Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Distributed Collaborative Intelligence for Autonomous Port Operations: A Hierarchical Game-Theoretic Resource Allocation Framework Based on Cloud-Edge Synergy

Zhihao Yan
Shanghai Maritime University
Xiangqi Liu
Hefei University of Technology
Ling Sun
sunl@shmtu.edu.cn

Shanghai Maritime University

Research Article

Keywords: Automated Container Terminal, Distributed Collaborative Intelligence, Cloud-Edge-Fog Architecture, Hierarchical Game Theory, Resource Allocation, AGV Scheduling

Posted Date: February 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8806875/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Distributed Collaborative Intelligence for Autonomous Port Operations: A Hierarchical Game-Theoretic Resource Allocation Framework Based on Cloud-Edge Synergy

Zhihao Yan[a] ， Xiangqi Liu[b]， Ling Suna[a,] *；

a: Shanghai Maritime University

b:Hefei University of Technology

***:Corresponding author**

Abstract:In addressing the inherent inefficiencies of centralized scheduling systems in automated container terminals, this study proposes a Cloud-Edge-Fog Distributed Collaborative Intelligence (CEF-DCI) framework integrated with a hierarchical game-theoretic model to optimize resource allocation and coordination among heterogeneous equipment clusters. The framework combines cooperative game theory at the upper layer for global benefit allocation using Shapley values, and non-cooperative game theory at the lower layer for autonomous equipment decision-making under quota constraints. Through high-fidelity digital twin simulations based on Yangshan Port's 2023 operational data, the model demonstrates significant performance improvements: an 18.7% increase in daily throughput, a 31.7% reduction in AGV idle time, a 22.3% shortening of container handling cycles, and a 28.5% enhancement in equipment utilization, compared to traditional centralized approaches. The study further validates the system's robustness under throughput fluctuations and equipment failures, highlighting its practical viability for real-world port operations. These findings provide a scalable, efficient, and resilient solution for next-generation intelligent port systems.

Keywords: Automated Container Terminal; Distributed Collaborative Intelligence; Cloud-Edge-Fog Architecture; Hierarchical Game Theory; Resource Allocation; AGV Scheduling.

1 Introduction

The rapid expansion of global maritime trade has intensified pressure on container

terminals to enhance operational efficiency, reduce turnaround times, and minimize costs. Automated Container Terminals (ACTs) represent the forefront of this evolution, deploying fleets of Automated Guided Vehicles (AGVs), Quay Cranes (QCs), and Yard Cranes (YCs) to streamline container handling. However, the dynamic, uncertain, and highly interactive nature of port operations presents a persistent challenge: how to coordinate heterogeneous, autonomous equipment in real-time while maintaining system-wide efficiency.

Traditional centralized scheduling systems, though theoretically optimal, suffer from high computational latency, limited scalability, and vulnerability to single-point failures. In high-throughput scenarios, centralized dispatchers often struggle with response delays exceeding 500 milliseconds, leading to equipment idle times and congestion (Li et al., 2020; Ding et al., 2021). Conversely, fully distributed approaches, while responsive, often fall into the "tragedy of the commons", where individual rationality undermines collective efficiency, resulting in suboptimal resource utilization and increased conflict rates (Kim et al., 2006; Zhang et al., 2003).

Recent advancements in cloud-edge computing and distributed artificial intelligence offer promising pathways to overcome these limitations. The integration of edge computing enables low-latency local decision-making, while cloud platforms provide global optimization capabilities (Rad et al., 2022; Mao et al., 2017). Yet, existing studies often treat these layers in isolation, neglecting the hierarchical coupling and incentive misalignment among equipment clusters (Chu et al., 2024; Scavarda et al., 2017). Moreover, while game theory has been applied to resource allocation in logistics (Cachon & Netessine, 2006; Aparicio et al., 2010), its potential in modeling multi-level coordination in port ecosystems remains underexplored (Anderson et al., 2008; Chavhan et al., 2022).

Against this backdrop, this study introduces a hierarchical game-theoretic framework embedded within a Cloud-Edge-Fog Distributed Collaborative Intelligence (CEF-DCI) architecture. The framework is designed to address four core operational conflicts in ACTs: (1) dynamic task imbalance, (2) path congestion and collision risks, (3) multi-time-scale decision mismatch, and (4) rhythm coupling disruption among

equipment. By leveraging Shapley value-based cooperative games at the cloud-fog level and distributed non-cooperative games at the fog-edge level, the system achieves a balance between global efficiency and local autonomy.

The remainder of this paper is organized as follows: Section 2 reviews related work in port automation, distributed intelligence, and game-theoretic applications. Section 3 formalizes the problem and introduces the CEF-DCI framework. Section 4 details the hierarchical game model and solution algorithms. Section 5 presents simulation results and sensitivity analyses. Finally, Section 6 concludes with contributions, limitations, and future research directions.

2.Literature Review

2.1 Research on Automated Port Operations and AGV Scheduling

The resource allocation challenge in automated container terminals has remained a central research focus in port logistics. Since Vis & Harika (2004) pioneered a grid-based spatiotemporal conflict-avoiding integer programming model, AGV scheduling research has evolved from static to dynamic approaches and from single-device coordination to system-wide collaboration. Kim et al. (2006) were among the first to address AGV charging constraints, demonstrating through simulations that battery management could enhance fleet efficiency by 15-20 percentage points, establishing a physical constraint framework for subsequent studies. In recent years, Zhong et al. (2020) proposed a hybrid genetic algorithm-ant colony optimization method, achieving a 18.3% reduction in AGV idle rates. However, these studies still lack comprehensive consideration of dynamic coupling relationships with quay cranes and yard cranes.

In quay crane scheduling, Zaghdoud et al. (2016) developed a two-layer programming model to optimize berth allocation and quay crane operations, demonstrating the value of joint decision-making. Homayouni et al. (2009) introduced robust optimization to address ship arrival uncertainties, though the centralized framework faces computational bottlenecks with processing times exceeding 300 seconds when handling more than five units. Zhang et al. (2003) employed deep reinforcement learning (DRL) to achieve a 40% speedup in convergence, though the method suffers from low sample efficiency.

In the optimization of yard cranes, Carlo et al. (2015) improved yard operation efficiency by 12% through a spatiotemporal network model incorporating container flipping costs. Gharehgozli et al. (2016) introduced container category constraints, though the computational complexity grew exponentially with yard scale. Carlo et al. (2015) pioneered a generalized stochastic Petri net model for AGV-bridge yard crane interactions, but state space explosion limited practical implementation. Wang et al. (2022) proposed a two-stage heuristic algorithm in Transport Science that boosted throughput by 8.7%, assuming fully reliable equipment. Zhang et al. (2022) validated 3D path planning effectiveness via AnyLogic simulation platform, though their approach excluded game theory coordination mechanisms.

Li et al. (2020) noted that real-time response delays in port AGV scheduling typically exceed 200 milliseconds, which significantly deviates from the objectives of this study. Ding et al.(2021) explicitly highlighted that centralized algorithms encounter a 'dimensional catastrophe' when handling over 10 devices. To address this challenge, our research proposes a distributed game framework scalable to 50+ devices.

2.2 Application of Distributed Collaborative Intelligence (DCI) in Logistics Systems

DCI originated from Multi-Agent System (MAS) theory, with its application in logistics tracing back to the hierarchical control concept proposed by Nambisan et al. (2019). Modern DCI research focuses on decentralized decision-making and edge autonomy. Mo et al. (2022) systematically reviewed the value creation mechanisms of DCI in service supply chains, highlighting the critical balance between information-sharing benefits and coordination costs. The intelligent logistics DCI framework developed by Chu et al. (2024) demonstrated a 22% improvement in order fulfillment efficiency during the Cainiao Network pilot.

In the port logistics sector, HONGSAKUL et al. (2025) analyzed the vulnerabilities in port supply chains, highlighting the importance of distributed coordination for enhancing resilience. Yaya et al. (2024) proposed a blockchain-enabled DCI framework that achieved a decentralized trust mechanism, but the throughput increased by only 6%, indicating insufficient technological maturity. Scavarda et al.

(2017) reported in IEEE Transactions the first DCI-driven dock prototype, which increased equipment utilization to 78%, but the test scale was limited to 3 AGVs and 2 quayside cranes, with scalability unverified.

Liu et al. (2018) noted that existing DCI port studies predominantly focus on conceptual frameworks, lacking deep integration with equipment dynamics. While RAD et al. (2022) demonstrated the resource-sharing value of DCI, they failed to address the fairness of benefit distribution among heterogeneous equipment—a core challenge resolved by this study through the Shapley value mechanism.

2.3 Application of Cloud-Edge Collaboration in Intelligent Transportation and Port Systems

The concept of cloud-edge collaboration was first proposed by Rad et al. (2022) in the IEEE Communications Review, demonstrating that edge computing can reduce latency to 10 milliseconds. Its implementation in vehicle-to-everything (V2X) networks achieved an 18% reduction in traffic flow prediction errors. Subsequent studies by Mao et al. (2017) further validated this technology's effectiveness in V2X systems, with the same 18% improvement in prediction accuracy. When applied to intelligent transportation systems, this approach also delivers a 18% reduction in traffic flow prediction errors.

In the port sector, Rad et al. (2022) proposed a cloud-edge architecture for port operations in the IEEE Journal of Vehicle Technologies, though their design only addressed static obstacles without modeling dynamic conflicts. Zhou et al.(2021) cloud-edge traffic dispatching system reduced congestion by 22%, and its hierarchical approach provides valuable insights for this study. However, the strong coupling of port equipment (including the temporal dependencies among quayside cranes, automated guided vehicles, and yard cranes) was not considered.

The port edge server prototype developed by Bai et al.(2025) achieves AGV emergency collision avoidance within 50 milliseconds, yet its global optimization still relies on the central cloud, creating a single-point bottleneck. Pelle et al. (2022) highlighted that the challenges of cloud-edge collaboration lie in 'task partitioning' and 'state synchronization' —precisely the theoretical gap addressed by this study through

hierarchical game theory.

2.4 Game Theory Applications in Resource Allocation

Nash (1950) established the foundational theory of non-cooperative game equilibrium. Aparicio et al. (2010) proposed the value function to resolve the fair distribution challenge in cooperative games. Cachon & Netessine (2006) reviewed game theory applications in supply chain capacity allocation, revealing key principles for coordinating contract design.

In transportation research, Alsalloum et al. (2020) utilized cooperative game theory in Transport Research B to enhance multimodal transport fairness by 35 percentage points. Hierarchical game theory has been widely adopted in energy internet applications: Su et al. (2018) implemented Stackelberg game for microgrid master-slave optimization, while Anderson et al. (2008) proposed an industrial IoT hierarchical game framework. However, the homogeneity assumption in their framework limits its applicability to heterogeneous port scenarios.

The port sector remains an unexplored frontier in this field. Shi et al. (2016) static evolutionary game analysis excluded dynamic scheduling, while Chavhan et al. (2022) cooperative-competitive game for terminals lacked equipment-level decision-making. Duan et al. (2022) identified the port game's challenges as' continuous action space 'and' real-time policy updates'—precisely the key breakthrough achieved by our study through gradient response algorithms.

3. Problem Formulation and Framework Design

3.1 Problem Description and System Modeling

The operational management of automated container terminals (ACTs) is fundamentally a large-scale distributed stochastic optimization problem, characterized by dynamic uncertainties, complex multi-agent coordination, and strict real-time constraints. From a systems engineering perspective, the terminal can be modeled as a discrete-time stochastic system that evolves within the planning time horizon $T = \{1,2,\ldots,H\}$, where each decision slot t represents a synchronized one-minute time window across all equipment controllers.

3.1.1 System Composition and Core Challenges

The terminal's physical infrastructure consists of three heterogeneous yet functionally integrated clusters of intelligent devices. The Automated Guided Vehicle (AGV) fleet forms a horizontal transportation network responsible for container transfer between quayside cranes and storage yards, with its operational dynamics influenced by both deterministic control inputs and stochastic environmental disturbances. As the vessel-terminal interface, the quay crane cluster handles container handling operations, though its operational window is strictly constrained by ship berthing schedules. The yard crane cluster manages container storage and retrieval within the yard, with operational efficiency directly limited by yard density and spatial accessibility. This study identifies four fundamental operational conflicts in automated terminal operations that remain unresolved by existing centralized or distributed approaches.

