vary significantly depending on the type of environment, the level of the transport network dynamism, and the requirements for autonomous decision-making.

Paper [5] reports the results of research on the integrated digital framework "BIM 4D–7D – IoT – digital twin – predictive analytics", which covers processes from design to operation in accordance with the project schedule. It is shown that the use of BIM 4D and the digital twins makes it possible to model reconfigurable construction site networks, plan work stages, and reflect changes to the object in the digital environment. However, issues related to the operational use of real-time data for replanning vehicle routes remain unresolved. The reason is that BIM-oriented systems mainly rely on the work schedule and short-term detailed planning, rather than on the continuous updates to the state of the transport network.

As shown in [4], static or quasi-static planning is one of the main limitations of BIM/4D logistics systems. Other significant limitations of these systems include weak support for real-time route replanning, limited integration of streaming telemetry, and lack of traffic conflict prediction. These limitations could be partially addressed by integrating streaming data processing and environment-state prediction into BIM-oriented models. However, such integration alone does not solve the problem of autonomous decision-making for route formation in a dynamic environment.

In [6], the main limitations of real-time IoT platforms were investigated. Such platforms are primarily designed for the transmission, accumulation, and processing of data from distributed sources. However, issues related to high-level vehicle coordination logic, multi-agent planning, the integration of spatial models, and ensuring state consistency in distributed systems remained unresolved. The reason is that IoT platforms perform the role of data collection and transmission infrastructure but do not always incorporate mechanisms for intelligent situation analysis, forecasting the consequences of the transport network changes, and selecting the optimal route.

In [7], the limitations of digital twin systems were investigated. It was shown that a digital twin makes it possible to represent the state of a physical object in a digital environment, integrate heterogeneous data sources, and support the analysis of changes occurring within the environment. However, issues related to the synchronization of the physical and digital environments under numerous dynamic changes, autonomous decision-making, and high requirements for streaming data remained unresolved. The reason is the difficulty of maintaining an up-to-date digital representation of a construction site where the transport network, the position of equipment, temporary obstacles, and the accessibility of individual paths can change within a short period.

It is shown in [8] that the key limitations of such systems are the lack of reliable mechanisms for autonomous decision-making and insufficient support for adaptive coordination of transport flows in real time. This allows to argue that modeling, monitoring, and dispatching systems alone do not provide a comprehensive solution to the problem of transport logistics management in a complex construction site environment. Such an environment requires not only fixing the current state but also the ability to predict changes in the traffic conditions, coordinate the actions of multiple vehicles, and replan routes under conditions of unstable communication with the decision-making center. One of the options for overcoming these difficulties is the use of RL (Reinforcement Learning) methods, which make it possible to form an agent's behavior policy in a changing environment.

In [9], a mining RL logistics approach is described that uses machine learning with reinforcement for adaptive control of transport processes in a dynamic environment, uncertainty, and a large number of interconnected agents. It is shown that this approach can work with closed reconfigurable networks at the production level. However, issues related to the combination of environmental prediction and route planning, ensuring system stability in the event of a control node failure, slow convergence in the presence of many agents, and insufficient explainability of the decisions made remain unresolved.

In addition to solutions that are directly focused on construction or production environments, there are approaches to building intelligent logistics systems for urban transport networks. In [2], innovative solutions to the problem of optimizing traffic using big data, artificial intelligence models and Internet of Things technologies were investigated. The possibility of forming the concept of an ontology of a "solution model" for managing traffic flows and defining tasks that can be solved by artificial intelligence models was shown. However, this study is focused on adaptive traffic management in open urban networks, where the topology of the transport network is relatively stable, and not on closed construction sites with constant reconfiguration of available paths. In [10], the architecture of an intelligent traffic management system for a large city, modeling the road network and traffic flows, is described. It is shown that the distributed architecture of the system can combine scalable technologies and batch data processing. This provides the ability to adaptively manage traffic flows. However, [10] focuses on traffic light control at a complex intersection in an open urban network. Therefore, this approach does not take into account the specifics of a construction site, where there is no fixed road infrastructure, and vehicles have different dimensions, kinematic characteristics and traffic restrictions.

In [11], the DiDi system is presented, which is used for ordering and repositioning a taxi fleet. It is shown that such systems are able to coordinate a large number of vehicles in an urban environment and use machine learning methods to improve the efficiency of logistics solutions. However, such a system requires constant communication with the control center, is characterized by slow convergence in the presence of many agents, has limited explainability and also operates on open urban networks. This limits the possibility of directly transferring such an approach to construction site conditions.

The analysis showed that most of the existing solutions were developed either for quasi-static planning of construction processes, or for open urban transport networks, or for monitoring and data processing systems without full-fledged logic of autonomous decision-making. An option to overcome these difficulties may be to develop an approach that will provide a specialized logistics system with the ability to function reliably and efficiently in the complex logistics environment of a construction site. Such a system should take into account the dynamic change of the transport network, provide adaptive replanning of routes in real time, support work under unstable communication conditions and provide explanations for the decisions made. At the same time, it should remain scalable for use on construction sites of different sizes and equipment composition.

3. The aim and objectives of the study

The aim of the study is to develop an approach to transport logistics management at a construction site that will ensure adaptive formation of vehicle routes in real time in accordance with changes in the state of the logistics environment. The application of this approach is aimed at reducing the time of route passage and increasing the stability of logistics processes to changes in the transport situation and temporary network restrictions.

To achieve the aim, the following objectives must be solved:

– to justify the choice of a traffic environment model and routing methods that will ensure adaptive implementation of vehicle route replanning in real time;

– to develop an architectural model of the system that will satisfy the requirements of adaptability, explainability of the route formation logic, scalability and resistance to loss or deterioration of communication;

– to experimentally evaluate the effectiveness of the approach.

4. Materials and methods

4. 1. The object and hypothesis of the study

The object of the study is the process of forming a vehicle route in a complex construction site environment.

The assumptions that were made before the start of the study were that a hybrid approach to forming vehicle routes in a construction site logistics environment should combine:

– an incremental algorithm for finding the final route;

– MARL for determining the corridor of the best routes;

– STGNN for predicting edge weights on the planning horizon.

The main hypothesis of the study is that an approach based on STGNN, MARL and the modified A* algorithm will better meet the conditions and requirements of a complex construction site environment than approaches based on A* or STGNN separately.

Significant simplifications of the system are the projection nature of the results of its work and behavior on a simulation model of the logistics environment, which is associated with confidentiality and security restrictions.

4. 2. Models of the logistics environment

One of the fundamental problems of transport logistics of a construction site is adequate modeling of the logistics environment.

The material of this study is a simulation model of the logistics environment of a construction site of a quarter-urban development (Fig. 1), which reproduces the routes of vehicles to logistics facilities and events that affect their availability.

Fig. 1. Simulation model of the logistics environment of the construction site of a quarter-city development

The model took into account speed limits on technological roads, speed reduction on turns and narrow sections, delays at intersections of routes, as well as the possibility of local overlap of individual edges of the graph. This model can be a digital twin of a real construction site, intended for experimental evaluation of vehicle routing algorithms in closed reconfigurable networks.

