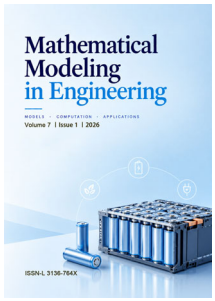
UNIVERSITY OF WEST ATTICA

eISSN: 3136-764X

Mathematical Modeling in Engineering

Vol. 7, No. 3, July, 2026



Robust Feature Fusion Perceiver IO-GNN for Digital Twin Quay Crane Scheduling in Smart Ports

Rintaro Soma[a], Genta Iwamoto[a,*]

[a] *Faculty of Information Engineering, Hokkaido University, Sapporo, 060-0808, Japan*

Received 28 April 2026; Revised 17 July 2026; Accepted 21 July 2026; Published 25 July 2026

Abstract

Quay-crane scheduling in smart ports must accommodate irregular vessel arrivals, unreliable sensor streams, yard-side congestion, and safety separation constraints that are difficult to represent with static dispatch rules. This study develops a robust feature fusion Perceiver IO-GNN model for digital-twin-based quay-crane scheduling. Vessel, crane programmable logic controller, berth, yard-truck, and weather observations are first aligned in a reliability-weighted feature space that retains missingness, observation age, and cross-source consistency. Perceiver IO then maps the asynchronous multimodal tokens into a fixed latent state, while a graph neural scheduling layer represents crane interference, task precedence, berth adjacency, and yard-transfer coupling. A multi-objective decoder generates feasible rolling assignments that are evaluated in the digital twin before release. Experiments use a simulated coastal terminal containing 42 vessel calls, 9 quay cranes, 1,260 container task batches, and five nominal or disturbed operating scenarios. Relative to the rule-based baseline, the proposed model reduces mean vessel turnaround time by 18.7% and crane idle time by 21.4%. Under 20% sensor dropout, robust task completion improves by 15.9%, while normalized crane energy per handled container decreases by 9.8%. These results indicate that reliability-conditioned fusion, latent multimodal encoding, and graph-constrained candidate evaluation can jointly improve scheduling continuity under noisy and coupled terminal conditions. The findings support further field-oriented validation as a dispatcher decision-support method rather than immediate autonomous deployment.

Keywords: Quay Crane Scheduling, Perceiver IO GNN, Robust Feature Fusion, Port Digital Twin, Crane Interference Optimization

1. Introduction

Container terminals are now closely linked cyber-physical production systems, and the former independent operating divisions for berth allocation, quay crane sequencing, yard transfer, gate appointment and safety supervision have been merged [1]. Modern smart ports can automatically collect data from the identification system, crane programmable logic controller signals, spreader load records, video-derived vessel bay status, and truck appointment streams; however, the actual benefits of this data will only be realized if a scheduling model can transform irregular observations into timely decisions [2]. Digital Twin technology can build a virtual model of this environment that changes dynamically in sync with the actual state of the equipment [3]. However, the twin has the problems of delayed sensor packets, measurement drift, uneven task density across bays, and sudden changes in vessel priority in quay crane operation [4]. The previous scheduling assumption of a fixed-length reliable task and predictable interference may not be valid in the presence of yard congestion and weather-induced productivity fluctuations, and many deep learning models still use uniform time grids or single-source event logs [5]. Therefore, the representation strategy of a smart port scheduler should be able to handle multi-modal input without flattening the engineering structure of crane interference and berth adjacency [6].

Corresponding author: genta.iw@eng.hokudai.ac.jp

https://doi.org/10.68406/mme.2026.vol7iss3nm4:42-55

➢ This is an open access article under the CC BY-NC-SA 4.0 license (https://creativecommons.org/licenses/by/4.0/).

© Authors retain all copyrights.

The Quay Crane Scheduling Problem also has the feature of both local and terminal-wide effects in its decisions. Assigning a single crane to a dense bay will reduce the service time of the priority vessel but may also interfere with other cranes, increase travel energy consumption, and starve yard trucks serving other vessels [7]. Rolling-horizon control modifies the initial plan based on changes in circumstances, and frequent adjustments disrupt the workers' work process and reduce their trust in the dispatch platform [8]. Graph neural networks are suitable for modelling cranes, bays and tasks as connected entities, but their performance is limited by the high-quality node features and edge attributes, which are precisely the elements degraded by noisy sensing in active terminals [9]. Transformer-based models improve the modelling of long-range dependencies; however, their quadratic attention cost and fixed tokenization can be restrictive in a port twin that integrates high-frequency machine signals with sparse vessel event updates [10]. Therefore, robust feature fusion serves as a prerequisite for the next stage of learning, and otherwise, it would be impossible to recognize real traffic congestion due to amplified data errors [11].

This paper introduces a robust feature fusion Perceiver IO-GNN model for smart-port quay-crane twin scheduling. The contribution is not merely the use of three established components, but their organization into a single decision loop. First, reliability-conditioned entity tokens preserve missingness, observation age, physical consistency, and cross-source disagreement before multimodal compression. Second, Perceiver IO absorbs a variable number of asynchronous vessel, crane, bay, and task observations without imposing a uniform raw-data grid. Third, graph decoding retains explicit crane interference, task precedence, berth adjacency, and yard-transfer relationships when ranking assignments. Finally, the digital twin evaluates several feasible candidates under delay, energy, safety, and schedule-stability objectives before an action is released. This separation of observation ingestion, relational reasoning, and short-horizon evaluation makes the model measurable at each stage and supports engineering inspection of rejected or accepted schedules.

The remainder of the paper links the methodological contribution to its engineering evidence through related work, model construction, controlled twin experiments, and deployment-oriented interpretation. Section 2 reviews quay-crane scheduling, digital-twin dispatch, robust multimodal fusion, Perceiver-style attention, and graph learning for terminal operations. Section 3 introduces the proposed robust feature fusion mechanism, the Perceiver IO GNN architecture, the scheduling objective, and the implementation workflow, and reserves Figure 1 and Figure 2 for the model architecture and decision flow, respectively. Section 4 shows the experimental twin, scenario construction, baselines, a single comparison table and six data figures of the scheduling performance, robustness, energy behavior and ablation effects. Section 5 presents the main results of the study and its limitations and future research directions.