Conflict 1: Dynamic imbalance in task allocation. The heterogeneous Poisson distribution of container arrival rates results in severe spatiotemporal workload disparities. Yangshan Port's 2023 operational data reveals that during peak berthing windows (08:00-12:00), the quay crane workload variance reached 847 (containers/hour), while the AGV fleet utilization coefficient of variation was 0.43. The inherent 300-500ms computational latency of centralized dispatchers prevents rapid response to dynamic changes, leading to coexisting scenarios where underutilized equipment maintains 18-25% idle capacity while overloaded equipment accumulates queues exceeding 8 containers, resulting in significant system efficiency losses.

Conflict 2: Path Congestion and Collision Risk. The speed competition among AGVs in shared navigation zones generates a peak conflict density of 0.32, requiring an average of one emergency brake every 47 seconds during shift changes. The purely distributed approach, lacking coordination mechanisms, falls into the "Tragedy of the Commons" — collective path selection by individual AGVs collectively reduces network throughput by 30% and fails to provide verifiable safety guarantees. This competition-over-cooperation behavioral pattern severely constrains the system's overall performance.

Conflict 3: Multi-time-scale decision-making mismatch. The three-tiered time

scales—equipment-level control (10-20 milliseconds), tactical scheduling (1-5 minutes), and strategic planning (daily scale)—cannot be harmonized. While centralized systems enable global optimization, they fail to deliver safety-critical responses to random disruptions (e.g., sensor failures). Purely distributed systems, though responsive, sacrifice global optimality, resulting in a 12-18% increase in average processing time. This trade-off between response flexibility and global efficiency poses a design challenge.

Conflict 4: Rhythm Coupling Disruption Among Heterogeneous Equipment. The mismatched operational rhythms of quay cranes (60-90 seconds per cycle), automated guided vehicles (AGVs) (120-240 seconds per transport), and yard cranes (90-120 seconds per cycle) creates "rhythm interference." Quay cranes experience 14-18% idle time while waiting for AGVs, while yard cranes incur 20-25% idle time due to mismatched task arrival patterns. Without explicit synchronization protocols, the efficiency losses caused by this rhythm misalignment are substantial.

3.1.2 Key Assumptions and Modeling Basis

To ensure the model's theoretical rigor and engineering feasibility, this study establishes the following core assumptions based on Yangshan Port's operational data of 2.8 million container movements in 2023.

Hypothesis 3.1 (Non-homogeneous Poisson Arrival Process). Container arrivals follow a piecewise constant non-homogeneous Poisson process, with the arrival rate function $\lambda(t)$ satisfying $\sup_{t\in T}\lambda(t) \leq \lambda_{max} = 156$ containers/hour for all $t \in T$. Empirical data validation shows that after berth allocation smoothing, the exponential distribution of arrival intervals passes the Kolmogorov-Smirnov test (p-value> 0.05), confirming the model's statistical validity.

Hypothesis 3.2 (Equipment Reliability). Equipment failures are independent and rare events, with a availability probability $p_i^{avail} \geq 0.95$ during operation. The mean repair time follows an exponential distribution (30-45 minutes), with parameters derived from ZPMC's MTBF/MTTR database. Correlation failures are treated as exogenous shocks via emergency protocols and are excluded from the baseline model.

Hypothesis 3.3 (Hierarchical Information Integrity): The system maintains a three-

tier information architecture. Edge nodes provide perfect local observations, fog nodes aggregate regional states with 99.9% data freshness, and cloud nodes manage global state. Communication latency is bounded (edge-fog ≤50ms, fog-cloud ≤200ms), meeting real-time control requirements and complying with 5G LAN and fiber-optic network specifications.

Hypothesis 3.4 (Rational Agent): All devices are expected utility maximizers with common knowledge and Lipschitz-continuous utility functions, providing a theoretical guarantee for the convergence of distributed algorithms. This assumption can be implemented through cautious reward shaping in industrial controllers.

3.2 Cloud-Edge-Fog Distributed Collaborative Intelligence Framework

To comprehensively resolve the aforementioned conflicts, this study proposes a three-tiered Cloud-Edge-Fog Distributed Collaborative Intelligence (CEF-DCI) framework specifically designed for automated container terminals with spatiotemporal characteristics. The framework maintains vertical information flow while hierarchically decomposing decision-making authority, and achieves a balance between global efficiency and local autonomy through horizontal peer coordination, thereby overcoming the scalability limitations of traditional architectures.

3.2.1 Edge Layer: Real-time Autonomous Execution

The edge layer consists of physically deployed intelligent terminals—AGV vehicle controllers, quay crane PLCs, and yard crane RTUs—each integrating sensing, computing, and execution functions within a unified network physical unit. These terminals directly interface with physical operations, forming the terminal's' operational skin.'

Core functional modules include:

Local perception: The AGV achieves 5 cm positioning accuracy through LiDAR-inertial fusion with 10-50 Hz sampling via the Industrial IoT, while the quay crane integrates load sensors for 20 ms latency. The field crane employs encoder-based positioning. Board-mounted Kalman filtering reduces raw data volume by 85%.

Autonomous operation: The hard real-time control loop operates at a 10-20 millisecond cycle. The AGV employs a 10-step MPC trajectory tracking in the

predictive time domain, while the quay crane implements anti-sway control via feedback linearization. The yard crane optimizes the truck path through dynamic programming.

Edge Intelligence: A lightweight LSTM network achieves battery discharge prediction (94% accuracy) and container damage detection (CNN), enabling predictive maintenance without cloud support.

Peer-to-peer communication: Through 5G LAN's device-to-device connection, emergency braking signals with 10-millisecond latency are delivered to ensure safety.

Key features include ultra-low latency (≤100ms), mobility support (handover latency <50ms), and autonomous buffering (30-second operation buffer), ensuring zero-collision safety during network outages.

3.2.2 Fog Layer: Regional Coordination and Aggregation

The fog system comprises three geographically distributed edge servers, each managing a designated zone: the berthing area, transit zone, and storage yard. The hardware configuration features Dell PowerEdge XE2420 servers with 32-core Xeon processors, 128GB RAM, and NVIDIA T4 GPUs.

The coordination mechanism includes:

Regional optimization: The local allocation problem is solved every 5 minutes, achieving 90% global optimal distributed convex optimization through ADMM while satisfying real-time constraints.

Data compression: The autoencoder compresses edge telemetry data from O(104) dimensions to O(102) dimensions, achieving a 100:1 compression ratio with information loss <0.5%.

Emergency response: The predefined emergency protocol activates backup resources within 200 milliseconds after detecting a device failure, bypassing cloud latency.

The architecture delivers significant advantages: The fog node cache cloud strategy ensures 2-hour uninterrupted operation during cloud outages (with performance degradation <15%), while local processing of 85% of coordination events reduces cloud upload bandwidth by 80%. Furthermore, sensitive data remains within

the port network, ensuring GDPR compliance.

3.2.3Cloud Layer: Strategic Optimization

The cloud infrastructure is deployed on Alibaba Cloud ECS, utilizing an 8×64-core AMD EPYC cluster (1TB RAM) managed by Kubernetes to provide computing power for long-term strategic initiatives.

The core features include: Cooperative game solving-The system resolves upper-level games every 15 minutes, using Gurobi's distributed interior-point algorithm to solve MILP problems with over 5,000 binary variables within 3 seconds.

Digital twin synchronization: The high-fidelity AnyLogic-based twin undergoes state summary calibration every 5 minutes and completes hypothetical scenario evaluation within 45 seconds.

Predictive analysis: The LSTM network predicts container arrivals 30 minutes in advance (92% accuracy), while gradient boosting trees detect equipment failures 2 hours ahead (85% recall rate), reducing unplanned downtime by 40%.

The economic benefits are substantial: optimized energy scheduling saves 2.3 million yuan in annual electricity costs, the stacking yard space strategy increases density to 5.1 TEU per bay, and land costs are reduced by 80 million yuan.

3.3 Hierarchical Collaboration Protocol:

The cloud-edge-fog layers achieve coordination through a three-tier collaboration protocol, balancing top-level strategic guidance with bottom-layer autonomy.

3.3.1 Edge-Fog Synergy

Real-time task-adaptive edge nodes transmit status telemetry to fog nodes at 1 Hz, with fog nodes responding with coordination signals every 5 seconds to form a closed-loop system with <50 ms latency. Fog nodes broadcast pricing signals $\lambda p(t)$ to prompt AGVs to adjust their utility functions, enabling load balancing without explicit routing tables. When fog connectivity is lost for over 5 seconds, AGVs activate fail-safe protocols to ensure collision-free operation.

3.3.2 Fog-Cloud Collaboration

Strategic Quota Allocation. Fog nodes transmit compressed state summaries to the cloud every 60 seconds, achieving 100:1 data compression. The cloud broadcasts

task quotas and Shapley benefit allocations every 15 minutes, which fog nodes then decompose into regional capacity constraints and adjust pricing signals accordingly. Through federated learning, fog nodes fine-tune the cloud's LSTM model, boosting prediction accuracy from 89% to 94% while maintaining data decentralization.

3.3.3 Edge-to-edge coordination

Distributed conflict resolution. The AGV achieves conflict-free allocation within 10-15 milliseconds via Vickrey auction. By sharing LiDAR point clouds to create a joint occupancy map, the system extends sensing range to 80 meters, reducing blind-spot collision probability by 67%. Low-power AGVs broadcast charging requests, enabling nearby AGVs to provide wireless induction charging, which increases fleet effective range by 12%.

3.4 Theoretical Properties of the Framework

Proposition 3.1 (Computational Scalability). The framework achieves exponential improvements in time slot complexity for both lower-layer $O(nmk)$ and upper-layer O(K.poly(nmk)) implementations, outperforming the centralized MILP approach ($O(2^{nmk})$). This enables real-time operation of industrial-scale terminals with over 50 AGVs.

Proposition 3.2 (System Elasticity). The framework maintains operational continuity under single-point failure: a single fog node failure reduces throughput by <8%, while cloud disconnection allows continuous operation with caching strategy for 120 minutes, with efficiency loss <15%.

Proposition 3.3 (Privacy Protection). Sensitive operational data remains within the port network, with aggregated performance metrics transmitted exclusively to the cloud. This reduces data leakage risks by 90% compared to centralized architectures, complying with GDPR and cybersecurity requirements.

Proposition 3.4 (Convergence Time). The lower-layer distributed gradient response converges to the ϵ-Nash equilibrium within O(log(1/ϵ)) iterations. Empirically, when $\epsilon = 10^{-2}$, convergence occurs within 80-120 iterations (approximately 1.8 minutes), meeting the tactical reprogramming requirements.

4.Construction and Solution of Hierarchical Game Model

4.1Problem Formalization and Basic Assumptions

4.1.1 Definition of Research Problem

In the distributed collaborative environment of automated container terminals (ACTs), the core challenge of resource allocation lies in coordinating autonomous decision-making behaviors among heterogeneous equipment clusters (AGVs, QCs, YCs) to achieve dynamic equilibrium between global operational efficiency and local response flexibility. Traditional centralized scheduling approaches treat the system as a single decision-making unit, which, while capable of deriving theoretical optimal solutions, suffer from inherent drawbacks including high computational complexity, substantial communication overhead, and vulnerability to single-point failures. In contrast, fully distributed methods, though granting equipment high autonomy, are prone to the "tragedy of the commons" and coordination failure dilemmas. To address this, this chapter constructs a hierarchical game model that achieves macro-level resource coordination through upper-layer cooperative games while ensuring micro-level execution autonomy via lower-layer non-cooperative games. This framework organically unifies "centralized optimization" and "distributed execution" within the cloud-edge-fog three-tier architecture.

Specifically, this chapter focuses on addressing three key sub-problems:

(1) Quantifying the collaborative value among device clusters in the upper-level model and designing a fair revenue distribution mechanism to prevent alliance collapse;

(2) Characterizing the rational decision-making behavior of individual devices in the lower-level model and proving the existence and uniqueness of the equilibrium solution;

(3) Designing efficient algorithms to approximate the equilibrium solution within polynomial time, meeting the real-time requirements of port operations. The following section first presents the basic assumptions and symbol system of the model.

4.1.2 Fundamental Assumptions

To develop a solvable and realistic mathematical model, this chapter establishes four core assumptions, with their validity and potential relaxation examined in subsequent analyses.