The graph model of such an environment is formed by highlighting significant elements of the site plan:

– graph vertices – road junctions, intersections, turning points and access points to construction zones;

– graph edges – temporary and permanent roads, technological routes and areas of possible vehicle movement between adjacent vertices.

Thus, topological road graphs directly encode permissible routes, reproducing the current topology of the road. However, using only a graph model of the logistics environment complicates spatial generalization of information, prediction of the state and constraints of transport networks.

The use of regular grids is better suited for reflecting the topological constraints of the road network, since the grid provides spatial indexing and aggregation of the state of the environment. However, the use of square grids, as standard structures for building an environment model based on sensor data, distorts the path metric and complicates the smoothing of trajectories due to the asymmetry between the 4-connectivity and 8-connectivity of the model [19]. This anisotropy distorts both the convolution and the shortest path heuristics at a scale of 1–3 m, which is essential for the operation of equipment on a construction site. Hexagonal grids eliminate the problem of anisotropy. This was shown in the comparative analysis of path planning for AGVs in production environments on real maps conducted in [20]. And in [21] it is shown that the use of hierarchical hexagonal indexing (H3) of level 9 is successfully used for aggregation of characteristics of the urban environment and prediction of delivery times in different types of buildings. This confirms the advantage of using a multi-level hierarchy of spatial cells with stable identifiers. Thus, the combination of H3 with an explicit road graph is methodologically justified and the key practical advantage of such a combination is that changes in the physical environment require only local restructuring of the graph.

4. 3. Routing methods

Each category of technologies that can be used in the development of an approach to vehicle control in a complex logistics environment includes several competing implementations with different characteristics. Therefore, it is necessary to justify the choice of STGNN varieties, multi-agent reinforcement learning algorithms and modifications of the A* algorithm that will provide adaptive replanning of vehicle routes in real time at a construction site.

Existing route search methods are divided into classical and RL.

The classical method of finding the shortest path according to the Dijkstra algorithm in a weighted graph with non-negative edge weights explores the space uniformly in all directions. However, this algorithm has a complexity of $O((V+E)\log V)$, which is excessive for problems with a known target point [12]. This means that the use of Dijkstra's algorithm to find the shortest path of a vehicle on a construction site is unjustified despite the guaranteed optimality.

The A* algorithm extends Dijkstra's algorithm with a heuristic function that estimates the distance from a given node to the target. Thus, under the admissibility of the heuristic, A* guarantees optimality by significantly reducing the number of nodes to be investigated [13]. A significant limitation of A* is the difficulty of scaling in dynamic environments.

The main competitors of A* in the incremental algorithm category are Focussed D* [22], LPA* [23] and D* Lite [24].

Focussed D* implements the same replanning strategy as D* Lite, but it is algorithmically more complex to analyze and extend. LPA* computes the shortest paths from a fixed origin to a fixed goal while changing edge weights, but does not support moving the starting point, which is unacceptable for a moving vehicle. D* Lite combines the properties of both: searching from the goal to the current position of the agent, incremental use of previous search results, and algorithmic simplicity, which facilitates adaptation and verification of correctness [24]. For a construction site, where the starting point (position of the vehicle) changes and the graph topology is updated locally, D* Lite is a better choice.

To coordinate multiple autonomous agents that simultaneously interact in a shared environment, it is advisable to use MARL (Multi-Agent Reinforcement Learning) [14].

Agent coordination is implemented through the CTDE (Centralized Training with Decentralized Execution) paradigm.

In this case [15]:

– during training, the centralized critic uses global information about the state of the environment and information from all MARL agents to evaluate their joint behavior;

– in the operational mode, agents act decentralized based on their own local observations;

– coordination of actions occurs implicitly due to the joint reward function and joint learning of the behavior strategy, so agents are able to make decisions locally, which ensures scalability and resilience to failures of individual system components.

The optimal MAPF (Multi-Agent Path Finding) problem in a decentralized mode is solved by the Conflict-Based Search (CBS) method using a two-level algorithm [16]. To form conflict-free routes, CBS builds a conflict tree at the lower level and paves the agent's path at the upper level. However, the large number of simultaneously operating vehicles makes CBS practically unsuitable for optimal solution of the MAPF problem on construction sites in real time due to NP-complexity.

In the category of MARL algorithms in the CTDE paradigm, the main competitors are MADDPG (Multi-Agent Deep Deterministic Policy Gradient) [26], QMIX (Monotonic Q-value Mixing Network) [27] and MAPPO (Multi-Agent Proximal Policy Optimization) [15].

MADDPG is designed for continuous action spaces and deterministic policies. This makes it difficult to learn stably in cooperative tasks with a large number of agents [26]. QMIX uses monotonic factorization of the team value function through the individual value functions of the agents. This simplifies the coordination of agents, but narrows the class of strategies that QMIX can approximate. This does not allow modeling situations where the beneficial action of one agent reduces the utility of another [27]. MAPPO adapts the PPO (Proximal Policy Optimization) algorithm to a multi-agent environment with a centralized critic and shows high efficiency in cooperative tasks no worse than specialized MARL algorithms with much simpler implementation and more stable learning [15]. Stability of learning is a critical requirement for application on a construction site, where the possibilities for long-term fine-tuning in the field are limited; MAPPO provides this property thanks to the built-in PPO mechanism for limiting the policy update step size, which prevents too sharp changes in agent behavior between training iterations.

STGNN (Spatio-Temporal Graph Neural Network) outperforms traditional MAPF approaches that do not take into account the spatio-temporal load forecast by modeling the diffusion of flow over the graph [17]. In addition, the use of exclusively convolutional operations over both the graph and time provides them with an advantage over hybrid recurrent approaches due to the smaller number of parameters and higher computational efficiency [18]. In this category, DCRNN (Diffusion Convolutional Recurrent Neural Network) [17], STGCN [18], and Graph WaveNet [25] compete for the prediction of the state of the transport network.

DCRNN uses recurrent GRU cells for the temporal dimension and diffusion convolution for the spatial dimension, which provides high prediction quality, but is relatively slow in training and inference due to the sequential nature of the recurrent blocks.

Graph WaveNet replaces recurrence with extended causal convolutions and introduces an adaptive adjacency matrix, which allows DCRNN to detect non-obvious spatial dependencies. This is a significant advantage for large and complex networks, but is overkill for the compact construction site graph.

DCRNN and Graph WaveNet models are effective for large urban networks with thousands of nodes, but for the construction site graph, which typically contains dozens of nodes, their computational complexity is unjustified. In contrast, STGCN completely avoids recurrent blocks, since both spatial and temporal dimensions are processed by convolutions. This reduces the training time and the number of parameters with comparable prediction accuracy [18]. Thus, STGCN provides sufficient prediction reliability with lower computational requirements and simpler hyperparameter tuning.

Thus, classical routing methods are deterministic and explainable, while RL can at best offer only partial explainability. However, classical algorithms do not sufficiently satisfy the requirements of scalability and adaptability, since they are unable to predict the state of the environment and require a complete recalculation of the route at the slightest changes in the environment.