2. Related Work

2.1. Quay Crane Scheduling and Digital Twin Dispatch

Traditionally, the formulation of quay crane scheduling has been a constrained optimization problem that takes into account task precedence, crane non-crossing, safety distance, bay-dependent processing time, and vessel service priority. Exact mixed-integer formulations are still useful for small- and medium-sized problems because they can show the structure of interference and precedence constraints [12]. Heuristic and metaheuristic methods are generally used for large terminals to generate feasible schedules rapidly under the constraints of actual dispatch [13]. Rolling-horizon methods have increased the flexibility of operation by adjusting the schedule in response to new vessel and yard events; however, they may exhibit oscillations when the state estimator is noisy or frequent disruptions occur outside the optimization cycle [14]. Recently, learning-assisted methods have been used to predict task durations and select neighborhood moves or approximate dispatch policies; however, many still treat crane features as clean observations and fail to consider the reliability of multimodal port data [15].

Digital Twin Scheduling combines the above methods to connect real-world equipment and facilities with simulations and decision-making models. A terminal twin can show the connections among vessel services, crane operations, yard truck queues, energy consumption, etc., to help evaluate the dispatch policy at the berth face before its implementation. Existing twin studies often emphasize visualization, state synchronization, or what-if simulation while retaining a conventional scheduling optimizer. The unresolved gap is therefore twofold: irregular sensor streams are not always converted into reliability-aware operational states, and scheduling decisions are seldom linked back to the uncertainty of the observations that produced them. Many twin workflows also assume a stable and complete state vector. Signal loss caused by equipment controllers, video misclassification and asynchronous updates of the yard in active terminals may introduce bias to the twins if not explicitly handled by the model.

Dispatch quality should not be judged solely by the time of final service completion, according to the operational literature. A slightly faster schedule that is frequently changed will be difficult for crane operators and yard transport teams to implement, and a very stable schedule will lead to the idle of expensive equipment during short congestion windows. Therefore, engineering-oriented research has gradually included utilization rate, rescheduling stability, energy consumption and frequency of infeasible actions in vessel turnaround. The above indicators are required for neural scheduling because, otherwise, the learned model may exploit hidden simplifications in the simulator to produce policies

that appear effective in simulation but are unstable when applied to real equipment. The objectives and outcomes of the experiment should directly correspond to the trade-offs in the digital twin scheduler for systems with high requirements for safety separation and human acceptance.

2.2. Robust Fusion and Neural Relational Scheduling

Robust multi-modal fusion is now widely employed in transportation, robotics and industrial monitoring to combine sensors with different uncertainty characteristics. Huang et al. [16] showed that uncertainty quantification and reliability learning can reduce the impact of corrupted multimodal signals before they enter the prediction or optimization layer. The problem of data fusion in port scheduling is aggravated by high-frequency machine telemetry and sparse event records, and a simple average may fail to identify the reason for a delay. Therefore, the fusion layer for quay crane scheduling should keep the operational meaning of crane productivity, bay workload, berth congestion, and yard transfer pressure, and these should be separate features rather than being combined into an opaque vector too early in the pipeline.

Lebonvallet et al. [17] demonstrated that Perceiver IO can process large-scale, heterogeneous signals efficiently by mapping the input array into a fixed latent space. Graph neural networks are used to solve problems involving relationships between entities in the construction of a scheduling feasible region [18]. Together, the two models can learn from irregular observation sequences and, at the same time, respect the graph structure of cranes, berths, tasks, and vessels. Recently, research in neural optimization has shown that the learned dispatch policies can be as effective as hand-crafted rules after several decision cycles; however, engineering application still requires interpretable indicators, disturbance tests and ablation analyses, not just single-scenario accuracy reports [19]. Therefore, this paper introduces a robust fusion Perceiver IO GNN model to address the shortcomings mentioned above in sensor dropout, congestion and productivity perturbation.

The last problem is the interface of perception and optimization. Most neural dispatch studies use a representation model to output a dense embedding, and then, in the decision layer, convert it to an action probability. Direct mapping is compact; however, it is not very feasible to ensure schedule accuracy and to identify the cause of any deviations. Port scheduling is based on an explicit intermediate structure, fused observations serve as the latent operating context, and this latent context is then queried by graph nodes that are still associated with cranes, bays, vessels, and task batches. This separation allows an abnormal recommendation to be inspected through the source reliability score, the latent query response, the binding graph constraint, and the objective components used to rank the candidate. The neural scheduler is therefore treated as an auditable decision module rather than an unrestricted black box.

3. Proposed Perceiver IO GNN Optimization Framework

3.1. Digital Twin State Construction and Robust Feature Fusion

The proposed framework is a continuously updated collection of operational entities for the smart port digital twin, not a fixed-state matrix. At each decision epoch t, the twin is provided with the following information: estimated completion times of vessels, bay workloads, quay-crane positions, spreader cycle times, yard-truck queues, weather conditions, safety-zone occupancy, and equipment maintenance status. These signals differ in sampling rate, numerical range, unit, and measurement reliability. Vessel information may be updated every few minutes, whereas programmable logic controller (PLC) signals can be generated at sub-second intervals. Weather and maintenance are relatively rare, but they can change the feasible scheduling area suddenly. At each five-minute decision epoch, the model generates entity tokens containing entity type, source identity, timestamp, observation age, unit-aware normalized value, missingness status, and a reliability descriptor. Raw streams retain their native sampling rates and are aligned only when the decision state is assembled.

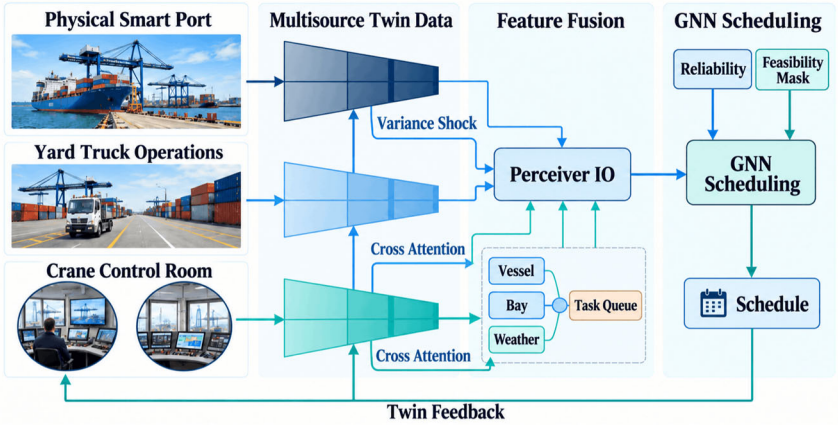


Figure 1. Robust Perceiver IO-GNN architecture for smart-port quay-crane twin scheduling

Figure 1 is the resulting architecture. Observations of the physical port are converted into multi-source twin tokens, processed by the robust fusion module, compressed into a fixed latent representation via Perceiver IO, and then passed to the graph scheduling layer. The generated schedule is provided to the digital twin for a short-term operational impact analysis and then applied.