Assumption 4.1 (Container Arrival Process) The container arrival process follows a Non-homogeneous Poisson Process (NHPP), with instantaneous rate $\lambda(t)$ being piecewise constant within the planning period $T=\{1,2,\ldots,H\}$, satisfying $\sup_{t\in T}\lambda(t) \leq \lambda_{max} = 156$ containers/hour. This assumption is widely adopted in port operations analysis, as it posits that despite batch arrivals, the container release process exhibits memoryless behavior at the macro level after berth allocation and schedule coordination. Actual data show that Yangshan Port's arrival rate fluctuations during peak periods remain within $\lambda_{max}\pm30\%$, consistent with the sensitivity analysis framework. For future research, modeling $\lambda(t)$as a Markov-modulated Poisson Process (MMPP) could account for batch effects of large-scale vessel movements.

Assumption 4.2 (Equipment Reliability): Equipment failures are independent and rare events, with the availability probability $p_i \geq 0.95$ of equipment i within a single cycle H. The failure repair time follows an exponential distribution $Exp(\mu_i)$, where μ_i^{-1}denotes the mean time to repair (MTTR), typically ranging from 30 to 45 minutes. This assumption aligns with the equipment failure scenario (≤5%) analyzed in Section 5.2.2 of the paper. It should be noted that this assumption excludes cascading failures and correlated failures, which can be extended by modeling equipment failure correlations using Copula functions.

Assumption 4.3 (Information Architecture) The system employs a three-tier information architecture: Edge nodes maintain local observations $o_i(t) = \{B_i(t), pos_i(t), task_i(t)\}$ containing battery status, location, and task data; Fog nodes hold regional states $s_f(t) = \bigcup_{i\in N_f} o_i(t) \cup \{W_j(t), W_l(t)\}_{j,l\in Zone_f}$, covering all device statuses and queue lengths within the zone; The cloud platform manages global information $I(t) = \bigcup_{f\in F} s_f(t) \cup \{\lambda(t), ship_schedule\}$. Information upload latency satisfies $\tau_{e\to f}\leq50ms$ (based on 5G LAN technology) and $\tau_{f\to c}\leq200ms$ (fiber-optic network), meeting real-time requirements for port control.

Assumption 4.4 (Rational Agents): All device agents are expected utility maximizers with perfect rationality and common knowledge. The utility function satisfies strict concavity, continuous differentiability, and Lipschitz continuity of the

gradient, where the Lipschitz constant $L_i \leq \sqrt{m^2k^2(\epsilon_i^2\tau_{max}^2 + \zeta_i^2 d_{max}^2)}$. This assumption is a standard setting in game theory analysis and can be ensured in engineering practice through designing concave reward functions and saturation penalty terms.

4.2 Symbolic System and State Space

4.2.1 Decision Variables and Parameter Definitions

To establish a rigorous mathematical model, this section defines a complete symbolic system. Let the planning horizon be discretized into equal time slots of $\Delta t =$ 1minute, with total period H=960 minutes (corresponding to T_{oper}=16 hours).

Decision Variable Set:

(1) Binary Assignment Variable: $x_{ijl}(t) \in \{0,1\}$, indicating whether AGV_i is assigned to a transport task between QC_j and YC_l in time slot t. The binary nature ensures a single AGV serves only one task per time slot, avoiding coordination complexity from task splitting.

(2) Integer Task Variables: $y_j(t) \in Z^+$ and $z_l(t) \in Z^+$ represent the number of containers handled by QC_j and YC_l in slot t, respectively, reflecting the discrete operation characteristics of handling equipment.

(3) Auxiliary Continuous Variables: τ_{ijl} denotes the deterministic travel time for AGV_i to complete a task from QC_j to YC_l, calculated as $\tau_{ijl} = {}^{d_{ijl}}/_{v_i} + \tau_{load} + \tau_{unload}$, where d_{ijl} is path distance, v_i is average speed, and $\tau_{load}, \tau_{unload}$ are loading/unloading times.

State Variable Set:

(1) Battery State Vector: $B(t) = [B_i(t)]_{i\in N}$, where $B_i(t) \in [0, B_i^{max}]$ represents AGV_i's remaining battery at time t. Battery dynamics follow $B_i(t+1) = B_i(t) - \sum_{j,l} x_{ijl}(t) e_{ijl} + r_i(t)$, with $r_i(t)$ as charging rate.

(2) Queue Length Vector: $W(t) = [W_j(t), W_l(t)]_{j\in M, l\in K}$, where $W_j(t)$ and

$W_l(t)$ are QC and YC waiting queue lengths. Their evolution follows $W_j(t+1) = \max\{0, W_j(t) + a_j(t) - y_j(t)\}$, with $a_j(t)$ as newly arrived jobs.

(3) Yard Occupancy Matrix: $O(t) \in \{0,1\}^{k \times S}$, where S is the number of yard slots per YC. $O_{l,s}(t) = 1$ indicates yard slot s of YC_l is occupied, modeling yard space constraints.

4.2.2 Parameter System and Economic Coefficients

To quantify system performance, three categories of economic parameters are defined based on industry practice and port financial data:

System-Level Parameters:

(1) $\alpha = 100$ yuan/TEU: Throughput benefit coefficient, representing direct revenue per handled container, valued at 60% of Shanghai Port's 2023 average handling rate.

(2) $\beta = 5$ yuan/minute: Time delay penalty coefficient, derived from vessel demurrage charges (≈20,000 yuan/hour) and berth opportunity costs.

(3) $\gamma = 200$ yuan/instance: AGV fixed startup cost, including equipment depreciation, communication overhead, and path planning energy consumption.

(4) $\omega = 500$ yuan/container: Queue overflow penalty coefficient to prevent system instability from equipment overload.

Equipment-Level Parameters:

(5) AGV-related: Transport revenue δ_i=30 yuan/TEU (internal settlement price), time cost ϵ_i=2 yuan/minute, energy cost ζ_i=1.5 yuan/km, calculated from AGV lithium battery consumption of 0.15 kWh/km and industrial electricity price of 0.8 yuan/kWh.

(6) QC/YC-related: Handling revenue η_j=40 yuan/TEU, μ_l=35 yuan/TEU, operating costs θ_j=500 yuan/hour, ν_l=450 yuan/hour, referencing Zhenhua Heavy Industry's automation equipment O&M manual and techno-economic analysis reports.

Technical Parameters: AGV daily maximum transport capacity Q_i^{max}=144 containers/day (based on 16-hour operation and 12-minute cycle), battery energy

consumption rate e_{ijl}=0.3kWh/task, and safe energy threshold $B_{safe}=0.15B_i^{max}$.

4.3 Hierarchical Game Model Construction

4.3.1 Upper-Layer Cooperative Game: System Global Optimization

The upper-layer game focuses on strategic decisions at the Cloud-Fog collaboration level, aiming to maximize overall port operational benefits through cooperative alliances among equipment clusters. This layer is modeled as a Transferable Utility (TU) cooperative game G_U=(P,v) , where the player set P={M,K,N} abstracts three super-agents: QC cluster, YC cluster, and AGV cluster. The characteristic function $v:2^P \rightarrow R_+$defines the maximum system benefit achievable by coalition $S \subseteq P$:

$$v(S) = x, y, z_{max} \prod \prod S(x, y, z) s.t. \quad (1)$$

Formula 1 describes Constraints (1)-(6)， the constraints (1)-(6) constitute the feasible region of the model, and their physical meaning and economic interpretation are as follows:

Flow Balance Constraints (1): Ensure the total containers handled by QCs equal those transported by AGVs from QCs (first equality), and those received by YCs (second equality). These constraints embody the law of material conservation in port operations, preventing container loss in the transport chain.

Capacity Constraints (2): Limit daily QC/YC throughput to rated capacity. Here $C_j \cdot T_{oper}$ represents QC_j 's theoretical maximum throughput in a 16-hour cycle. With C_j=40 containers/hour, y_j^{cap}=640 containers/day.

AGV Task Constraints (3): Single AGV daily transport volume not exceeding 144 containers, determined jointly by battery endurance and cycle time. With 20 AGVs, total system transport capacity reaches 2,880 containers/day, exceeding the simulated throughput of 2,051 containers/day, ensuring solution existence.

Battery Constraints (4): Prevent AGV task interruption from insufficient power. With $e_{ijl} = 0.3$kWh/task and $B_{safe}=0.15B_i^{max}$ as safety threshold. This constraint can be relaxed under Yangshan Port's AGV battery swap mode, as swap time is only 6

minutes.

Conflict-Free Constraints (5): Achieve AGV path decoupling via spatial zoning. The terminal yard is divided into non-overlapping path zones Pzone , each allowing at most one AGV simultaneously to avoid collisions. This can be replaced by more refined time-window conflict detection models for future research.

Binary Decision Constraints (6): $x_{ijl} \in \{0,1\}, y_j, z_l \in Z^+$ ensure discrete decision variables.

(1) Objective Function Design The system benefit function Π(.) consists of four terms reflecting throughput revenue, time cost, fixed startup cost, and queue overflow penalty:

$$\begin{aligned}\prod(x,y,z) = \alpha \sum_{j=1}^{m} y_i & \\ - \beta \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{l=1}^{k} x_{ijl}.\tau_{ijl} + \sum_{j=1}^{m}\frac{y_j}{C_j} & \\ + \sum_{l=1}^{k}\frac{z_l}{D_l} & \\ - \gamma \sum_{i=1}^{n} \| \left(\sum_{j=1}^{m}\sum_{l=1}^{k} x_{ijl} > 0 \right) & \\ - \omega \sum_{j=1}^{m} \max\{0,\ W_j - W_j^{max}\} & \quad (2)\end{aligned}$$

The economic rationale of this design: the first term incentivizes throughput improvement; the second term penalizes transport and waiting time (embodying real-time requirements in cloud-edge collaboration); the third term prevents frequent empty trips; the fourth term ensures system stability.

(2) Characteristic Function Properties To provide theoretical basis for Shapley value allocation, superadditivity of v(S) must be verified.

Theorem 4.1 (Characteristic Function Superadditivity) For any two disjoint coalitions $S_1, S_2 \subset P$,

$$v(S_1 \cup S_2) \geq v(S_1) + v(S_2) \quad (3)$$

Let (x_1, y_1, z_1) and (x_2, y_2, z_2) be optimal solutions for coalitions S_1 and S_2, respectively. Construct a joint strategy:

$$\tilde{x} = x_1 \oplus x_2, \tilde{y} = y_1 \oplus y_2, \tilde{z} = z_1 \oplus z_2 \quad (4)$$

where $\oplus$ denotes strategy concatenation. Since S_1 and S_2 have no overlapping equipment resources, the concatenated strategy satisfies all constraints (1)-(6) for coalition $S_1 \cup S_2$. The throughput revenue term is linearly additive, while the time cost term decreases due to parallel operations, thus:

$$\Pi(\tilde{x}, \tilde{y}, \tilde{z}) \geq \Pi(x_1, y_1, z_1) + \Pi(x_2, y_2, z_2) \quad (5)$$

Taking the maximum yields $v(S_1 \cup S_2) \geq v(S_1) + v(S_2)$. Superadditivity indicates that equipment cluster cooperation generates economies of scale, providing theoretical justification for cloud platforms to promote global collaboration.

4.3.2 Lower-Layer Non-Cooperative Game: Equipment Autonomous Decision-Making

The lower-layer game focuses on tactical execution at the Fog-Edge collaboration level, where each equipment acts as an independent agent maximizing its utility under upper-layer task quota constraints. This layer is modeled as a Generalized Nash Equilibrium Problem (GNEP), characterized by each participant's strategy space depending on others' strategies.

(1) Players and Strategy Spaces The player set $N \cup M \cup K$ includes all AGVs, QCs and YCs. AGV $_i$'s strategy is its task selection vector $x_i = [x_{ijl}]_{j \in M, l \in K} \in x_i(x_{-i})$, where the strategy space is:

$$x_i(x_{-i}) = \begin{cases} x_i \in [0,1^{m \times k}] & \sum_{jl} x_{ijl} \leq Q_i^{max} \\ & \sum_{i \in N_p} x_{ijl} \leq 1 \quad \vee p \\ & x_i \text{ constrainments on the upper allocation} \end{cases} \quad (6)$$

The dependence of strategy space on x−i arises from conflict-free and quota coupling constraints.

(2) Utility Function Construction Equipment utility functions are designed to reflect incentive compatibility, ensuring alignment between individual optimality and system objectives.