These arguments justify the feasibility of developing an approach to transport logistics management at a construction site that combines the following models and methods:

– MARL for determining the corridor of best routes;

– D* Lite for finding the final route within the corridor;

– STGNN for predicting edge weights at the route planning horizon.

4. 4. Hardware and software, input and output data of experimental modeling

Experimental modeling of vehicle movement in the construction site environment was performed on a personal computer with a 13th Gen Intel(R) Core(TM) i7-13700H processor with a base clock frequency of 2.40 GHz, 64.0 GB of RAM and an integrated graphics adapter. Windows 11 Pro and Linux Ubuntu 24.04 LTS were used as operating systems, depending on the stage of the experiment and the requirements for launching individual software components.

To conduct the experimental study, the following were used:

– SUMO 1.27.0 software package (Simulation of Urban MObility, Germany) for simulation modeling of vehicle movement;

– Python 3.12.10 programming language for preparing source data, launching simulation scenarios and processing results;

– Python library OSMnx 2.1.0 (USA) for automatic loading and processing of the transport network graph;

– netconvert 1.27.0 utility for converting the road network graph of the construction site $G = (V, E)$ into SUMO format;

– NetworkX 3.6.1 library for working with the graph structure;

– NumPy 2.4.6 and pandas 3.0.3 libraries for numerical processing of the results;

– PyTorch 2.7.0 library for local launch of the simplified STGCN implementation.

The main metric of the experimental comparison is the route travel time, since this indicator directly reflects the efficiency of logistics management in the complex conditions of the construction site.

The input data of the experiment were:

– a simulation model of the logistics environment of the construction site of a quarter-urban development (Fig. 1), on the basis of which a directed weighted road network graph of a synthetic construction site was constructed;

– vertex coordinates and basic information (Table 1);

– graph edges; section lengths; basic edge weight (expected section travel time assuming no delays or overlaps) (Table 2);

– data on vehicles and their operating time and information about each trip (Table 3).

Based on these data, a set of route queries and the corresponding SUMO script file were generated for each run.

Table 1

Route graph vertex information form

Vertex (V_k)		Basic information			
No.	Coordinate ($k = 1, \ldots, K, K = 241$)	Characteristic	Scope of work		
			Plan	Actual	Deviation
1	V_1	Initial	pt_1	rt_1	$\Delta_1 = rt_1 - zt_1$
2	V_2	Loading	pt_2	rt_2	$\Delta_2 = rt_2 - zt_2$
…	…	…	…	…	…
k	V_k	Unloading	pt_k	rt_k	$\Delta_k = rt_k - zt_k$
…	…	…	…	…	…
K	V_K	End	pt_k	rt_k	$\Delta_K = rt_K - zt_K$

Table 2

Route graph edge information form

Edge characteristics (E_l, $l = 1, \ldots, L, L = 389$)								
No.	Length	Base weight	Overlap		Permissible speed	Throughput	Real weight	Deviation time ($\Delta_{k,k+1,}$ $k = 1, \ldots, K$)
			Attribute	Time				
1	$l_{1,2}$	$w_{1,2}$	nS	…	$v_{1,2}$	$pz_{1,2}$	$rw_{1,2}$	$rw_{1,2} - w_{1,2}$
2	$l_{2,3}$	$w_{2,3}$	nS	…	$v_{2,3}$	$pz_{2,3}$	$rw_{2,3}$	$rw_{2,3} - w_{2,3}$
…	…	…	nS	…		…	…	…
l	$l_{k,k+1}$	$w_{k,k+1}$	S	T_0, ΔT	$v_{k,k+1}$	$pz_{k,k+1}$	$rw_{k,k+1}$	$rw_{k,k=1} - w_{k,k=1}$
…	…	…	…	…	…	…	…	…
L	$l_{K-1,K}$	$w_{K-1,K}$	nS	…	…	$pz_{K-1,K}$	$rw_{K-1,K}$	$rw_{1,2} - w_{1,2}$
Σ	$\sum l_{k,k+1}$	$\sum w_{k,k+1}1$	…	…	…	…	$\sum rw_{k,k+1}$	$\sum \Delta_{k,k+1}$

Table 3

Form of information about the vehicle and its route

ID	Vertex	Time			Trip information		
		Start	Finish	Construction site work	Time to cover the route section	Average speed	Time loss
ID_{TZ}	V_1	st_1	ft_1	$\delta_1 = ft_1 - st_1$	Data from Table 2	v	Data from Table 2
	V_2	st_2	ft_2	$\delta_1 = ft_2 - st_2$			
	…	…	…	…			
	V_k	st_k	ft_k	$\delta_k = ft_k - st_k$			
	…	…	…	…			
	V_K	st_k	ft_k	$\delta_1 = ft_{K-1} - stK$			
Σ	K	…	…	$\sum\delta$	$\sum rw_k$	$\sum l_k / \sum rw_k$	$\sum\delta + \sum rw_k$

The initial data for the simulation was a file obtained through the tripinfo.xml file generation mechanism in the SUMO environment.

5. Results of the development of an approach to transport logistics management at a construction site

5. 1. Justification of the choice of a traffic environment model and methods for adaptive replanning of vehicle routes

Justification of the traffic environment model and routing methods for adaptive replanning of routes in real time involves the choice of a method for representing the environment, forming a corridor of the best routes, predicting the weights of its edges and final route planning.

In this work, it is proposed to use a hierarchy of hexagonal cells H3 (Fig. 2, *a*), which reflects the aggregated state of the territory, and a road graph (Road graph) – $G = (V, E)$, which reflects the topology of vehicle movement, to implement the logistics environment model. The connection between these representations is implemented on the micro layer H3 (micro H3 layer), the size of which is selected for the dimensions of the vehicle.

The "Road graph" block contains edges (e_1, e_2, …, e_k), which correspond to individual sections of the construction site roads. Each edge is mapped to one or more cells of the H3 microlevel using a spatial mapping (φ). The relationship between macro and micro cells is determined by the standard H3 hierarchy, according to which each macrocell contains a set of nested microcells, and it did not change within the framework of this study.

The choice of two-level H3 indexing when modeling the logistics environment of a construction site is justified by the need to simultaneously support two resolutions:

– at the macrolevel, cells of scale 11–12 are suitable for GNN, but too large for resolving conflicts between machines;

– at the microlevel of scale 14–15, the number of nodes for GNN becomes orders of magnitude larger than the spatial process requires.