Operational Normalization is used before feature fusion. Crane position is relative to the index of the corresponding vessel bay, trolley displacement is normalized by the feasible movement range, and handling time is divided by the recent median cycle time of the relevant crane. Yard pressure is represented by the absolute truck count and its percentile within the current shift. This paired encoding distinguishes an exceptional queue in a low-volume period from a similar queue during a scheduled vessel wave. Safety-related observations are still explicit binary or distance variables because the graph decoder needs to use them to construct hard feasibility masks. Weather categories and maintenance statuses are coded as disturbance descriptors rather than general productivity attributes because they affect both the processing time and the availability of operations.

For source i, the reliability score combines observation age $a_i(t)$, missingness $m_i(t)$, short-horizon variance shock $v_i(t)$, physical inconsistency $c_i(t)$, and cross-source disagreement $d_i(t)$. The reliability-estimation equation is

$$r_i(t) = b_i - \gamma_a a_i(t) - \gamma_m m_i(t) - \gamma_v v_i(t) - \gamma_c c_i(t) - \gamma_d d_i(t), \quad \gamma_a, \gamma_m, \gamma_v, \gamma_c, \gamma_d \geq 0 \tag{1}$$

where b_i is the prior reliability of the sensing source and the nonnegative coefficients control the sensitivity to each degradation indicator. Observation age is measured relative to the expected update interval of the source rather than in absolute seconds. The variance term compares the current feature variation with its recent operational envelope, while physical inconsistency identifies states such as an indicated crane displacement that exceeds the attainable motion during one sampling interval. Cross-source disagreement is evaluated only between operationally related observations, such as crane cycle time, completed lifts, and vessel-bay workload.

The reliability scores are converted into normalized source weights and used to form the fused token representation. The feature-fusion equation takes the form

$$\boldsymbol{z}_{e,t} = \sum_{i \subset \mathcal{S}_e} \alpha_i(t) \phi_i\left(\boldsymbol{x}_{i,t}, \tau_{i,t}\right), \alpha_i(t) = \frac{exp(r_i(t)/\eta)}{\sum_{j \in \mathcal{S}_e} exp\left(r_j(t)/\eta\right)} \tag{2}$$

where $\mathcal{S}_e$ denotes the sources associated with entity e, $\phi_i(\cdot)$ maps a source observation into the shared operational feature space, $\tau_{i,t}$ is its temporal encoding, and η controls the concentration of the reliability weights. The normalized weighting mechanism reduces the contribution of an unreliable channel without deleting the complete entity state. Consequently, a drifting crane-position sensor can be downweighted while cycle-time, lift-count, and berth-camera observations continue to support the corresponding crane token.

A missing observation is shown by a learned mask embedding and the time since the last valid update. Therefore, when a sensor is temporarily unavailable, it will not be taken as zero by default. Reliability weights are also capped during training to prevent the weight of a single, very reliable channel in the fused state from rising too much. The fused entity tokens are still linked to the original reliability indicators, and therefore can help the downstream scheduler differentiate between a low-workload prediction based on consistent observations and one obtained under high uncertainty.

3.2. Perceiver IO-GNN Scheduling Architecture

After robust fusion, the heterogeneous token set is processed by Perceiver IO. A set of 128 latent vectors with width 256 interacts with all input tokens through eight-head cross-attention; four latent self-attention blocks then refine the global operational context using GELU activation and dropout of 0.1. The number of active vessel, crane, bay, and task tokens can vary throughout the day without changing the latent-array dimensions. This property is important for quay-crane scheduling because vessel arrivals, completed task batches, temporary crane outages, and yard-side disturbances continuously change the size and composition of the input set.

The latent-update equation combines cross-attention and self-attention within one residual transformation:

$$\boldsymbol{L}^{k+1} = \boldsymbol{L}^k + SelfAttn\left(\boldsymbol{L}^k + CrossAttn\left(\boldsymbol{L}^k, \boldsymbol{Z}_t, \boldsymbol{Z}_t\right)\right) \tag{3}$$

where $\mathbf{Z}_t$ is the fused token set and $\mathbf{L}^k$ is the latent operational state at layer k. Cross-attention transfers information from the variable-size input set into the fixed latent space, whereas self-attention models system-level dependencies among berth workload, crane productivity, yard pressure, and vessel priority. Task and crane output queries then retrieve

scheduling-specific features from the final latent state. Positional information is represented through berthrelative coordinates, and temporal information is encoded through both absolute decision time and elapsed time from the latest valid observation.

The scheduling graph contains crane nodes, vessel nodes, bay nodes, and task-batch nodes. Crane-to-crane edges describe adjacency and potential interference. Crane-to-task edges represent candidate assignments and expected travel distance. Task-to-task edges preserve precedence relations, while vessel-to-task and bay-to-task edges encode workload ownership and spatial position. Yard-transfer coupling is introduced through edges between task batches that compete for the same short-horizon truck capacity. Edge attributes include relation type, remaining workload, longitudinal distance, estimated processing time, safety margin, and current congestion pressure.

For node v, relational information is propagated using the graph message-passing equation

$$\boldsymbol{h}_v^{\ell+1} = \sigma\left(\boldsymbol{W}_0^{\ell}\boldsymbol{h}_v^{\ell} + \sum_{u\in\mathcal{N}(v)} \beta_{uv}^{\ell}\boldsymbol{W}_{e(u,v)}^{\ell}\boldsymbol{h}_u^{\ell}\right) \tag{4}$$

where $\mathbf{h}_v^{\ell}$ is the representation of node v at graph layer $\ell, \mathcal{N}(v)$ is its relational neighborhood, and $\mathbf{W}_{e(u,v)}^{\ell}$ is specific to edge type $e(u,v)$. The coefficient β_{uv}^{ℓ} depends on relation type, spatial distance, remaining workload, source reliability, and estimated interference risk. Relation-specific transformations prevent a safety-distance edge from being interpreted in the same manner as a task-precedence edge.