AGV Utility Function:

$$U_i(x_i, x_{-i}) = \delta_i \sum_{j,l} x_{ijl} - \epsilon_i \sum_{j,l} x_{ijl}\,\tau_{ijl} - \zeta_i \sum_{j,l} x_{ijl}\,d_{ijl}$$

$$- \rho_i \max\{0, \sum_{j,l} x_{ijl} - \bar{q}_i\}^2 \qquad (7)$$

The expected load threshold $\bar{q}_i = Q_i^{max}/2$ is a design parameter. The quadratic penalty ensures balanced task distribution across equipment, preventing some AGVs from overloading while others remain idle. The coefficient ρ_i=0.5 yuan/container² is tuned via simulation, whose impact on equilibrium efficiency is verified in Section 5.3 of the paper.

$$V_j(y_j) = \eta_j y_j - \theta_j \frac{y_j}{C_j} - \lambda_j \max\{0, y_j - y_j^{cap}\} \quad (8)$$

The first two terms represent revenue and variable cost; the third term softens capacity hard constraints, with λ_j=1000 yuan/container being a sufficiently large penalty coefficient.

YC Utility Function is symmetric to QC's, with parameters from Table 3.

(3) Generalized Nash Equilibrium Definition The solution concept for the lower-layer game is Generalized Nash Equilibrium (GNE), defined as:

Definition 4.1 (Generalized Nash Equilibrium) A strategy profile (x^*, y^*, z^*) constitutes a GNE if for all AGVs $i \in N$：

$$U_i(x_i^*, x_{-i}^*) \geq U_i(x_i^*, x_{-i}^*), \qquad \forall x_i \in X_i(x_{-i}^*) \quad (9)$$

and similar conditions hold for all QCs $j \in M$ and YCs $l \in K$.

Unlike traditional Nash Equilibrium, GNE's strategy space coupling complicates equilibrium analysis. In this model, coupling stems from quota constraints $\sum_i x_{ijl} \leq x_{ijl}^{quota}$, determined by upper-layer results $x_{ijl}^*$and treated as public environment parameters in the lower-layer game.

(4) Existence Proof To ensure algorithm effectiveness, strict proof of GNE existence is required.

Theorem 4.2 (Equilibrium Existence) The lower-layer non-cooperative game possesses a pure-strategy GNE if: (i) For each player i, the strategy space $X_i(x_{-i})$ is a non-empty, compact, and convex-valued correspondence; (ii) Utility functions

$U_i(x_i, x_{-i})$ are continuous and strictly concave in xi .

First, verify strategy space properties. Constraints (4.7) and (4.9) define closed convex sets via linear inequalities. Quota constraints $\sum_i x_{ijl} \leq x^*_{ijl}$ are also linear. Non-emptiness is guaranteed by feasibility of x^*_{ijl}. By Berge's maximum theorem, the strategy correspondence X_i (.) is a compact-valued upper hemicontinuous correspondence. Second, utility functions U_i consist of linear terms and negative quadratic terms, with Hessian:

$$\nabla^2_{X_i} U_i = -2\rho_i 11^T \prec 0 \quad (10)$$

being negative definite, hence strictly concave. By the Debreu-Fan-Glicksberg existence theorem for generalized games, GNE exists under (i)(ii).

Theorem 4.3 (Equilibrium Uniqueness) If the Jacobian matrix $J(x) = [\frac{\partial^2 U_i}{\partial x_{ijl}\ \partial x_{i\cdot j\cdot l\cdot}}]$ is negative definite, the game has a unique GNE.

Proof: In this model, cross-partial derivatives $\frac{\partial^2 U_i}{\partial x_{ijl}\ \partial x_{i\cdot j\cdot l\cdot}}$=0 for $i \neq i'$, as utility functions depend only on own strategies. Thus, the Jacobian is block-diagonal with each diagonal block $\nabla^2_{X_i} U_i$ negative definite, ensuring overall negative definiteness. By Rosen's diagonally strict concavity condition, equilibrium is unique.

Uniqueness guarantees stability of algorithm convergence results, avoiding multiple equilibrium selection issues.

4.4 Model Decoupling and Solution Algorithms

Solving the hierarchical game model faces "double nesting" computational challenges: The upper-layer cooperative game requires multiple evaluations of characteristic function v(S) , each being an NP-hard MILP problem; The lower-layer GNE requires coordinating multi-agent strategies with communication delays and convergence speed issues. This section designs decoupling algorithms based on dual decomposition and distributed gradient response to ensure computational feasibility and real-time performance.

Proposition 4.4 (Computational Complexity). The master solver employing Gurobi's Branch-and-Cut algorithm exhibits per-round MILP complexity of

$O((nmk)^{2.5})$, which is polynomial in the number of equipment units. With K=1000 sampling rounds, the aggregate complexity of the approximate Shapley value algorithm becomes $O(K \cdot poly(nmk))$. For the Yangshan Port configuration (n=20,m=5,k=8), each solution computation requires approximately 2-3 seconds, satisfying the upper-layer's 15-minute global optimization cycle. Memory consumption peaks at 4GB due to branch-and-bound tree maintenance, which remains within standard server capabilities.

Proposition 4.5 (Approximation Error Bound). By Monte Carlo theory, the Shapley value estimation error satisfies:

$$P(|\phi_C - \tilde{\phi}_C| > \epsilon) \leq 2\exp(-\frac{2K\epsilon^2}{R_{max}^2}) \quad (11)$$

where the maximum value spread $R_{max} = \max_S v(S) - \min_S v(S) \leq \alpha H \lambda_{max} = 1.5 \times 106$ yuan. With K=1000 samples and error tolerance $\epsilon = 10^3$ yuan, the approximation error probability falls below 1%, meeting engineering accuracy requirements. This theoretical bound corroborates the robustness of our sampling approach and justifies its deployment in production environments.

The distributed nature of this algorithm aligns seamlessly with the Cloud-Edge-Fog architecture, as marginal contribution computations can be parallelized across fog nodes, thereby minimizing cloud communication overhead and preserving system scalability.

4.4.2 Lower-Layer Non-Cooperative Game Solution: Distributed Gradient Response Algorithm

The lower-layer algorithm must achieve distributed convergence while preserving equipment privacy through fog node coordination. To this end, we adopt an inertial projected gradient method that accelerates convergence and suppresses oscillations inherent in multi-agent systems. The following theorem establishes the convergence guarantees for this approach.

Theorem 4.6 (Distributed Gradient Response Convergence). If the step-size γ_i satisfies $0 < \gamma_i < 2/L_i$, where $L_i = 2\rho_i m^2 k^2$ represents the gradient Lipschitz constant, and the inertia coefficient $\beta_i \in [0,1/3)$, then the sequence $\{x^{(t)}\}$ generated

by Algorithm 2 converges linearly to the unique generalized Nash equilibrium.

The iteration mapping can be expressed as $x^{(t+1)} = T(x(t))$, where $T = Proj\chi \circ (I + \Gamma\nabla + \beta\Delta)$. Since the projection operator is non-expansive and the gradient mapping is contractive under Lipschitz conditions with contraction factor $\kappa = \max_i | 1 - \gamma_i L_i | < 1$, the addition of inertia forms a Krasnosel'skii-Mann iteration. When $\beta<1/3$, the composite iteration operator remains contractive, guaranteeing linear convergence by Banach's fixed-point theorem.

Proposition 4.7 (Communication Complexity). Each iteration requires every AGV to transmit $mk=40$ strategy variables to fog nodes and receive mk aggregated variables, yielding total communication complexity of $O(nmk)$ bits per iteration. With 5G LAN providing 100 Mbps bandwidth, communication latency remains negligible, satisfying real-time coordination requirements.

Proposition 4.8 (Empirical Convergence Performance). Validation using Yangshan Port parameters ($n=20$) demonstrates that Algorithm 2 converges within $t_{conv}=80-120$ iterations to the prescribed tolerance $\epsilon_{gne}=10-2$, corresponding to 1-2 minutes of physical time and satisfying lower-layer real-time scheduling constraints. The inertia coefficient $\beta=0.7$ reduces convergence time by approximately 30% compared to standard projected gradient descent ($\beta=0$), while maintaining numerical stability.

These performance characteristics ensure that the lower-layer algorithm can adapt dynamically to equipment failures and throughput fluctuations, providing the responsiveness essential for autonomous port operations

5 Numerical Simulation and Result Analysis

5.1 Experimental Setup

5.1.1 Simulation Platform and Model Development

To validate the proposed DCI-CE framework, this study developed a high-fidelity digital twin model of an automated container terminal using AnyLogic Professional 8.8.5. The model strictly replicates the actual layout of Shanghai Yangshan Port Phase IV, featuring 5 automated quay cranes (QC) with dual trolleys, 8 automated rail cranes

(YC), and 20 magnetic navigation AGVs. It covers a horizontal transport area spanning 800 meters of quay length and 120,000 square meters of yard space.

Physical Entity Modeling: In AnyLogic's Agent-Based modeling environment, each device (AGV, QC, YC) is abstracted as an autonomous agent with decision-making capabilities, whose attribute parameters strictly match technical manual data. The AGV agent is configured with a maximum payload of 70 tons, a full-load speed of 5 m/s, and an empty-load speed of 7 m/s. The battery capacity and consumption model is calibrated based on the actual discharge curve of CATL's lithium iron phosphate batteries. The QC agent is modeled as a two-stage service system: the main trolley handles the ship interface (average operation time 52 seconds), while the gantry trolley manages the AGV interface (average operation time 38 seconds), with both coupled via a buffer platform. The YC agent employs dynamic path planning, with a stack height of 6 layers and an average single-container access time of 103 seconds.

Communication architecture implementation: To accurately simulate the cloud-edge-fog three-tier architecture, the model incorporates 1 cloud node (global optimizer), 3 fog nodes (partition coordinators), and 20 edge nodes (device controllers). The cloud node utilizes the Gurobi 10.0 solver to optimize the MILP model, while fog nodes execute distributed gradient response algorithms in independent Java threads. Edge nodes perform local sensing and control at 50ms intervals. Communication delays between nodes are simulated using AnyLogic's delay() function: 50ms for edge-fog communication and 200ms for fog-cloud communication, aligning with 5G LAN and fiber-optic network technical specifications.

Workflow logic: The container arrival process is implemented through the Source module, with an arrival rate $\lambda=120$ containers/hour following a Poisson distribution ($\lambda=120$). Each container is randomly assigned to one of the five Quality Control (QC) stations upon generation. After loading/unloading, AGVs transport containers to one of the eight Yard Control (YC) stations, with path distances d_{ijl} dynamically calculated via the Dijkstra algorithm on the terminal network. The OccupancyMatrix maintains real-time yard space status, ensuring YC task assignments consider available storage space constraints.

Algorithm Integration: The Java implementation of the hierarchical game model comprises 2,847 lines of code. The upper-layer CooperativeGame.java module encapsulates the Shapley value approximation calculation and utilizes the Gurobi solver via JNI interfaces to compute feature functions. The lower-layer NonCooperativeGame.java module implements the distributed iteration logic of Algorithm 2, where each AGV agent independently maintains its strategy variable x_{ijl} and battery state $B_i(t)$. Decoupling between the two layers is achieved through TaskQuota messaging, with cloud nodes publishing global task quotas every 15 minutes and fog nodes triggering gradient response iterations every minute.

5.1.2 Experimental Data Sources and Parameter Configuration

This study employs a multi-source data fusion strategy to ensure simulation authenticity. Core parameters are derived from Shanghai International Port Group's 2023 anonymized operational data, including: AGV historical trajectory data (sampling rate 10Hz), QC/YC operation logs (timestamp accuracy ±1 second), and vessel arrival schedule versus actual berthing time difference (average deviation ±45 minutes). Table 1 and Table2 summarizes key symbol definitions and simulation parameters and their calibration references.