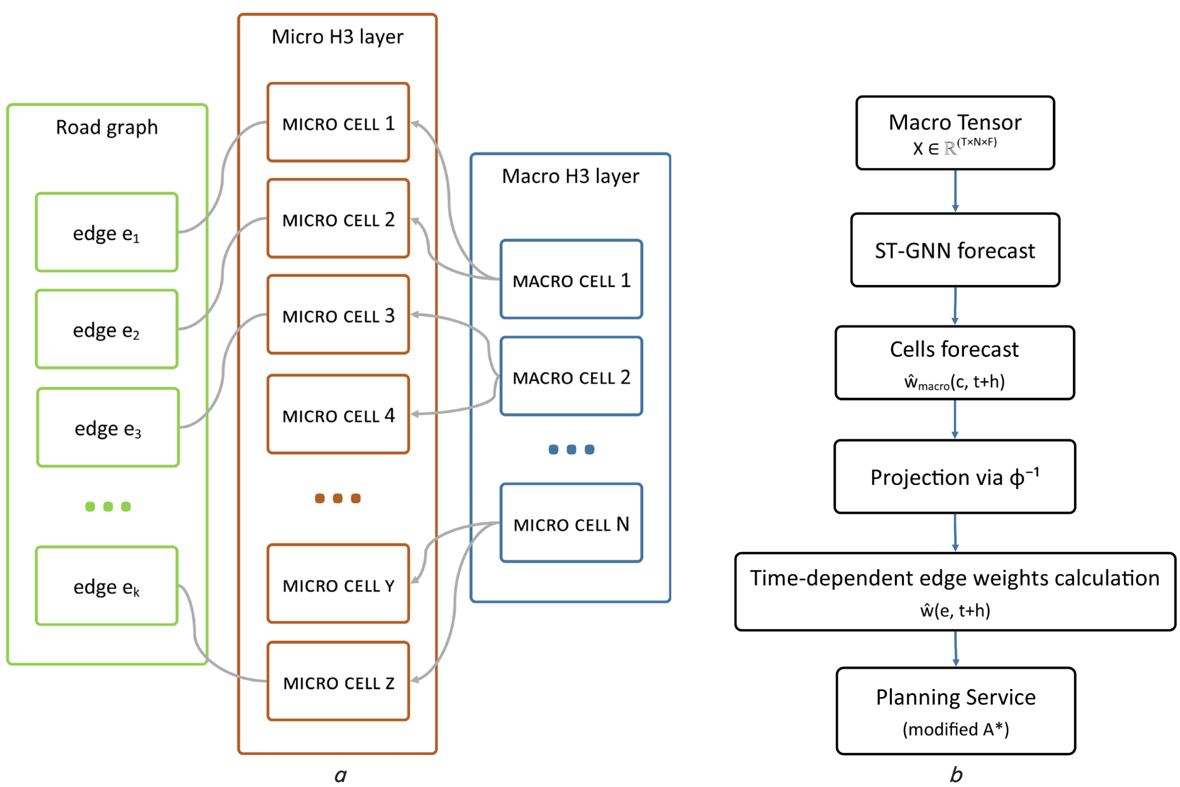

Fig. 2. Transport logistics control schemes at a construction site:
a – modeling of the logistics environment; *b* – formation of vehicle routes

The use of other levels of H3 indexing for a limited area of a construction site is redundant for performing the functions of the system.

Fig. 2, *b* shows the process of forming vehicle routes.

The "Macrotensor" block contains a time series of features for (N) spatial elements, where (T) is the number of time steps, (N) is the number of nodes or cells, (F) is the number of features. The features include traffic load, average speed, site accessibility and the presence of local restrictions. MARL is used to determine the corridor of the best routes on the H3 macrolayer (Fig. 2, *a*).

The "ST-GNN-forecast" block performs spatio-temporal prediction of the network state taking into account the connections between neighboring elements and changes in their parameters over time. The result is a forecast of the state of spatial elements at time $(t + h)$.

The "Cell Prediction" block converts the ST-GNN output into predicted weights of hexagonal cells $(\hat{w}_{macro}(c, t + h))$. These weights characterize the predicted difficulty or feasibility of passing the corresponding cell, taking into account traffic load, risk, accessibility, and other constraints.

The "Projection via φ^{-1}" block transfers the predicted values from hexagonal cells to the edges of the road graph. For each edge, the cells through which it passes are determined, after which their predicted values are aggregated.

The "Calculation of time-dependent edge weights" block forms the weight of the edge $(w(e, t + h))$, which takes into account its base weight and the predicted state of the cells associated with it. The resulting weights reflect the expected passage time or temporary inaccessibility of the section.

The "Planning Service" block uses a modified A* algorithm to search for a route on the road graph with updated time-dependent weights. In case of a change in the forecast or the state of the environment, the edge weights are recalculated, after which adaptive route replanning is performed.

The two-way connection between H3 cells and graph nodes makes it possible to form spatial features for STGNN without coordinate transformation.

The forecast for H3 cells (Fig. 2, *b*) is projected onto the road graph (Fig. 2, *a*) through "macro-micro" inheritance and backlinking according to:

$$\hat{w}(e, t + h) = \hat{w}_{macro}(\text{parent}(\varphi(e)), t + h), h = 1, \ldots, H, \tag{1}$$

where $\hat{w}$ – the predicted weight of edge e of graph G at time $t + h$; $\hat{w}_{macro}$ – the predicted weight of edge e from the macro-cell (Fig. 2, *a*); h – the prediction step; H – the prediction horizon.

Thus, each edge e of the road graph G is assigned a predicted weight from the macro-cell, which is the ancestor of its micro-cell.

A two-way binding to the road graph is defined on the micro-layer, according to:

$$\varphi: E \to C_{micro}, e \mapsto \varphi(e), \tag{2}$$

$$\varphi^{-1}: C_{micro} \to 2^E, c \mapsto \{e \in E: \varphi(e) = c\}, \tag{3}$$

where φ – the mapping that assigns a micro-cell C_{micro} to each edge $e \in E$; φ^{-1} – inverse mapping that defines the edges of the graph that belong to the selected microcell C_{micro}.

The aggregated weights on the macrocell are calculated from the edges of all child microcells, and the predicted weights are returned to the edges of the graph G through φ^{-1}. Thus, in the event of a trench opening or a transfer of the route for any other reason, the graph can be regenerated from H3 telemetry without retraining the predictor, which works on H3, and not on G.

5. 2. Architectural model of a specialized intelligent information system for managing transport logistics

Fig. 3 shows the architectural model of a specialized intelligent information system for managing transport logistics on a construction site (Intelligent Information System for Managing Transport Logistics on a Construction Site – IISMTLCS), which is proposed for adaptive formation of vehicle routes on a construction site

This model:

– takes into account the interaction of the system with individual vehicles and external information sources;

– combines event-driven microservices, which will provide the system with the ability to remain scalable for use on construction sites of various sizes and equipment composition.

Each microservice has a separate environment model layer and interacts with the rest of the system through explicit event log topics. The correspondence between architectural layers and services is direct: one layer - one service.

Apache Kafka simultaneously performs the functions of a message bus, an event log, and a system state recovery mechanism:

– graf.uhdate – a topic for changing the state of the environment, namely the road graph;

– telemetry – a topic for updating information about vehicles;

– forecasts – a topic for predicting the state of the environment;

– corridors – a topic for forming corridors of the best routes;

– routers – a topic for forming routes consumed by vehicles.

The Graph Service is responsible for normalizing the received environment data and also forms an environment graph from it. The Graph Service has a graph $G = (V, E)$ and a two-way connection to H3. A topology change is delivered as a single event to the graph.updates topic, and each subordinate service reacts independently.