Two graph message-passing layers are used with 128-dimensional node states and 32-dimensional edge embeddings. The first layer transfers information between directly connected crane, task, bay, and vessel nodes, while the second allows a crane to receive context through shared tasks and neighboring cranes. This receptive field captures local interference and short-range workload coupling without excessively smoothing representations among distant cranes. Edge-type-specific transformations use GELU activation and dropout of 0.1. Each feasible crane-task pair is then scored from the crane state, task state, latent context, expected processing time, movement cost, and safety margin.

Figure 2 presents the rolling decision process. Every five minutes, the decoder produces candidate assignments, the feasibility checker removes invalid actions, and the twin simulates the remaining candidates over a 15-minute horizon. A revised plan is accepted only when its normalized objective improves on the released plan by at least 0.02 after the schedule-change penalty is applied. Candidate ties are resolved by lower safety exposure, fewer assignment changes, and shorter crane travel, in that order. If no candidate survives masking, the current feasible schedule is retained and the state is flagged for dispatcher review.

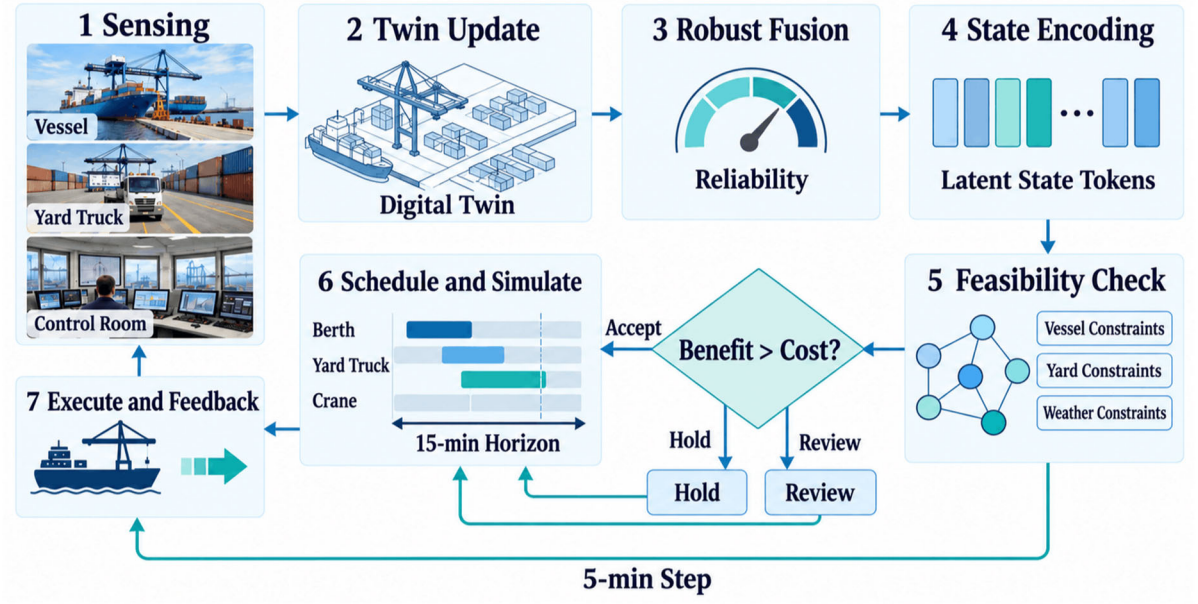


Figure 2. Twin-synchronized rolling optimization and schedule-feedback process

The feasibility mask blocks crane crossings, insufficient safety distances, unavailable equipment, incomplete task predecessors, and assignments that exceed short-horizon yard-transfer capacity. Masking is performed before probability normalization so that infeasible actions receive no assignment probability. The decoder returns up to five highest-scoring feasible schedules rather than a single deterministic action. The digital twin compares their delay, energy, stability, and

interference consequences; score ties follow the safety-first rule defined above, and an empty candidate set retains the released schedule.

3.3. Optimization Objective and Training Protocol

The scheduling decoder estimates an assignment probability for every feasible crane-task pair. The soft assignments are converted into an executable schedule through masked matching, subject to the condition that one crane cannot process multiple batches simultaneously and one task batch cannot be assigned to multiple cranes. The learning process combines supervised imitation of high-quality schedules with policy improvement in the digital-twin simulator.

A Multi-Objective Scheduling Problem Considering Vessel Delay, Crane Energy Consumption, Idle Time, Schedule Instability and Safety Risk.

$$\mathcal{J}(\theta) = \lambda_T \tilde{T}_{delay} + \lambda_E \tilde{E}_{crane} + \lambda_I \tilde{I}_{idle} + \lambda_S \tilde{S}_{change} + \lambda_R \tilde{R}_{safety} \tag{5}$$

where all objective terms are normalized by operational reference values. The delay term measures deviation from the planned vessel completion time. The energy term includes productive lifting, trolley movement, gantry travel, and standby consumption. The idle-time term penalizes avoidable nonproductive intervals, whereas the schedule-change term measures the number and magnitude of revisions relative to the previously released plan. The safety term increases as neighboring cranes approach the interference boundary, even when the candidate schedule remains technically feasible.

The schedule-change penalty is asymmetric. Reassigning an unstarted task incurs a moderate cost, interrupting an active batch incurs a larger cost, and reversing an announced crane path receives the largest penalty because it can disrupt operators and yard transport. The safety term increases continuously as neighboring cranes approach the interference boundary, while the deterministic mask still blocks any action that violates crossing, distance, availability, precedence, or yard-capacity constraints.

Training states are augmented with missing-channel masks, delayed vessel updates, weather-dependent cycle-time inflation, crane productivity drift, and yard-queue shocks. The disturbance set $\mathcal{D}$ defines bounded but operationally meaningful deviations from the nominal twin state. Robust learning is governed by the following min-max equation:

$$\theta^* = arg\min_{\theta} \left[\mathbb{E}_{\boldsymbol{Z}_t} \mathcal{J}(\theta; \boldsymbol{Z}_t) + \rho \max_{\delta \subset \mathcal{D}} \mathcal{J}(\theta; \boldsymbol{Z}_t + \boldsymbol{\delta}) \right] \tag{6}$$

where $\boldsymbol{\delta}$ represents an admissible disturbance and ρ determines the importance of adverse operating states during optimization. Disturbances are applied selectively rather than independently to every variable. For example, yard congestion jointly changes truck availability, transfer delay, and the effective productivity of bays supplied by the affected yard block. Sensor dropout modifies the observation mask and reliability descriptors but does not directly alter the underlying physical state.