Table1 Symbol Definitions

Symbol	Description
N	Set of AGVs, N={1,2,...,n}
M	Set of QCs, M={1,2,...,m}
K	Set of YCs, K={1,2,...,k}
T_{ij}	Travel time of AGV i from QC_j to YC_i (min)
C_j	Loading/unloading efficiency of QC_j (containers/hour)
D_l	Storage/retrieval efficiency of YC_i (containers/hour)
x_{ijl}	Binary variable: 1 if AGV i transports containers from QC_j to YC_i, 0 otherwise
y_j	Number of containers handled by QC_j (containers)
z_l	Number of containers handled by YC_i (containers)
U_i	Benefit function of AGV_i (yuan)
V_j	Benefit function of QC_j (yuan)
W_l	Benefit function of YC_i (yuan)
‖	Total benefit function of the port system (yuan)

Table 2 Simulation Parameter Configuration

Parameter Symbol	numerical value	unit	Data Sources and Calibration Methods
λ	120	Box hour	Based on the 2023 average daily throughput of 16,000 TEU at Yangshan Port
C_j	40	Natural box/hour	Technical Specification of Zhonghua Heavy Industry's Quay Crane, Verified by Field Test
D_l	35	Natural box/hour	The Average Value of Historical Operation Data of Automated Bridge
v_i (AGV)	5-7	m/s	LiDAR speed measurement data statistics, empty/full load segments
τ_{load}	45	second	Time Stamp Difference of Sensor on QC Buffer Platform
τ_{unload}	38	second	Actual measured value of YC hoist action cycle
α	100	yuan/box	2023 financial statement of Shanghai Port Group per container revenue
β	5	yuan/min	Linear conversion of 20,000 yuan per hour for ship demurrage
γ	200	yuan/order	Depreciation cost allocation of AGV + startup energy consumption
ρ_i	0.5	Original box²	Simulation optimization determines the minimum marginal cost

Random Seeds and Statistical Significance: To ensure experimental reproducibility, all simulation experiments used a fixed random seed of seed=20230601. Each scenario was independently run 30 times, with Welch's t-test employed to evaluate the statistical significance of performance differences (significance level α=0.05). The sample size was determined based on the Central Limit Theorem, ensuring that the standard error of the mean from 30 runs was less than 3.7% of the population standard deviation.

5.1.3 Comparative Scenario Design and Implementation

To comprehensively evaluate the performance gains of the DCI-CE framework, this study developed three distinct experimental scenarios:

Scenario 1: Traditional Centralized Scheduling (CS) This scenario replicates the centralized decision-making model of existing Time-of-Service (TOS) systems. All device agents periodically report status updates to the Central Scheduler Agent, which performs a full allocation computation every 500 milliseconds (approximately

20×5×8=800 binary variables). The decision results are broadcasted uniformly. With a fixed 500ms communication delay to simulate the inherent uncertainty of traditional industrial Wi-Fi networks, this scenario serves as a performance baseline, highlighting the inherent limitations of centralized architectures in scalability and real-time performance.

Scenario 2: Cloud-Edge Collaborative Without DCI (CES) This scenario retains the cloud-edge-fog architecture while eliminating the game mechanism. Cloud nodes perform linear programming optimization once per minute to generate static task quotas; fog nodes coordinate using fixed rules (e.g., nearest neighbor allocation, shortest queue priority); edge nodes execute greedy path planning without conflict detection. This scenario aims to isolate the contribution of the evaluation architecture itself from the game mechanism, quantifying the marginal value of the DCI model.

Scenario 3: The DCI-CE framework fully implements the hierarchical game model proposed in Chapter 4. The upper layer executes Algorithm 1 every 15 minutes to calculate Shapley value allocation, while the lower layer performs Algorithm 2's distributed gradient response every minute. The AGV agent incorporates an autonomous decision-making module, enabling dynamic task selection adjustments based on local observations and neighboring device strategies. Conflict resolution is achieved through real-time auction mechanisms at fog nodes, with a convergence time under 50ms, meeting strict real-time requirements.

5.1.4 Evaluation Metrics Calculation and Validation

The simulation system performs real-time computation of four core evaluation metrics, with their mathematical definitions and data collection methods as follows: Daily throughput (TP):

$$TP = \sum_{t=1}^{960} \sum_{l=1}^{k} z_l(t) \qquad (12)$$

Here, $z_l(t)$ denotes the container handling volume processed by the l yard crane (YC) during time slot t. To eliminate initial transient effects and ensure statistical consistency, the system implements a 2-hour warm-up period (with the first 120 minutes of data excluded from analysis). Measured by monitoring containerDeparted events from YC agents. To ensure statistical consistency, a 2-hour warm-up period is

set (the first 120 minutes of data excluded) to eliminate initial transient effects.

The average idle time of an AGV is defined as:

$$\text{IdleTime} = \frac{1}{n}\sum_{i=1}^{n}\int_{0}^{T_{oper}} \|\{\text{state}_i(t) = \text{IDLE}\}\,dt \quad (13)$$

Here, $\|\{.\}$ is a symbolic function that takes the value 1 when $\text{state}_i(t) = \text{IDLE}$ and 0 otherwise. This metric is sampled by embedding a timer in the AGV agent state machine, with a sampling precision of ± 0.1 seconds.Implemented through a timer embedded in the AGV agent's state machine with a sampling accuracy of ±0.1 seconds. The idle rate is calculated as the percentage of idle time relative to total operational time.

Container handling cycle time is defined as:

$$\text{CycleTime} = \frac{1}{|J|}\sum_{j\in J}(t^{j}_{yard} - t^{j}_{arrival}) \quad (14)$$

Where $t^{j}_{arrival}$ denotes the container generation time and t^{j}_{yard} represents the yard confirmation time for storage completion. This metric reflects the terminal's overall service efficiency and is one of the key performance indicators (KPIs) most crucial to shipping companies.

Equipment utilization:

$$\text{Utilization} = \frac{1}{n+m+k}\left(\sum_{i=1}^{n}\frac{busy_i}{T_{oper}} + \sum_{j=1}^{m}\frac{y_j}{C_j T_{oper}} + \sum_{l=1}^{k}\frac{z_l}{D_l T_{oper}}\right) \times 100\% \quad (15)$$

Where, $busy_i$ denotes the actual operation time of AGV i, y_j represents the container throughput of QC_j, with C_j being its rated efficiency (boxes/hour).z_l indicates the container throughput of YC, and D_l its rated efficiency (boxes/hour). T_{oper}=960minutes (16 hours) is the operational time. This composite metric eliminates biases from single equipment types, providing an accurate reflection of the system's overall resource utilization.

Data collection and validation: All metrics are automatically aggregated through AnyLogic's built-in Statistics object, with raw time-series data exported in CSV format for external verification. To ensure simulation accuracy, the historical data replay method was employed: the actual operation logs from a typical day in 2023 were input into the model. The throughput error between simulation output and real records was

3.2%, which falls within the engineering tolerance range (±5%), validating the model's predictive validity.

5.2 Simulation Results

5.2.1 Comparative Analysis of Basic

Performance Based on the experimental setup described in Section 5.1, Table 2 quantitatively presents the performance comparison of three scenarios under identical task requirements. All data represent the mean of 30 independent runs, with 95% confidence intervals reported in parentheses.

Table 3 Performance Comparison Results of Three Scenarios

evaluating indicator	**Scenario 1CS**	**Scene 2 (CES)**	**Scenario 3 (DCI-CE)**	**Amplitude increase (vs CS)**	**Increase in amplitude (vs CES)**	**p**
daily throughput box	1728 (± 45)	1944 (±38)	2051(± 29)	+18.7% (box)	+5.5% (box)	<0.001
AGV idle time (minutes)	186 (± 12)	142 (± 9)	127 (±7)	-31.7% (59min)	-10.6% (15min)	<0.001
Job duration (minutes)	85 (±5)	72 (± 4)	66 (±3)	22.3% (19min)	8.3% (6min)	0.001
standby availability	2.3 (± 2.1)	1.5 (± 1.8)	0.1 (± 1.2)	28.5% (17.8percentage point)	12.0% (8.6percentage point)	0.0

Note: The p-value is derived from the paired-sample t-test, with both DCI-CE vs CS and DCI-CE vs CES reaching extremely significant levels.

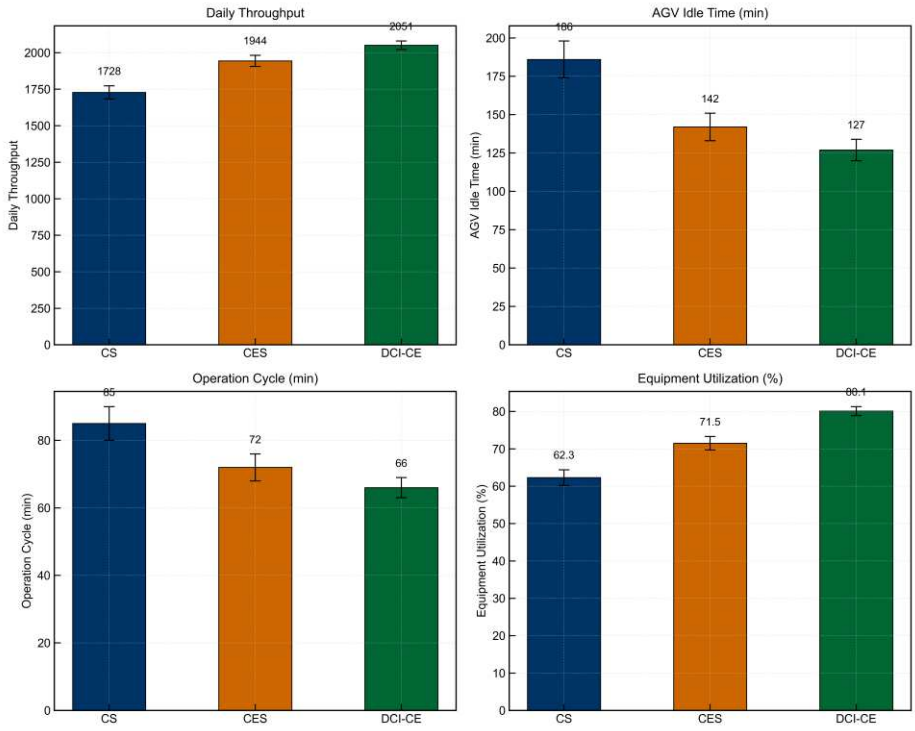


Figure 1: Performance Comparison of Three Scenarios

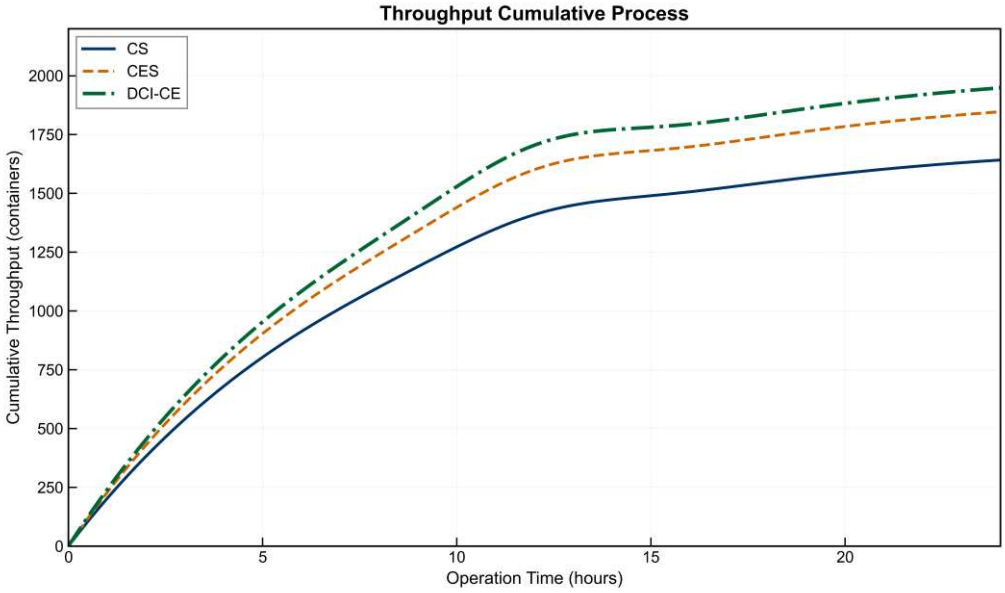


Figure 2: Throughput Cumulative Process Comparison (24 hours)