The Aggregation Service, implemented using Apache Flink [26], performs stateful streaming processing that transforms raw telemetry into macro-layer features F that are consumed by the forecaster.

The Forecast Service, implemented using Ray RLlib, has an STGCN model and publishes weight forecasts.

The Meta-Controller Service, which has corridor logic and MAPPO inference, is implemented using Ray Rllib.

The Route Planning Service, using D* Lite, publishes routes based on the generated forecasts and corridors.

This architecture model has three main advantages:

– component separation means that the prediction service can be replaced from the STGCN class [18] to DCRNN [17] or ASTGCN [28] without impacting the scheduler, as the interface – the predicted weights topic – remains stable;

– state recovery via the event log means that any service failure is recovered by reproducing the relevant Kafka topic prefix, without manually restarting the dependency chain;

– exactly-once semantics in Flink aggregations ensure that macro-layer features are computed correctly even in the presence of rework and message redelivery. This is an important condition for auditability, as all edge weights included in the scheduler's decision are exactly reproducible from the log.

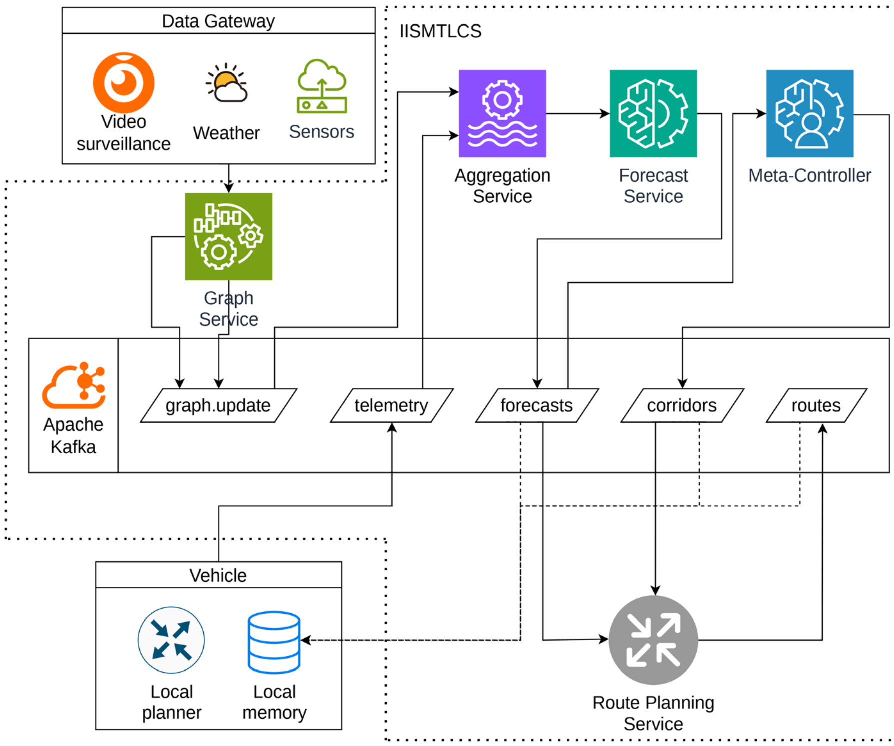

Fig. 3. IISMTLCS architectural model

5. 3. Experimental evaluation of the effectiveness of finding the final route

The aim of the experiment was to determine the ability to form routes with minimal time to overcome them of the A* algorithm, the STGCN model and the proposed approach, which combines MARL to determine the corridor of the best routes; D* Lite to find the final route within the corridor; STGNN to predict the weights of edges on the route planning horizon.

Testing was performed on a model of the logistics environment of a construction site (Fig. 1).

To build a transport network graph, a construction site plan (Fig. 1) and the Python library OSMnx (USA) were used, which automates loading and.

The procedure for processing input data consisted of several stages.

At the first stage, the logistics environment of the construction site was transformed into a road network graph $G=(V, E)$. The forms of information about the vertices and edges of the road graph are shown in Tables 1, 2.

At the second stage, the traffic parameters and the basic weight were determined for each edge of the graph (Table 2).

At the third stage, the basic and crisis scenarios were formed:

1) basic (basic edge weight and standard vehicle traffic intensity);

2) crisis (random time intervals of overlapping of individual edges of the graph were set, which simulated delays, changes in vehicle traffic intensity over time, local overlap of passages as a result of trench digging, and vehicle collisions).

To simulate dynamic changes in the traffic situation in the SUMO environment, rerouter objects were used, which temporarily overlapped individual edges of the graph at different time intervals. This allowed to simulate a crisis scenario and assess the effectiveness of the route replanning algorithm in conditions of a changing traffic situation.

At the fourth stage, the graph was converted to SUMO format using the netconvert utility.

In the fifth stage, multiple microsimulation runs were performed for the A*, STGCN, and the proposed hybrid approaches.

In the sixth stage, the generated SUMO tripinfo.xml files were processed using Python to extract the values of the route travel time, average speed, and time losses (Table 3).

In the seventh stage, the obtained route travel time values were sorted in ascending order and used to construct cumulative distribution function graphs (Fig. 4).

The output data was collected through the tripinfo.xml file generation mechanism, which allowed to obtain detailed information about each trip and measure key traffic metrics: trip duration, average speed, and time loss during replanning (Table 3).

To ensure the reproducibility of the experimental data processing procedure, a results table was formed for each approach in the following form: launch identifier, scenario type, initial vertex, final vertex, number of active vehicles, list or number of overlapped edges, overlap start time, overlap end time, route formation algorithm, actual route travel time, average speed, and time loss. This table was the starting point for further statistical analysis and construction of CDF curves.

To ensure the correct construction of the empirical cumulative distribution function, comparison of average values and behavior of algorithms on the entire set of scenarios of each approach, 500 experimental observations were generated. One observation corresponded to the result of the route passage for a given pair of initial and final points, a given network state and a specific scenario of overlap or edge loading.

The evaluation was carried out on a set of basic and crisis scenarios and their combinations. For each scenario, multiple simulations were performed, and the result of each run was the actual route passage time.