Training is conducted in three stages. The first stage learns reliability scoring and masked feature reconstruction from digital-twin logs using both random packet masks and structured channel outages. The second stage trains the Perceiver IO-GNN decoder by imitating high-quality schedules generated by the offline rolling optimizer and filtered by deterministic feasibility rules. The third stage performs simulator fine-tuning under nominal and disturbed states. AdamW is used with an initial learning rate of 2e-4, batch size 32, weight decay 1e-4, and gradient clipping at 1.0. Congestion, equipment degradation, irregular arrivals, and sensor dropout are oversampled during training, while validation weights preserve the intended operating distribution.

The third stage fine-tunes the entire model in the simulator using the multi-objective and robust-training equations. Disturbance scenarios are sampled more frequently than their natural occurrence rate, and importance weights restore the target operating distribution during validation. Early stopping is triggered when the disturbed-scenario completion rate, feasibility ratio, and revision frequency fail to improve for 12 validation epochs. Neural inference produces candidate scores in milliseconds, decisions are issued every five minutes, and the twin evaluates each candidate over a 15-minute simulation horizon before releasing the highest-ranked executable action.

4. Experimental Data and Engineering Analysis

4.1. Experimental Twin, Baselines, and Evaluation Data

The experimental environment represents a coastal automated container terminal with 1,250 m of berth, 9 quay cranes, 42 vessel calls, and 1,260 task batches. The task batches aggregate bay-level container moves into executable dispatch units, with processing times sampled from crane PLC distributions calibrated for low, medium, and high congestion windows. The twin records vessel arrival deviation, crane cycle time, trolley travel distance, spreader load state, truck queue length, yard block congestion, and wind category. Disruption-oriented port studies recommend evaluating

correlated operating shocks rather than reporting only nominal schedules [20]. Accordingly, the experiment includes nominal operation, a peak arrival wave, yard congestion, 10% sensor dropout, and 20% sensor dropout. The baselines are an earliest-feasible rule, a rolling mixed-integer optimizer with a 45 min time budget, a pure GNN scheduler, and a transformer scheduler without explicit robust fusion.

The dataset is partitioned by vessel call rather than by individual task batch to avoid leakage between training and testing. Thirty calls are used for training, six for validation, and six for testing; every test call is replayed under all five scenarios with identical berth geometry and workload. The peak wave compresses mean inter-arrival time from 68 to 39 min between 09:30 and 15:30, whereas the yard-congestion case raises truck queue formation by 30% and closes two-yard blocks temporarily. Dropout is applied independently to AIS, crane PLC, and truck appointment streams, while physical crane position and safety-distance signals remain available. Turnaround time, idle time, energy index, robust completion rate, infeasible attempts, and revisions per vessel are recorded. Because rolling optimization responsiveness is constrained by the time available between state updates [21], the mixed-integer baseline is evaluated under the same fixed computation budget at every replay. All learned baselines use the identical vessel-level split and are evaluated with five random seeds; reported values are means across matched scenario replays.

Table 1 reports the aggregate descriptive results across matched scenario replays. The proposed model achieves a mean vessel turnaround time of 16.9 h, compared with 20.8 h for the rule-based baseline, 17.8 h for the pure GNN, and 18.4 h for the transformer scheduler. Its crane idle rate is 11.4%, which is 4.8 percentage points below the rule-based result, and its disturbed-scenario completion rate reaches 93.1%. The comparatively small decline from nominal to disturbed completion indicates that the gain is not restricted to an easy operating regime. The mixed-integer optimizer remains competitive in nominal turnaround, but its longer solve cycle provides fewer opportunities to react before a queue shock propagates from the berth to the yard. These comparisons are interpreted as simulation-based performance differences rather than field-validated causal effects.

Table 1. Comparative scheduling performance across the smart port twin dataset

Method	Turnaround *h*	Idle time *%*	Energy index	Robust completion *%*	Plan revisions
Rule based heuristic	20.8	14.5	1.00	81.4	4.7
Rolling MIP optimizer	18.1	12.9	0.95	88.6	3.9
Pure GNN scheduler	17.8	13.2	0.94	87.2	4.2
Transformer scheduler	18.4	12.1	0.93	89.5	3.6
Proposed Perceiver IO GNN	16.9	11.4	0.902	93.1	2.8

Figure 3(a) separates the load trajectory from the method comparison: active vessel calls rise from 3 at 08:00 to 7 at 12:00 and 14:00, while the pending queue peaks at 11 batches at 12:00 before falling to 5 by 20:00. The two-series separation shows that queue pressure starts increasing before the arrival count reaches its maximum, giving the rolling scheduler a measurable early-warning interval. Figure 3(b) shows a median batch service time of 8.7 min in the nominal case, followed by 9.4, 10.2, 10.5, and 10.9 min across the four disturbed cases. Service-time distributions are commonly used to distinguish recurrent productivity variation from exceptional operational delay [22]. Here, the upper extent crosses the 12 min risk threshold in every disturbed case and reaches 14.4 min at 20% dropout, whereas the median rises more gradually; the widening upper tail therefore matters more to queue propagation than the shift in the center alone.