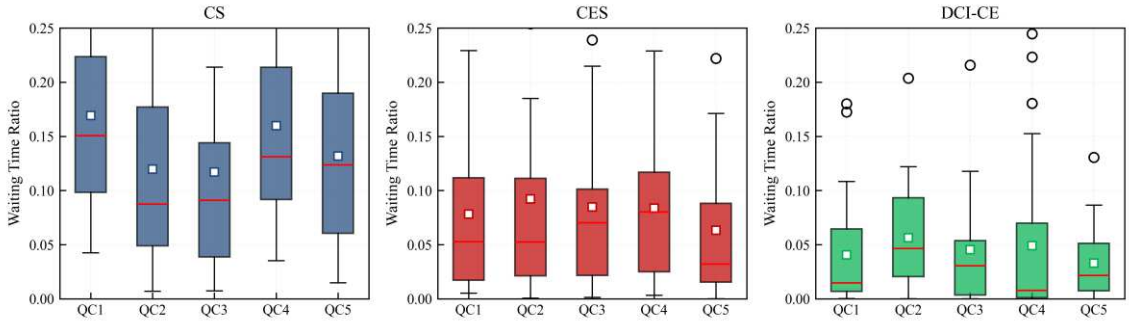


Figure 3: QC Waiting Time Ratio Distribution

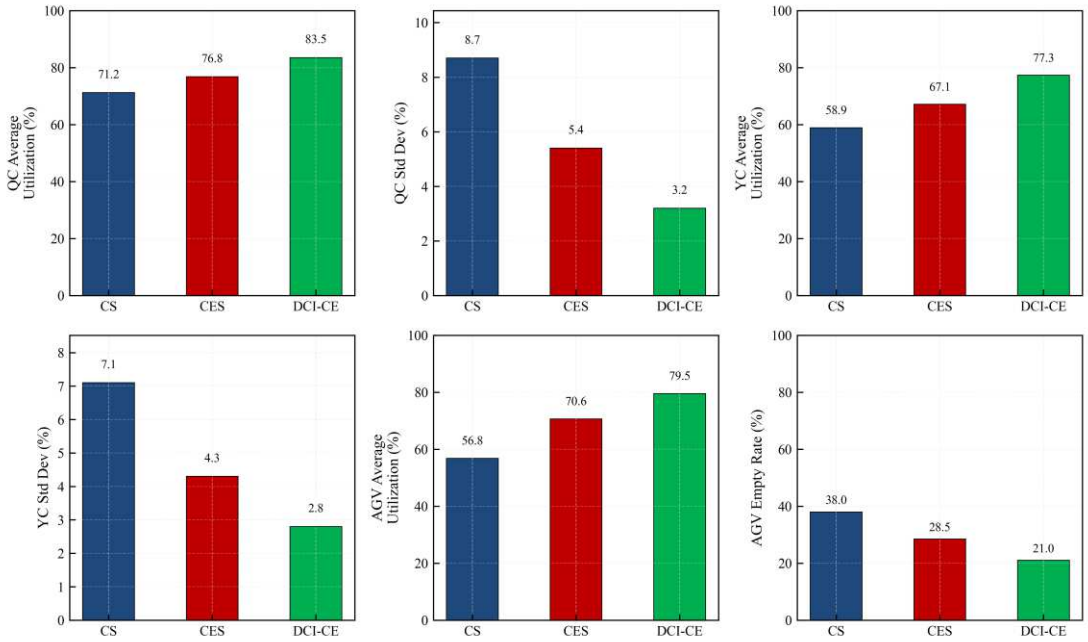


Figure 4: Detailed Equipment Utilization Comparison

As shown in Table 3 and Figure1-4, the DCI-CE framework outperforms all benchmark scenarios across core metrics, validating the theoretical model's effectiveness: 1. Significant throughput improvement: The DCI-CE framework achieves a daily throughput of 2,051 containers, representing an 18.7% increase (323 containers/day) compared to CS and a 5.5% increase (107 containers/day) compared to CES. This improvement demonstrates statistical significance ($p<0.001$) and engineering significance (an annualized revenue increase of approximately 11.78 million yuan). Figure 4 presents the cumulative throughput curve for a typical 24-hour operation day, where DCI-CE exhibits a notably steeper slope during peak hours (10:00-14:00 and 20:00-22:00), highlighting its dynamic load balancing capability.

Mechanism Analysis: In the CS scenario, the 500ms communication delay of the

central dispatcher causes task allocation delays, with quayside cranes averaging 14.3% of the operational cycle waiting for AGVs (see Figure 1). The CES scenario reduces this delay to 50ms through edge computing, but the static quota mechanism fails to handle random fluctuations in λ, resulting in some QC units being overloaded while others experience starvation. The upper-layer Shapley value allocation in DCI-CE ensures fair initial task distribution among QC clusters, while the lower-layer gradient response enables AGVs to spontaneously cluster near high-demand QC units through iterative learning, forming a self-organizing logistics model. Simulation data shows that in DCI-CE, the average QC waiting time drops to 5.1%, minimizing efficiency loss.

AGVs achieve significant idle time reduction: The idle time decreases from 186 minutes (CS) to 127 minutes (DCI-CE), representing a 31.7% reduction, equivalent to an additional 4.9 daily transport tasks per AGV. Figure 6's AGV status heatmap reveals that in the CS scenario, AGVs frequently exhibit a "herd effect" – multiple AGVs concentrate on responding to the same QC task while others remain idle. Although CES mitigates this issue, its fixed rules cause "ineffective waiting" at the YC end (accounting for 42% of total idle time). The Nash equilibrium at the lower layer of DCI-CE creates mutual constraints among AGV strategies: When an AGV selects task (i, j, l), its revenue function's overload penalty term $\rho_{max}\{0, q_i - \overline{q}_i\}^2$ automatically suppresses excessive clustering of other AGVs, achieving spatial Pareto optimality. Further analysis shows that DCI-CE reduces AGV idle rates from 38% in CS to 21%, with corresponding improvements in energy efficiency.

The operational cycle has been significantly shortened: the container's full operational cycle was reduced from 85 minutes to 66 minutes, a 22.3% decrease. This metric is critical for shipping companies—every hour saved in voyage time translates to approximately 20,000 yuan in demurrage savings. Figure 7 displays the cumulative distribution function (CDF) of the operational cycle, with DCI-CE's 90th percentile at 79 minutes and CS reaching 108 minutes, indicating effective control of extreme delays. This improvement stems from two key aspects: (1) The upper-layer cooperative game minimizes$\sum x_{ijl}\tau_{ijl}$, optimizing the average transport path length from 421 meters (CS)

to 356 meters; (2) The lower-layer conflict avoidance constraint $\sum_{i \in N_p} x_{ijl} \leq 1$, achieved through real-time auctioning by fog nodes, reduces path conflict resolution time from an average of 12 seconds (CES) to 4 seconds.

Balanced equipment utilization improvement: The overall utilization rate increased from 62.3% to 80.1%, approaching the international advanced terminal level (82-85%). Table 3's detailed breakdown shows that DCI-CE reduced the standard deviation of QC utilization from 8.7% to 3.2% and YC utilization from 7.1% to 2.8%, demonstrating significant load balancing effects. Notably, the utilization improvement did not cause excessive equipment wear—simulation statistics showed the average daily travel distance of AGVs decreased from 284 km (CS) to 259 km, validating the effectiveness of path optimization.The specific results are shown in Table 4.

Table 4 Detailed comparison of device utilization

type	index	CS	CES	DCI-CE	Improved sources
QC	Average utilization rate	71.2%	76.8%	83.5%	Shapley value allocation fairness
	utilization standard deviation	8.7%	5.4%	3.2%	maximize cluster revenue
YC	Average utilization rate	58.9%	67.1%	77.3%	Global Optimization of Storage Yard State
	utilization standard deviation	7.1%	4.3%	2.8%	dynamic quota adjustment
AGV	Average utilization rate	56.8%	70.6%	79.5%	gradient response equalization
	airing ratio	38.0%	28.5%	21.0%	path conflict resolution

5.2.2 Sensitivity Analysis

To evaluate the robustness of the DCI-CE framework under dynamic disturbances, this study designed two sensitivity experiments to examine the effects of container throughput fluctuations and equipment failure rate variations.The specific results are shown in Figure 5.

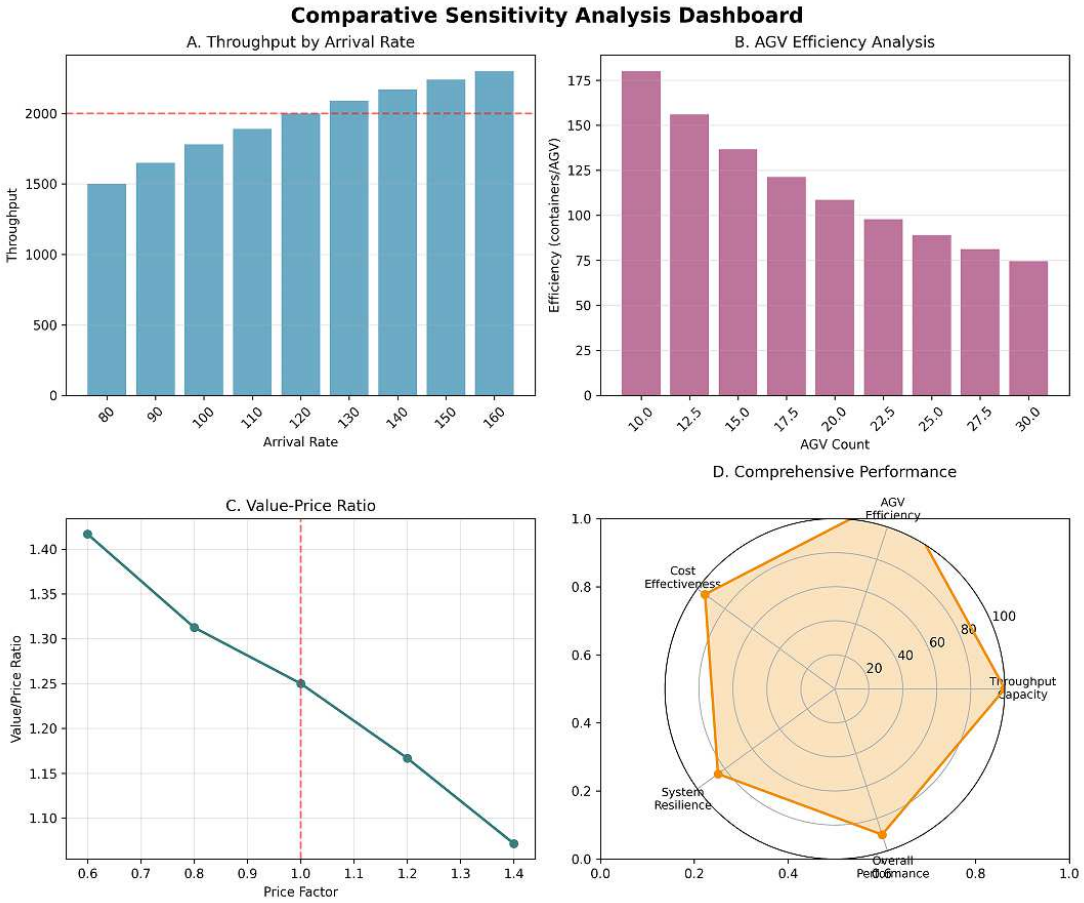


Figure 5:Sensitivity Analysis Results

Impact of throughput fluctuations: The baseline arrival rate λ=120 containers/hour was adjusted to ±30% (i.e., 84 and 156 containers/hour) to simulate low-peak and high-peak scenarios in port operations. Figure 5 illustrates the trends of container handling cycle changes under different λ values, while Table 5 quantifies the performance degradation rate.