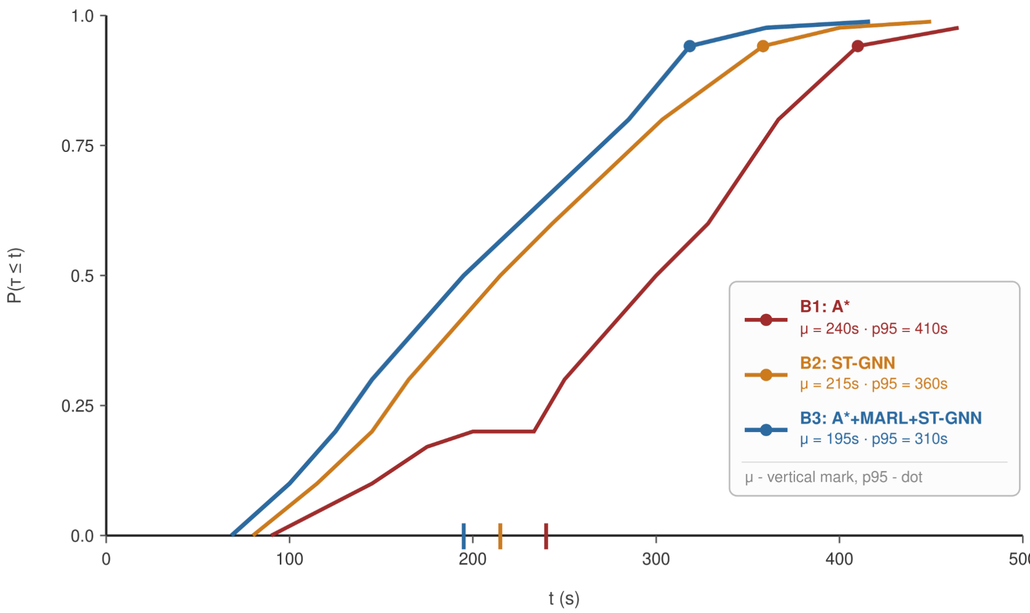

Fig. 4. Cumulative distribution function curve graphs

To compare the approaches, the cumulative distribution function (CDF) was used, which allowed to estimate the average result and the general behavior of the approach on the entire set of scenarios.

The practical CDF construction was carried out as follows:

1) a set of route passage time results was collected for each approach;

2) the obtained values were sorted in ascending order;

3) for each time value x, the value of the empirical distribution function is calculated

$$\hat{F}(x)=\frac{1}{n}\sum_{i=1}^{n}I(t_i \le x), \tag{4}$$

where n – the number of experiments; t_i – the route time of the i-th experiment; $I(*)$ – an indicator function equal to 1 if the condition is met, and 0 otherwise.

The obtained CDF curves showed a significant difference between the studied approaches (Fig. 4).

The curve of the cumulative distribution function of the route time for the proposed approach is located to the left and above the CDF curves of the route time for alternative approaches (A* and STGCN), which indicates its dominance by the route time criterion. This means that the proposed approach is more likely to form faster routes for all time thresholds.

At a threshold value of the route time of 300 seconds:

– A* achieved this indicator in 57% of cases;

– STGCN achieved this indicator in 79% of cases;

– the proposed hybrid approach achieved this indicator in 90% of cases.

In addition, the curve for the proposed approach had a steeper growth pattern, indicating lower variance of results and higher stability of operation.

Analysis of the 95th percentile metric confirmed the superiority of the proposed hybrid approach by demonstrating lower values of route travel time.

The results of the experimental comparison confirmed the effectiveness of the proposed approach for forming logistics routes. CDF analysis showed its statistical dominance over A* and STGCN both in terms of delivery speed and stability of results. Thus, the results obtained confirmed the main hypothesis of the study.

6. Discussion of the results of the study of the approach to adaptive route formation at the construction site

The main result of the study is the developed approach to the formation of vehicle routes in the logistics environment of the construction site.

The method of representing the logistics environment in the form of a road graph, consistent with the two-level hexagonal model H3, is substantiated. The relationship between the edges of the road graph, microcells and macrocells is shown in Fig. 2, *a*. The spatial mapping (φ) establishes a correspondence between the edges of the graph and the microcells through which the corresponding sections of the transport network pass. The inverse mapping (φ^{-1}) allows to transfer the predicted characteristics of the cells to the edges of the graph. This provides the opportunity to use the hexagonal representation for spatio-temporal analysis of the environment, and the road graph for the final route search.

Presented in Fig. 2, *b* the process explains the formation of a vehicle route as a combination of three components: prediction of the environment state using STGCN [18], formation of the corridor of best routes through MARL in the CTDE paradigm [15] and final route planning using D* Lite [24]. Unlike the use of a single class of algorithms, the combination of these components provides adaptability, scalability and explainability of the results. It takes into account the current state of the transport network, predicted changes in its capacity, possible conflicts between vehicles and temporary overlaps of individual sections of the graph.

The modular structure of the developed architectural model of the system (Fig. 3) separates data collection, spatial normalization, prediction, multi-agent assessment of the environment and final planning. This distribution of functions ensures the ability of the system to maintain adaptability and scalability, since individual components can be updated or replaced without changing the general logic of its functioning.

The results of the experimental study showed that the proposed approach statistically dominates over analogues based on A* and STGCN by the criterion of route travel time. This is confirmed by the cumulative distribution function graphs shown in Fig. 4. The curve of the proposed approach is located to the left and above the curves of alternative approaches, which indicates a higher probability of forming faster routes in most scenarios. In particular, at a threshold value of the route passage time of 300 seconds, the A*-based approach achieved this indicator in 57% of cases, the STGCN-based approach achieved this indicator in 79% of cases, and the proposed hybrid approach achieved it in 90% of cases. In addition, the steeper nature of the growth of the CDF curve indicates a smaller dispersion of results and a more stable behavior of the system in crisis scenarios. The analysis of the 95th percentile also confirmed the superiority of the proposed approach, since even in the worst-case scenarios the system demonstrated a lower route passage time compared to alternative solutions.

The peculiarity of the obtained results compared to the classical Dijkstra [12] and A* [13] path finding algorithms is that the proposed approach is able to take into account the predicted state of the environment and perform adaptive route replanning without completely recalculating the entire network. Classical algorithms are effective for finding a route in a given graph, but they do not take into account future changes in edge weights, the appearance of temporary obstacles and the mutual influence of several vehicles. In the complex environment of a construction site, these factors are significant, since the transport network is constantly changing, and individual sections may become inaccessible or change their capacity.

Compared to BIM and digital twin-based systems [3, 5], the proposed approach is focused not only on static or quasi-static planning according to the work schedule, but also on working with flow changes in the transport environment. Systems of this class are well suited for representing the state of the object, modeling construction stages and planning work, but their capabilities for real-time route replanning remain limited. The proposed approach complements this logic with the ability to predict the state of the transport network and adaptively form routes taking into account the current situation. Compared to RL-based approaches to production logistics [9], the proposed solution is distinguished by the combination of decentralized decision-making with the ability to audit the routing process. This is important for a construction site, since the system must not only choose an efficient route, but also explain what parameters influenced its choice. That is why explainability is not an additional property, but one of the key requirements for a transport logistics management system in a complex environment.

Compared to urban traffic flow management systems [2, 10, 11], the proposed approach is adapted to the closed reconfigured environment of a construction site with an unstable road network topology and unstable connectivity. Urban transport systems usually operate with open networks, where the road topology is relatively stable, and the main task is to optimize flows on a city scale or individual intersections. In a construction environment, the situation is different: the road network can change in a short time, the availability of individual roads depends on the stage of work, and vehicles have different dimensions, kinematic characteristics and traffic restrictions.

Thus, the results of the study confirmed the main hypothesis of the work and showed that the proposed approach to the formation of vehicle routes better meets the conditions of the complex environment of a construction site compared to existing approaches. Its advantage is explained by the combination of spatiotemporal forecasting, multi-agent decision-making, hierarchical representation of the environment and incremental route planning. At the same time, the practical use of the obtained results requires further verification on real data, expansion of the number of scenarios and consideration of additional factors characteristic of real construction sites.