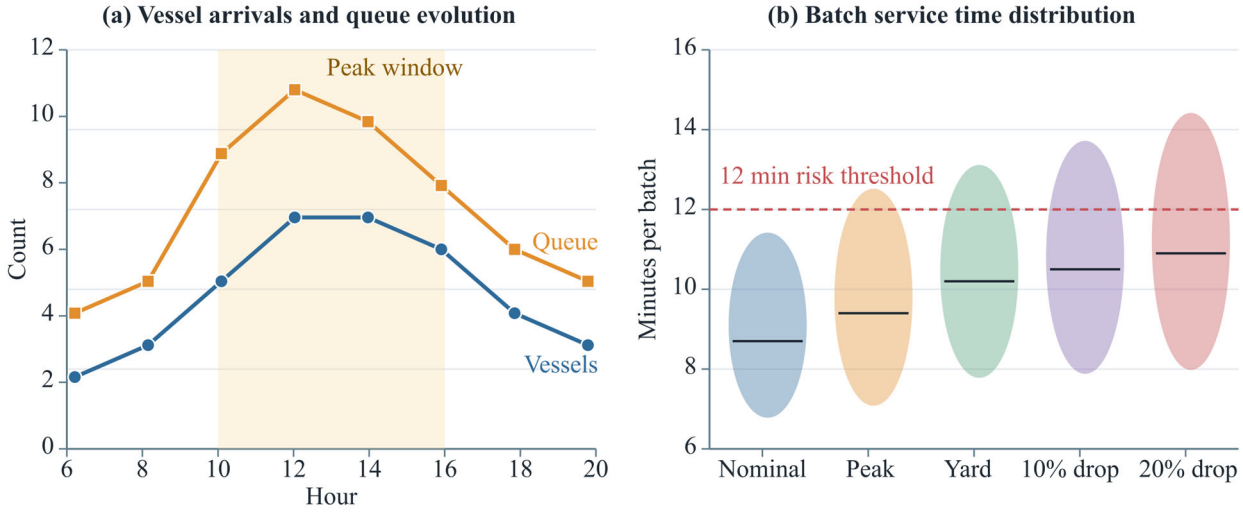


Figure 3. Operational load and service time distribution in the experimental twin: (a) vessel arrivals and task queue evolution; (b) crane batch service time under five scenarios

The pressure trace also explains how the proposed policy generates the aggregate gains in Table 1. During the peak-wave period, active vessel calls are 75% higher than in the early morning and the mean yard-truck queue is 46% longer. Across congested decision epochs, the proposed model assigns the first available crane to the nearest high-priority bay in 38% of cases; in the remaining epochs it rejects that locally attractive action when the twin predicts crane adjacency conflicts or yard starvation. By comparison, the minimum-waiting rule selects the nearest eligible berth-side task in 71% of its decisions, concentrating work around the central berth. Of all revisions released by the proposed model, 64% occur before the pending queue reaches 11 batches. The timing is associated with the divergence between arrival and queue trajectories and is interpreted as an early response to congestion rather than proof of a unique causal mechanism.

4.2. Scheduling Accuracy, Robustness, and Ablation Analysis

Reliability-aware fusion should lower confidence when independent signals disagree, rather than merely when the terminal becomes busy [23]. Figure 4(a) shows that the complete model requires 41 ms per decision, compared with 44 ms for the Perceiver-only variant, 49 ms for the pure GNN, and 55 ms for the transformer; all remain below the 60 ms online budget. Its feasible-action ratio is 98.6%, 4.4 percentage points above the pure GNN result, so the latency reduction is accompanied by fewer masked or rejected assignments.

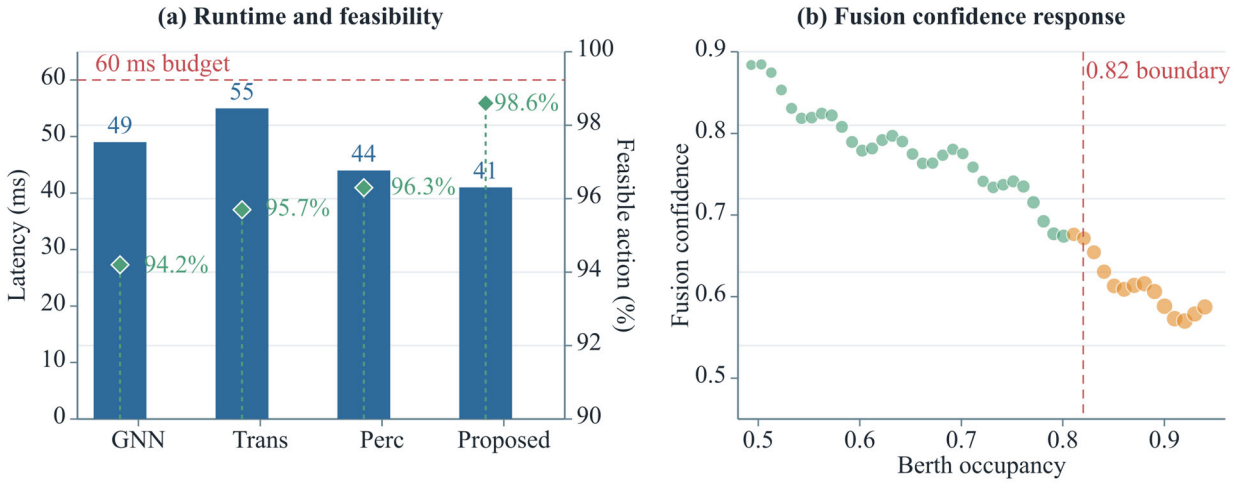


Figure 4. Runtime and feature reliability behavior: (a) inference latency and feasible action ratio by model; (b) fusion confidence, berth occupancy, and yard queue pressure

Figure 4(b) places confidence against berth occupancy while point size and color encode yard queue pressure. Confidence remains predominantly above 0.75 below 0.82 occupancy, but drops toward 0.55-0.65 when occupancy exceeds 0.82 and queues exceed 28 trucks. The concentration is therefore associated with coupled congestion and cross-source inconsistency, not a uniform penalty across the state space.

Component ablation is used to identify whether performance comes from representation capacity, relational constraints, or regularization rather than parameter count alone [24]. Figure 5(a) gives the complete model normalized scores of 0.91 for delay control, 0.88 for energy efficiency, 0.94 for feasibility, 0.90 for stability, and 0.93 for robustness; the weakest axis is energy, revealing that travel reduction remains harder than constraint satisfaction. Figure 5(b) separates contribution across five operating conditions. Total turnaround reduction rises from 12.4 percentage points nominally to 18.7 points in the peak wave, with corresponding totals of 17.2, 15.8, and 16.9 points under yard congestion, 10% dropout, and 20% dropout. Robust fusion contributes 3.2 points nominally, 6.1 points in the peak wave, and 7.0 points at 20% dropout, whereas graph reasoning reaches its largest contribution of 5.6 points under yard congestion. Figure 5(c) shows that removing the stability penalty increases revisions from 2.8 to 5.9 per vessel, while removing fusion raises them to 4.8, which means noisy state estimates also induce plan churn. Figure 5(d) reports the largest fusion gain at 20% dropout: completion increases from 77.2% to 93.1%, a 15.9-point recovery, compared with gains of 9.6, 9.1, and 8.3 points in the other disturbed settings.