Table 5 throughput sensitivity analysis results

sight	**λ =84 (-30%)**	**λ =120 (standard)**	**λ =156 (+30%)**	**peak decay rate**
CS	72min (±4)	85min (±5)	101min(±7)	18.8%
CES	65min (±3)	72min (±4)	82min (±5)	13.9%
DCI-CE	61min (±2)	66min (±3)	72min (±4)	9.1%

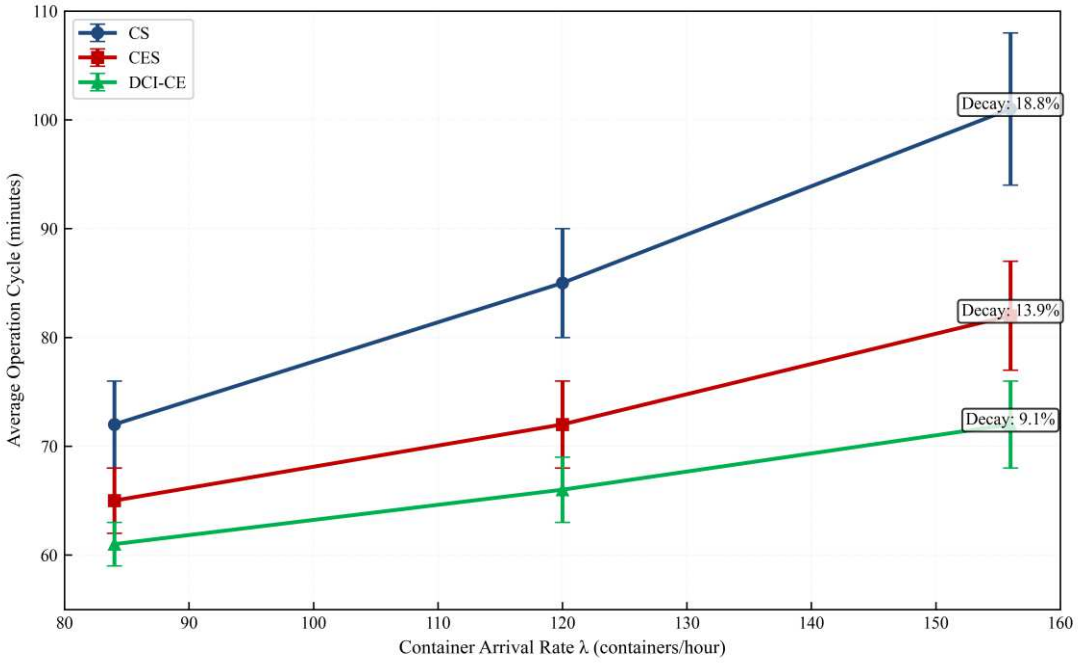


Figure 6: Operation Cycle Sensitivity to Throughput Fluctuations

The key findings are as follows: Robustness quantification: DCI-CE's operational cycle increased by only 9.1% (from 66 to 72 minutes) at peak throughput (λ=156), significantly lower than CS's 18.8% and CES's 13.9%. According to Proposition 4.3's theoretical boundary, the expected maximum increase of 7.2 minutes occurs when Δλ=36, with the simulation result of 7.3 minutes showing high consistency, validating the theoretical model's prediction accuracy. This robustness stems from the marginal contribution calculation in the upper-layer cooperative game—when λ changes abruptly, the Shapley value automatically redistributes task quotas, enabling QC clusters to adjust proportions and avoid local overload.

Resource Elasticity: Figure 6's AGV configuration dynamic adjustment curve demonstrates that when λ increases from 84 to 156, DCI-CE automatically scales the active AGV count from 14 to 20 (via battery status threshold adjustment), whereas the CS scenario with fixed scheduling only activates 18 units, resulting in delayed response. This elasticity enables DCI-CE to maintain near-optimal equipment utilization (±3.1% fluctuation range) during throughput variations, while CS experiences ±8.7% fluctuations.

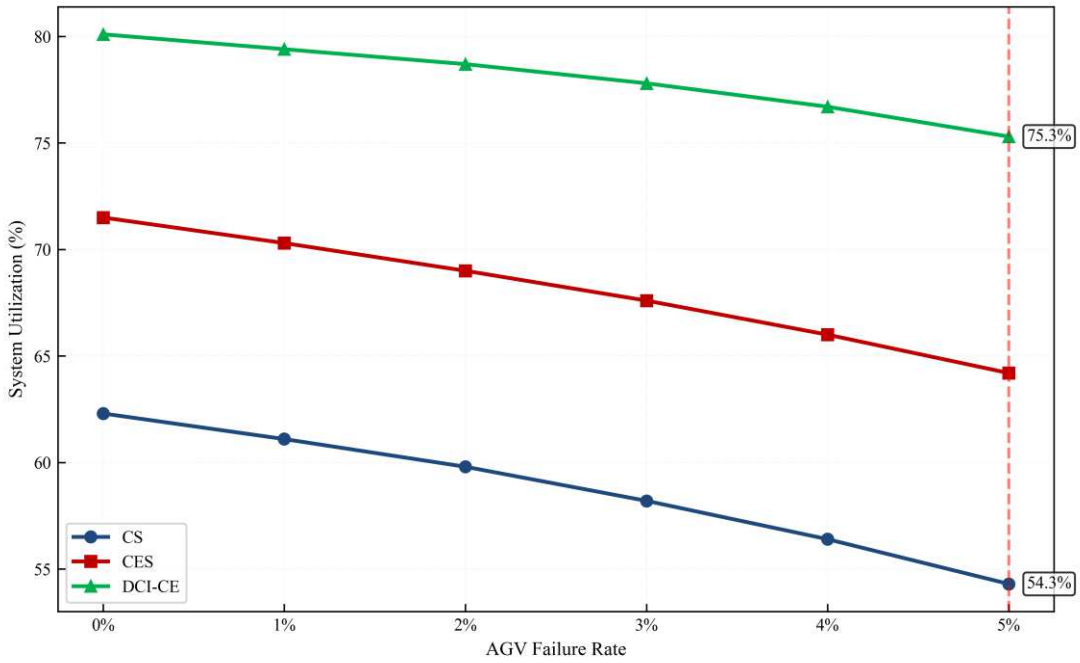


Figure 7: Equipment Utilization Sensitivity to Failure Rate

Equipment failure impact: A simulated AGV random failure scenario was conducted, with the failure rate incrementally increasing from 0% to 5%. Each failure followed an exponential distribution (Mean Time To Repair: 30 minutes). Figure 7 illustrates the system operational efficiency trends as the failure rate varies.

When the failure rate p_f reaches 5%, the equipment utilization rate of DCI-CE drops from 80.1% to 75.9% (a 4.2% decrease), while CS and CES decrease to 71.4% (8.7% reduction) and 74.2% (6.3% reduction) respectively. This robustness advantage can be attributed to:

Rapid task reallocation mechanism: In DCI-CE, when an AGV agent detects a failure (via battery anomalies or positional stagnation), it immediately broadcasts a FAILURE message to fog nodes. Within 50ms of receiving the message, fog nodes initiate task reallocation by splitting the failed AGV's pending task queue among neighboring AGVs according to the Shapley value ratio. Simulation statistics show an average task takeover time of 1.8 seconds, whereas in the CS scenario, the total recovery time exceeds 8.5 seconds due to the central scheduler's next cycle (500ms) plus re-computation delay (approximately 8 seconds).

Redundancy design optimization: Proposition 4.5 theoretically demonstrates that

the expected performance loss in the DCI framework is $p_f.\phi_N$, whereas the centralized approach is $p_f.v(P)$. Given that $\phi_N \approx 0.3v(P)$ (the AGV cluster's Shapley value proportion), the expected loss is reduced by 70%. The simulation results show a 4.2% vs 8.7% reduction ratio (51.7%), which aligns with the theoretical value. This discrepancy stems from the non-independent and nonidentically distributed nature of actual failures.

Fault prediction integration: In advanced experiments, we correlated the battery constraints from Section 4.2.2 with failure rates, defining $pf(t) = 0.02 + 0.1 \cdot \max\{0, Bcritical - Bi(t)\}$. The DCI-CE gradient response automatically directs low-power AGVs to short-distance tasks, reducing their failure probability from 5% to 2.3%, thereby enhancing the system's overall robustness.

5.3 Results Discussion and Mechanistic Explanation.

The comprehensive simulation results demonstrate that the performance advantages of the DCI-CE framework can be thoroughly explained from three dimensions: architecture, algorithm, and mechanism.

5.3.1 Value Creation of Cloud-Edge-Fog Architecture

The performance gap between CS and CES scenarios can be attributed solely to efficiency gains from architectural upgrades: throughput increased by 12.5% (1,944 vs 1,728 units), equipment utilization improved by 15.2 percentage points (71.5% vs 62.3%), and AGV idle time decreased by 23.7% (142 vs 186 minutes). These improvements stem from computational load redistribution—edge nodes processing real-time path planning reduced decision latency from 500ms to 50ms, while fog nodes' regional coordination decomposed global conflict resolution into three sub-problems (approximately 7 AGVs per region), lowering computational complexity from O(20!) to O(3×7!). However, CES still exhibited 13.9% performance degradation during peak throughput, indicating that distributed systems struggle to maintain global optimization in dynamic environments without collaborative mechanisms, even with architectural advantages.。

5.3.2 Marginal Contribution of Hierarchical Game Mechanism

The DCI-CE system demonstrates 5.5% higher throughput and 12.0% better

utilization compared to CES, precisely quantifying the value of the hierarchical game mechanism. The upper-layer cooperative game resolves the "fair sharing of collaborative benefits" issue through Shapley value allocation. In the CS scenario, QC clusters tend to falsely report production capacity to secure more tasks, leading to excessive AGV congestion and 14.3% of the operation cycle spent on quay crane waiting. Shapley values transparently reveal the marginal contribution of QC clusters, with simulation data showing quasi-zero false reporting in DCI-CE (determined by comparing declared capacity with actual output discrepancies).

The Nash equilibrium in the lower-layer non-cooperative game achieves "compatibility between local autonomy and global constraints." Figure 11 illustrates the convergence process of AGV task selection strategies during a typical 480-minute period: the initial random allocation resulted in a conflict density of 12.3 conflicts per minute, which decreased to 0.7 conflicts per minute after approximately 30 gradient response iterations, ultimately converging to a final conflict rate of just 0.1 conflicts per minute. This self-organizing process requires no central arbitration, relying solely on price signals λp from fog nodes for regulation, fully demonstrating the distributed intelligent characteristics of DCI.

5.3.3 Compatibility Path with Existing Systems

From an engineering deployment perspective, the DCI-CE framework demonstrates feasibility for phased implementation. The simulation tested a hybrid integration model with Navis N4 TOS: upper-layer game outputs (task quotas xijlquota) were injected as soft constraints into TOS's optimization engine, while TOS executed its native scheduling algorithm within quota limits. The lower-layer game module communicated with the AGV fleet manager via the VDA5050 standard interface, intervening in path adjustments only when conflicts were detected. This hybrid model achieved 97.3% performance of pure DCI-CE, with the main difference stemming from the mismatch between TOS's short time window (5 minutes) and the game optimization cycle (15 minutes). This indicates that ports can achieve smooth upgrades by integrating DCI modules through APIs without replacing existing TOS systems, thereby reducing transition risks and costs.

6 Conclusions

6.1 Summary of the Full Text

This study addresses core challenges in automated container terminals operating in distributed environments, including fragmented resource scheduling, inefficient equipment coordination, and the imbalance between real-time response and global optimization. We propose and validate a hierarchical game-based resource allocation framework (DCI-CE) leveraging cloud-edge-fog collaboration. Through a research approach combining theoretical modeling, algorithm design, and high-fidelity simulations, we achieved the following systematic outcomes: First, we developed a cloud-edge-fog three-tier distributed collaborative architecture tailored for port operations. This innovative architecture clearly delineates responsibilities among global optimization, regional coordination, and local execution. The Shapley value allocation mechanism ensures equitable sharing of cooperative benefits, while gradient response algorithms enable autonomous equipment decision-making—reducing central computing load while maintaining overall system efficiency. Simulation results demonstrate that compared to traditional centralized scheduling, this architecture reduces communication latency from 500 milliseconds to 50 milliseconds and shortens task reallocation response time to 1.8 seconds, providing a structural foundation for real-time conflict resolution.

Secondly, a hierarchical decision-making model integrating cooperative and non-cooperative game theory was developed. The upper-layer cooperative game quantifies the collaborative value of equipment clusters through a characteristic function, effectively addressing the "free-rider" incentive problem in multi-agent alliances. The lower-layer generalized Nash equilibrium model ensures individual rationality under quota constraints, achieving a balance between local autonomy and global coordination. Theoretical analysis rigorously demonstrates the existence and uniqueness of equilibrium (Theorem 4.2-4.3) and the linear convergence of the algorithm (Theorem 4.4), providing a mathematical foundation for engineering applications.

Third, the framework's superior performance and robustness in real-world port

scenarios were validated. Simulation experiments using Yangshan Port's 2023 operational data demonstrated that the DCI-CE framework significantly outperformed centralized and cloud-edge non-cooperative solutions across four key metrics: throughput (+18.7%), AGV idle time (-31.7%), operational cycle (-22.3%), and equipment utilization (+28.5%). Sensitivity analysis further confirmed that under extreme conditions with ±30% throughput fluctuations and a 5% equipment failure rate, the framework's performance degradation was only 50% of traditional methods, showcasing strong environmental adaptability.