The obtained results also confirm the feasibility of using a hierarchical representation of the environment based on H3. Unlike regular square grids, the hexagonal structure reduces the anisotropy of the model and provides a more natural distribution of transport flows between neighboring cells [19]. The use of two levels of H3 made it possible to avoid excessive complexity of the model while maintaining sufficient spatial detail for forecasting and deconflicting routes. This also made it possible to combine the macro-level analysis of the transport environment with local route search on the road network graph.

The results of the study confirmed the feasibility of using STGCN [18] specifically for construction site conditions. Unlike DCRNN [17] and Graph WaveNet [25], STGCN provides sufficient prediction accuracy with fewer parameters and lower computational costs, which is important for local intelligent systems with limited resources. The use of MAPPO [15] as a multi-agent component is also justified due to the stability of learning and the ability to work in conditions of decentralized execution without constant communication between agents.

However, this does not mean that the selected components are universal for all possible scenarios. If the priority is not incrementality, but the quality of the path under conditions of uncertainty of edge weights, then ARA* [29] may be promising, which finds an acceptable, but not necessarily optimal path immediately and improves it if time permits. The use of ARA* is appropriate in cases where it is necessary to maximize the speed of the algorithm, and not to guarantee the optimality of the route at each moment of time.

Graph WaveNet is appropriate to consider when scaling the system to large construction complexes, industrial parks or airports with hundreds of nodes and non-obvious long-range spatial dependencies between sites. In such conditions, the adaptive adjacency matrix of Graph WaveNet is able to detect correlations that do not directly follow from the topology of the road graph. At the same time, for a local construction site, such complexity may be excessive, especially if the system has to work with limited computing resources.

QMIX is appropriate to consider in scenarios where agents are strictly cooperative and understanding the impact of each agent on the common result is important. If the vehicle fleet is homogeneous and all decisions are made centrally during shift planning, QMIX can demonstrate higher selective learning efficiency. On the other hand, MAPPO is a better choice in the case of a heterogeneous fleet or the need for decentralized execution without stable communication between agents.

Among the main limitations of this study, the use of a synthetic construction site instead of a real production environment should be highlighted. This means that the results obtained are correct within the modeled transport network, given crisis scenarios and accepted assumptions regarding the behavior of vehicles. Another limitation is the projection nature of the evaluation of the system's performance, since full-scale implementation on a real construction site was not carried out. Another limitation is the number of crisis scenarios used during the experimental study. Therefore, the reproducibility of the claimed effects requires the preservation of similar modeling conditions, in particular the graph structure, the range of edge weights, the number of vehicles and the nature of temporary overlaps.

Separately, it is worth noting the shortcomings of the study, which are not identical to its limitations. The current implementation of the system does not take into account the degradation of the quality of sensor data, the priority of individual categories of vehicles and interaction with real BIM models and construction equipment dispatching systems. In addition, the issue of system behavior with simultaneous degradation of communication, changes in positioning accuracy and an increase in the number of vehicles was not fully considered within the framework of this study. These shortcomings can be eliminated by expanding the experimental model, connecting real telemetry streams and testing the system on real construction site data.

Further development of the study may consist in integrating the system with real telemetry streams of RTK-GNSS, UWB and IoT sensors, testing on real construction sites, expanding the forecasting model to take into account communication degradation and exploring the possibility of using Graph WaveNet [25] for large construction complexes with complex topology. Along the way, methodological, experimental and computational difficulties may arise. Methodological difficulties are associated with the need to formalize the heterogeneous constraints of the construction site in a single environment model. Experimental difficulties are associated with obtaining representative data from real sites and reproducing crisis scenarios without violating the safety of the production process. Computational difficulties may arise when scaling the system to a larger number of agents, increasing the detail of the spatial representation and simultaneously using more complex forecasting models.

7. Conclusions

1. The choice of a two-level hexagonal H3 model and a road graph for representing the traffic environment, STGCN for predicting the state of the transport network, MARL for forming route corridors and D* Lite for route planning has been justified. The combination of these components provides adaptive routing in real time and creates the prerequisites for the scalability and explainability of the route formation process.

2. An architectural model of a specialized intelligent information system for managing transport logistics of a construction site has been developed, built on event-driven microservices using Apache Kafka, Apache Flink, STGCN and MAPPO. It has been shown that the modular nature of the architecture makes it possible to replace individual system blocks in the event of additional requirements or restrictions.

3. An experimental comparison of the proposed approach with analogues based on A* and STGCN has been carried out on a simulation model of the logistics environment of a construction site in the SUMO environment. The results of the analysis of CDF curves and the 95th percentile confirmed the statistical dominance of the proposed approach by the criteria of route speed and stability of operation in the basic and crisis scenarios.

Conflict of interest

The authors declare that they have no conflict of interest in this study, whether financial, personal, authorial or otherwise, which could affect the study and its results presented in this article.

Financing

The study was conducted without financial support.

Data availability

Data will be provided upon reasonable request.

Use of artificial intelligence

In the process of preparing the article, the authors used artificial intelligence tools: ChatGPT, the GPT-5.5 Thinking model, and Claude Opus 4.7. Artificial intelligence tools were used as auxiliary tools and did not replace the author's analysis, problem statement, interpretation of results, and formulation of conclusions.

In the section "Literature review and problem statement" section, the GPT-5.5 Thinking model was used to search for scientific sources using keywords and criteria specified by the authors. In particular, the search was performed on the topics of intelligent transport logistics management systems, BIM-oriented logistics systems, digital twins, IoT platforms, spatiotemporal graph neural networks, multi-agent reinforcement learning, and route formation algorithms. The authors additionally set criteria for selecting sources, including the availability of open access, relevance to the research topic, and relevance of publications.

In the "Materials and methods" section, the Claude Opus 4.7 model was used to visualize the road graph previously generated by the authors.

In the "Experimental evaluation of the effectiveness of finding the final route" section, the Claude Opus 4.7 model was used to construct graphs of the cumulative distribution function according to the data obtained as a result of the experiment.

The results obtained using artificial intelligence tools were manually checked by the authors. For sources found using AI, the availability of the publication, the correspondence of its content to the research topic, the correctness of bibliographic data, the relevance of the conclusions and the correspondence of the citation to the actual content of the source were checked. For visualizations, the correspondence of figures, graphs, diagrams and tables to the original author's data, the correctness of numerical values, captions and notations were checked.

The use of artificial intelligence tools did not affect the scientific conclusions of the study. The final selection of sources, interpretation of literary data, formulation of the problem, choice of methods, analysis of experimental results and formulation of conclusions were performed by the authors independently. The results provided by artificial intelligence tools were used only to assist in the search for sources, structuring the material and visual presentation of the author's data.

Authors' contributions

Pavlo Pasieka: Conceptualization, Methodology, Software, Formal analysis, Investigation, Validation, Visualization, Writing – original draft, Project administration; **Oleksii Panko**:

Validation, Resources, Writing – review & editing; **Natalia Poltorachenko**: Formal analysis, Validation, Writing – review & editing; **Svitlana Terenchuk**: Supervision, Methodology, Validation, Writing – review & editing; **Bohdan Yeremenko**: Methodology, Validation, Resources, Writing – review & editing.