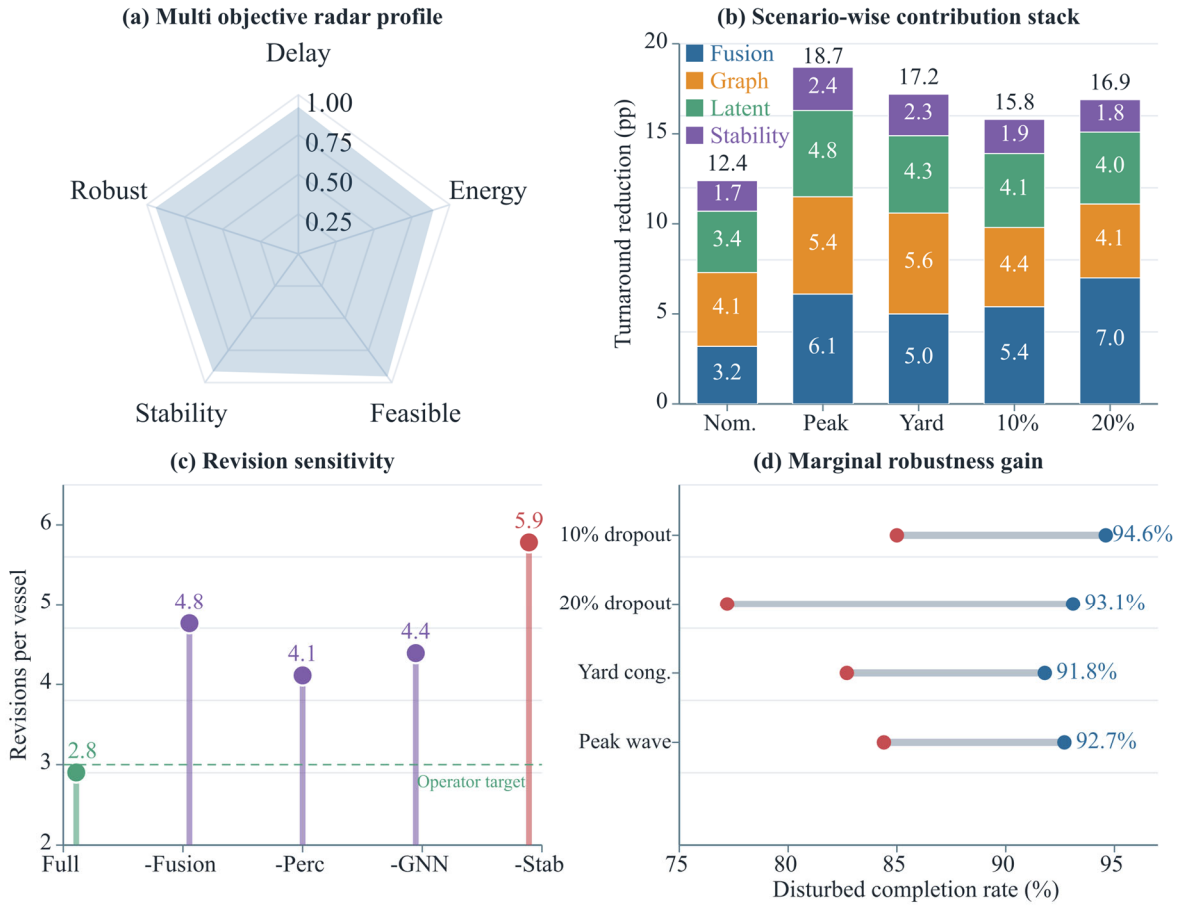


Figure 5. Ablation and contribution analysis: (a) normalized multi objective radar profile; (b) scenario-wise stacked contributions to turnaround reduction; (c) lollipop comparison of rescheduling revisions by ablation setting; (d) marginal robustness gain under dropout and congestion

Energy-aware crane scheduling should consider both the effect of travel and the demand pressure from downstream transfer; otherwise, a short quay-side move may still result in idle trucks [25]. Figure 6(a) is a map of crane-bay interference risk, and only five cells are above 0.30; the maximum is 0.42 at crane 2 and bay zone 4, and cranes 5 and 6 are below 0.20 in all zones. Because berth geometry and safety-distance constraints are asymmetric, the graph mask does not distribute central-bay workload uniformly. Figure 6(b) shows the normalized energy response over trolley travel distance and yard queue length. The response remains below 1.05 when travel is less than approximately 82 m and the queue contains fewer than 22 trucks, but it rises as both factors increase. At a queue length near 34 trucks, the low-energy region is no longer sustained even at moderate travel distances. This controlled response supports inclusion of yard coupling in the scheduling objective rather than treating it only as an external diagnostic variable.

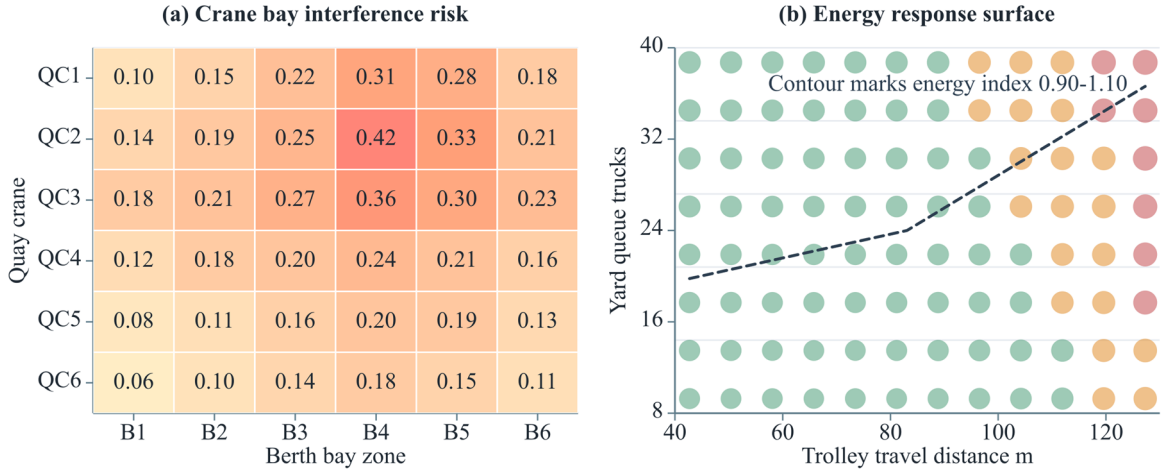


Figure 6. Spatial interference and energy response: (a) crane bay interference heatmap; (b) energy response surface over trolley travel distance and yard queue length