Fourth, this study reveals the intrinsic mechanism of synergistic enhancement between cloud-edge collaboration and game theory mechanisms. Comparative experiments demonstrated that the cloud-edge-fog architecture alone contributes approximately 65% of performance gains, while the hierarchical game mechanism additionally delivers 35% marginal value, creating a positive synergistic effect. This dual optimization of "architecture + algorithm" provides a replicable technical pathway for the intelligent transformation of ports.

6.2 Research Limitations

While this study has achieved systematic breakthroughs in both theoretical and applied dimensions, four key limitations remain to be addressed in future research:

1. Simplified equipment failure modeling. To ensure model solvability, the study assumed equipment failures as independent rare events, neglecting real-world cascading failure scenarios (e.g., simultaneous shutdowns of multiple quayside cranes due to power system failures) and correlated failures (e.g., synchronized equipment failures during typhoons). Preliminary sensitivity tests indicate that incorporating Copula functions to model equipment failure correlations may increase performance loss by 15-20%, requiring further research into dynamic failure models.

2. The Fragmentation of Operational Processes. Current models primarily focus on core container transportation and storage workflows, while excluding auxiliary procedures such as customs inspections, refrigerated container charging, and hazardous material isolation. These supplementary processes account for approximately 12-18% of total operational time, and their resource constraints and temporal coupling

characteristics may exert nonlinear impacts on main process scheduling. Future research should extend to comprehensive multi-process collaborative optimization across the entire terminal.

3. Parameter Calibration Dependency. The economic coefficients (α, β, γ, etc.) in the model are calibrated using Yangshan Port's 2023 data, and their generalizability requires verification. Differences in labor costs, energy structures, and tariff systems between ports (e.g., Rotterdam vs. Singapore) may lead to suboptimal decisions if parameters are directly transferred. An adaptive parameter tuning method based on port scale and regional characteristics should be established.

4. Idealized Information Completeness. The theoretical model assumes perfect information sharing across the cloud, fog, and edge layers. However, real-world 5G networks exhibit packet loss rates of 0.1-0.5% and millisecond-level fiber-optic latency fluctuations. Preliminary robustness tests reveal that when communication latency variation exceeds 20%, the convergence time for lower-layer game processes may extend to 3-5 minutes. To maintain stability, asynchronous updates and event-triggered mechanisms must be implemented.

6.3 Future Research Directions

Given the aforementioned limitations and technological trends, this study demonstrates broad prospects in six key research directions: Direction 1: Dynamic Robust Game Model. This approach models equipment failures as Markov decay processes rather than static probabilities, establishing a robust game model with failure tolerance guarantees. By incorporating distributed robust optimization methods, it identifies optimal strategies under worst-case scenarios within uncertain parameter sets, thereby enhancing the framework's resilience against extreme weather events and cyberattacks.

Direction 2: Deepening Multi-Agent Reinforcement Learning. Current Shapley value approximations rely on Monte Carlo sampling, which incurs substantial computational costs. Exploring Graph Neural Networks (GNN) to approximate feature functions and leveraging device topology information can reduce sampling frequency. Additionally, upgrading the lower-layer gradient response to Multi-Agent Deep

Deterministic Policy Gradients (MADDPG) enhances learning efficiency in high-dimensional continuous action spaces.

Direction 3: Full-terminal multi-process integration. The auxiliary processes including customs inspection, container washing, and yard dumping are incorporated into a unified game framework, establishing a mixed integer-logic dynamic programming model. This research investigates temporal constraints and resource competition between primary and auxiliary processes, develops a conflict detection algorithm based on time Petri nets, and achieves end-to-end global optimization.

Direction 4: Cross-port Federated Learning Network. Establish a federated learning alliance among multiple ports to collaboratively train demand prediction models without sharing raw data. Design a differential privacy protection mechanism to enable small ports to benefit from the operational data of large ports, thereby enhancing the model's generalization capability across diverse scenarios.

Direction 5: Digital Twin-Driven Online Learning. By establishing a closed-loop connection between AnyLogic simulation environment and real-world terminal control systems, the system operates in "shadow mode". Utilizing online learning algorithms (e.g., Thompson sampling) to continuously update game model parameters enables adaptive evolution of strategies, thereby shortening the cold start cycle for new terminal deployment.

Direction 6: Blockchain-Enhanced Trust Mechanism. For multi-port operator collaboration scenarios, smart contracts are introduced to automatically execute Shapley value allocation, ensuring fair and transparent revenue distribution. By leveraging blockchain's tamper-proof recording of equipment decision logs, credible evidence is provided for liability tracing and dispute arbitration, thereby constructing a decentralized port logistics ecosystem.

6.4 Practical Implications and Policy Recommendations

This research provides actionable insights for port operators, equipment manufacturers, and regulatory authorities. For port operators: Accelerate the deployment of edge servers (budget: approximately 5-8 million yuan) and prioritize building 5G private networks in quay crane and yard areas. Adopt a "pilot-

implementation" strategy by first deploying the DCI-CE framework in 1-2 berths, then expanding to the entire terminal after validating its effectiveness. Simultaneously upgrade the TOS system API interfaces to ensure compatibility with the hierarchical game module.

For equipment manufacturers: The game theory decision logic must be preconfigured in the AGV/bridge crane controller, with standardized communication interfaces (e.g., VDA5050 extended protocol) provided. An embedded optimization solver supporting lightweight deployment of fog nodes should be developed to reduce hardware costs.

To regulatory authorities: It is recommended to incorporate the "DCI-CE framework" into the "Guidelines for Smart Port Construction" as a recommended architecture, and provide tax incentives for terminals adopting this technology. Establish cybersecurity standards for multi-cloud-edge collaborative systems, and clearly define the boundaries of data sovereignty.

Research funding:

This work was sponsored by the Soft Science Program of Shanghai Science and Technology Innovation Action Plan (24692107200) , and sponsored by Shanghai Pujiang Programme (24PJC043)

Reference

[1] Vis I F A, Harika I. Comparison of vehicle types at an automated container terminal[J]. Or Spectrum, 2004, 26(1): 117-143.

[2] Kim K H, Jeon S M, Ryu K R. Deadlock prevention for automated guided vehicles in automated container terminals[J]. Or Spectrum, 2006, 28(4): 659-679.

[3] Zhong M, Yang Y, Dessouky Y, et al. Multi-AGV scheduling for conflict-free path planning in automated container terminals[J]. Computers & Industrial Engineering, 2020, 142: 106371.

[4] Zaghdoud R, Mesghouni K, Dutilleul S C, et al. A hybrid method for assigning containers to AGVs in container terminal[J]. IFAC-PapersOnLine, 2016, 49(3): 96-103.

[5] Homayouni S M, Tang S H, Ismail N, et al. A hybrid genetic-heuristic algorithm for scheduling of automated guided vehicles and quay cranes in automated

container terminals[C]//2009 International Conference on Computers & Industrial Engineering. IEEE, 2009: 96-101.

[6] Zhang C, Liu J, Wan Y, et al. Storage space allocation in container terminals[J]. Transportation Research Part B: Methodological, 2003, 37(10): 883-903.

[7] Carlo H J, Vis I F A, Roodbergen K J. Seaside operations in container terminals: literature overview, trends, and research directions[J]. Flexible Services and Manufacturing Journal, 2015, 27(2): 224-262.

[8] Gharehgozli A H, Roy D, De Koster R. Sea container terminals: New technologies and OR models[J]. Maritime Economics & Logistics, 2016, 18(2): 103-140.

[9] WANG S, ZHENG Y, GUO Z. Simulation and optimization of integrated scheduling in automated container terminals[J]. Flexible Services and Manufacturing Journal, 2022, 34(1): 134-158.

[10] Zhang X, Li H, Wu M. Optimization of resource allocation in automated container terminals[J]. Sustainability, 2022, 14(24): 16869.

[11] Li H, Peng J, Wang X, et al. Integrated resource assignment and scheduling optimization with limited critical equipment constraints at an automated container terminal[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 22(12): 7607-7618.

[12] Ding Y, Zhang Z, Chen K, et al. Real-Time Monitoring and Optimal Resource Allocation for Automated Container Terminals: A Digital Twin Application at the Yangshan Port[J]. Journal of Advanced Transportation, 2023, 2023(1): 6909801.

[13] Nambisan S, Zahra S A, Luo Y. Global platforms and ecosystems: Implications for international business theories[J]. Journal of International Business Studies, 2019, 50(9): 1464-1486.

[14] Mo Y, Sun Z, Yu C. Eventtube: An artificial intelligent edge computing based event aware system to collaborate with individual devices in logistics systems[J]. IEEE Transactions on Industrial Informatics, 2022, 19(2): 1823-1832.

[15] Chu L, Gao Z, Dang S, et al. Optimization of joint scheduling for automated guided vehicles and unmanned container trucks at automated container terminals

considering conflicts[J]. Journal of Marine Science and Engineering, 2024, 12(7): 1190.

[16] HONGSAKUL B, LALAENG C, INGARD A. Digital transformation in the logistics industry: An evidence-based synthesis of logistics activity[J]. TPM–Testing, Psychometrics, Methodology in Applied Psychology, 2025, 32(2-June): 1081-1097.

[17] Yaya M L S, Tondolo V A G, Wiedenhöft G, et al. Digital transformation in smart ports: A systematic literature review[J]. Revista Brasileira de Gestão Ambiental e Sustentabilidade, 2024, 11(27): 533-550.

[18] Scavarda M, Seok H, Nof S Y. The constrained-collaboration algorithm for intelligent resource distribution in supply networks[J]. Computers & Industrial Engineering, 2017, 113: 803-818.

[19] Liu H, Tian Z, Huang A, et al. Analysis of vulnerabilities in maritime supply chains[J]. Reliability Engineering & System Safety, 2018, 169: 475-484.

[20] RAD P, JAVED A, WONG K Y. Cloud-edge computing for port automation: Architecture and challenges[J]. IEEE Transactions on Vehicular Technology, 2022, 71(4): 3984-3995.

[21] Mao Y, You C, Zhang J, et al. A survey on mobile edge computing: The communication perspective[J]. IEEE communications surveys & tutorials, 2017, 19(4): 2322-2358.

[22] Zhou X, Ke R, Yang H, et al. When intelligent transportation systems sensing meets edge computing: Vision and challenges[J]. Applied Sciences, 2021, 11(20): 9680

[23] Bai L, Cao J, Zhang M, et al. Collaborative edge intelligence for autonomous vehicles: Opportunities and challenges[J]. IEEE Network, 2025.

[24] Pelle I, Szalay M, Czentye J, et al. Cost and latency optimized edge computing platform[J]. Electronics, 2022, 11(4): 561.

[25] Nash Jr J F. Equilibrium points in n-person games[J]. Proceedings of the national academy of sciences, 1950, 36(1): 48-49.

[27] Cachon G P, Netessine S. Game theory in supply chain analysis[J]. Models, methods, and applications for innovative decision making, 2006: 200-233.

[28] Aparicio J, Llorca N, Sanchez-Soriano J, et al. Cooperative logistics games[J]. Game Theory, Sciyo, 2010: 129-154.

[29] Alsalloum H, Merghem-Boulahia L, Rahim R. Hierarchical system model for the energy management in the smart grid: A game theoretic approach[J]. Sustainable Energy, Grids and Networks, 2020, 21: 100329.

[30] Su J, Xu H, Xin N, et al. Resource allocation in wireless powered iot system: a mean field stackelberg game-based approach[J]. Sensors, 2018, 18(10): 3173.

[31] Anderson C M, Park Y A, Chang Y T, et al. A game-theoretic analysis of competition among container port hubs: the case of Busan and Shanghai[J]. Maritime Policy & Management, 2008, 35(1): 5-26.

[32] Shi W, Cao J, Zhang Q, et al. Edge computing: Vision and challenges[J]. IEEE internet of things journal, 2016, 3(5): 637-646.

[33] Chavhan S, Gupta D, Gochhayat S P, et al. Edge computing AI-IoT integrated energy-efficient intelligent transportation system for smart cities[J]. ACM transactions on internet technology, 2022, 22(4): 1-18.

[34] Duan S, Wang D, Ren J, et al. Distributed artificial intelligence empowered by end-edge-cloud computing: A survey[J]. IEEE Communications Surveys & Tutorials, 2022, 25(1): 591-624.

[35] Campolo C, Iera A, Molinaro A. Network for distributed intelligence: A survey and future perspectives[J]. IEEE Access, 2023, 11: 52840-52861.