References

1. Nguyen, H. A. D., Ha, Q. P. (2022). Robotic autonomous systems for earthmoving equipment operating in volatile conditions and teaming capacity: a survey. Robotica, 41 (2), 486–510. https://doi.org/10.1017/s0263574722000339
2. Mazurenko, R., Yeremenko, B. (2024). Intelligent traffic management system of a big city: ontology concept "decision models". Management of Development of Complex Systems, 57, 174–180. https://doi.org/10.32347/2412-9933.2024.57.174-180
3. Bortolini, R., Formoso, C. T., Viana, D. D. (2019). Site logistics planning and control for engineer-to-order prefabricated building systems using BIM 4D modeling. Automation in Construction, 98, 248–264. https://doi.org/10.1016/j.autcon.2018.11.031
4. Whitlock, K., Abanda, F. H., Manjia, M. B., Pettang, C., Nkeng, G. E. (2021). 4D BIM for Construction Logistics Management. CivilEng, 2 (2), 325–348. https://doi.org/10.3390/civileng2020018
5. Jiang, Y., Li, M., Li, M., Liu, X., Zhong, R. Y., Pan, W., Huang, G. Q. (2022). Digital twin-enabled real-time synchronization for planning, scheduling, and execution in precast on-site assembly. Automation in Construction, 141, 104397. https://doi.org/10.1016/j.autcon.2022.104397
6. Choudhary, A. (2024). Internet of Things: a comprehensive overview, architectures, applications, simulation tools, challenges and future directions. Discover Internet of Things, 4 (1). https://doi.org/10.1007/s43926-024-00084-3
7. Tuhaise, V. V., Tah, J. H. M., Abanda, F. H. (2023). Technologies for digital twin applications in construction. Automation in Construction, 152, 104931. https://doi.org/10.1016/j.autcon.2023.104931
8. Le, T., Fan, R. (2023). Digital Twins for Logistics and Supply Chain Systems: Literature Review, Conceptual Framework, Research Potential, and Practical Challenges. https://doi.org/10.2139/ssrn.4516184
9. de Carvalho, J. P., Dimitrakopoulos, R. (2021). Integrating Production Planning with Truck-Dispatching Decisions through Reinforcement Learning While Managing Uncertainty. Minerals, 11 (6), 587. https://doi.org/10.3390/min11060587
10. Yeremenko, B., Mazurenko, R., Stetsyk, O., Buhrov, A. (2023). Intelligent Management of Traffic Flows in Large Cities. TRANSBALTICA XIII: Transportation Science and Technology, 33–42. https://doi.org/10.1007/978-3-031-25863-3_4
11. Lin, K., Zhao, R., Xu, Z., Zhou, J. (2018). Efficient Large-Scale Fleet Management via Multi-Agent Deep Reinforcement Learning. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 1774–1783. https://doi.org/10.1145/3219819.3219993
12. Duszak, P., Siemiątkowska, B., Więckowski, R. (2021). Hexagonal Grid-Based Framework for Mobile Robot Navigation. Remote Sensing, 13 (21), 4216. https://doi.org/10.3390/rs13214216
13. Zhang, Z., Zhang, J., Zhao, Y., Zhu, Z., Guo, J. (2024). A novel hexagonal grid map model and regenerated heuristic factor based strategy for intelligent manufacturing system's AGV path planning problem solving. Computers & Industrial Engineering, 192, 110154. https://doi.org/10.1016/j.cie.2024.110154
14. Schrader, M., Kumar, N., Collignon, N., Sørig, E., Yoon, S., Srivastava, A. et al. (2023). Modelling the performance of delivery vehicles across urban micro-regions to accelerate the transition to cargo-bike logistics. arXiv. https://doi.org/10.48550/arXiv.2301.12887
15. Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. Numerische Mathematik, 1 (1), 269–271. https://doi.org/10.1007/bf01386390
16. Hart, P., Nilsson, N., Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. IEEE Transactions on Systems Science and Cybernetics, 4 (2), 100–107. https://doi.org/10.1109/tssc.1968.300136
17. Stentz, A. (1995). The focussed D* algorithm for real-time replanning. In IJCAI'95: Proceedings of the 14th international joint conference on Artificial intelligence, 1652–1659. Available at: https://dl.acm.org/doi/10.5555/1643031.1643113
18. Koenig, S., Likhachev, M., Furcy, D. (2004). Lifelong Planning A*. Artificial Intelligence, 155 (1-2), 93–146. https://doi.org/10.1016/j.artint.2003.12.001
19. Koenig, S., Likhachev, M. (2002). D*Lite. Eighteenth national conference on Artificial intelligence, 476–483. Available at: https://dl.acm.org/doi/10.5555/777092.777167
20. Busoniu, L., Babuska, R., De Schutter, B. (2008). A Comprehensive Survey of Multiagent Reinforcement Learning. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 38 (2), 156–172. https://doi.org/10.1109/tsmcc.2007.913919
21. Yu, C., Velu, A., Vinitsky, E., Gao, J., Wang, Y., Bayen, A., Wu, Y. (2022). The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games. Advances in Neural Information Processing Systems 35, 24611–24624. https://doi.org/10.52202/068431-1787
22. Yu, J., LaValle, S. (2013). Structure and Intractability of Optimal Multi-Robot Path Planning on Graphs. Proceedings of the AAAI Conference on Artificial Intelligence, 27 (1), 1443–1449. https://doi.org/10.1609/aaai.v27i1.8541
23. Rashid, T., Samvelyan, M., de Witt, C. S., Farquhar, G., Foerster, J., Whiteson, S. (2018). QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. arXiv. https://doi.org/10.48550/arXiv.1803.11485
24. Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., Mordatch, I. (2017). Multi-agent actor-critic for mixed cooperative-competitive environments. arXiv. https://doi.org/10.48550/arXiv.1706.02275

25. Li, Y., Yu, R., Shahabi, C., Liu, Y. (2018). Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. arXiv. https://doi.org/10.48550/arXiv.1707.01926
26. Yu, B., Yin, H., Zhu, Z. (2018). Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, 3634–3640. https://doi.org/10.24963/ijcai.2018/505
27. Wu, Z., Pan, S., Long, G., Jiang, J., Zhang, C. (2019). Graph WaveNet for Deep Spatial-Temporal Graph Modeling. Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, 1907–1913. https://doi.org/10.24963/ijcai.2019/264
28. Zhu, J., Wang, Q., Tao, C., Deng, H., Zhao, L., Li, H. (2021). AST-GCN: Attribute-Augmented Spatiotemporal Graph Convolutional Network for Traffic Forecasting. IEEE Access, 9, 35973–35983. https://doi.org/10.1109/access.2021.3062114
29. Likhachev, M., Gordon, G., Thrun, S. (2003). ARA*: anytime A* with provable bounds on sub-optimality. NIPS'03: Proceedings of the 17th International Conference on Neural Information Processing Systems. Available at: https://dl.acm.org/doi/10.5555/2981345.2981441