The isolated-disturbance replays clarify the ablation patterns. With fusion active, a 15-minute AIS delay reduces assignment volatility by 12.6% because the model shifts weight toward crane PLC and berth-occupancy streams. Under cycle-time drift, the reliability weight of the corrupted productivity feature falls from 0.31 to 0.12 within three decision cycles. Perceiver IO contributes most when active task tokens increase from 164 to 247 during the peak wave, whereas graph message passing removes 43 neighboring-crane conflicts over the test horizon. Removing fusion adds 0.9 h in the yard-congestion case through misestimated productivity, while removing graph reasoning adds 1.2 h through safety-distance and travel conflicts. These are controlled component-removal effects: fusion governs state credibility, latent attention handles variable-size context, and the graph enforces relational feasibility.

4.3. Disturbance Scenario Analysis and Engineering Interpretation

Disturbance evaluation is most informative when performance degradation and its operational causes are reported together [26]. Figure 7(a) shows that proposed-model turnaround rises from 15.8 h nominally to 18.6 h at 20% dropout, an increase of 2.8 h, whereas the pure GNN rises from 17.6 to 23.4 h, an increase of 5.8 h.

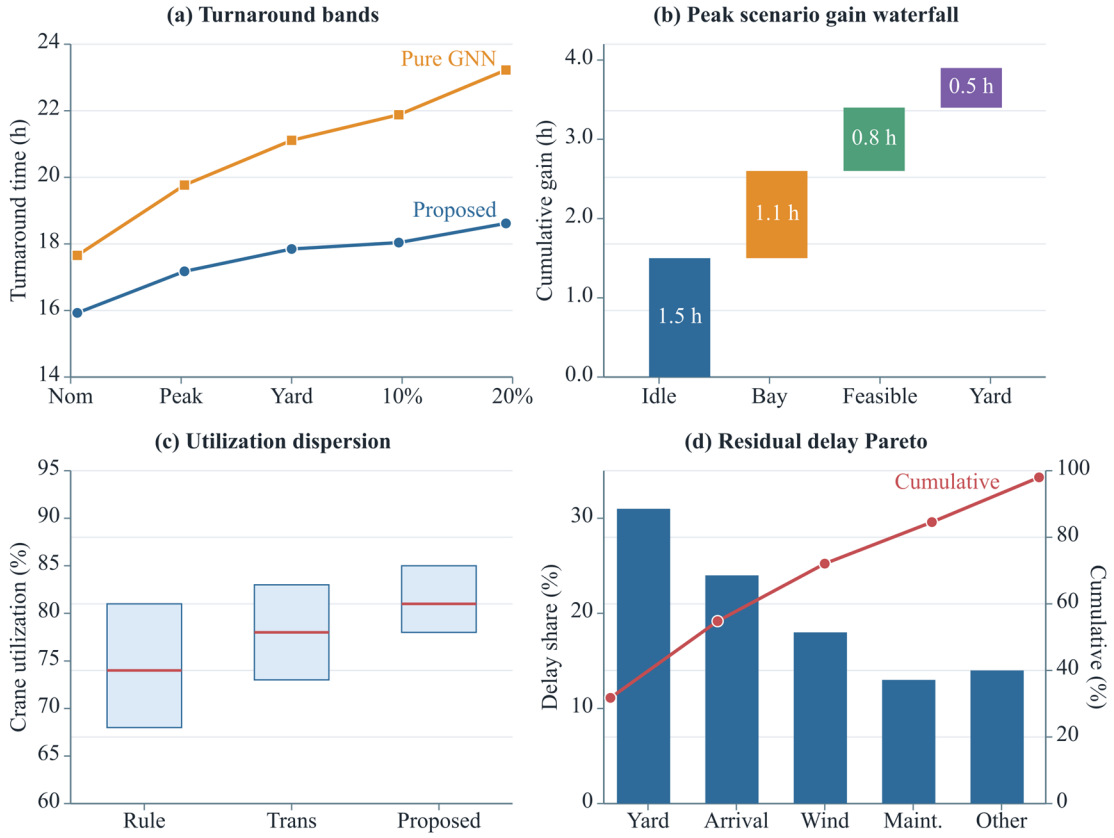


Figure 7. Disturbance and residual delay analysis: (a) turnaround time bands across scenarios; (b) waterfall decomposition of peak scenario gain; (c) crane utilization distribution; (d) residual delay Pareto profile

Figure 7(b) decomposes the 3.9 h peak-wave gain into 1.5 h from reduced idle periods, 1.1 h from improved bay sequencing, 0.8 h from fewer infeasible reassignments, and 0.5 h from yard alignment; idle reduction is the largest contributor but accounts for less than half of the gain. Figure 7(c) shows an interquartile crane-utilization range of 7 percentage points for the proposed model, compared with 10 for the transformer and 13 for the rule baseline. Figure 7(d) assigns 31% of residual delay to yard saturation, 24% to arrival bunching, 18% to wind, 13% to maintenance, and 14% to other causes; the first three sources therefore account for 73% of the remaining delay.

Calibrated benefit estimates are important in human-supervised scheduling because acceptance depends on whether predicted improvements are realized consistently [27]. Figure 8(a) compares rolling intervals from 5 to 30 min: the 5 min setting gives 16.5 h turnaround but 7.1 revisions per vessel, while the selected 15 min setting gives 16.9 h and 2.8 revisions. Moving from 15 to 30 min saves only 0.9 revisions but adds 1.3 h of turnaround, so the stability gain becomes disproportionately expensive. Figure 8(b) compares predicted and realized improvement. The proposed model follows the calibration diagonal with an expected calibration error of 0.041, versus 0.086 for the transformer and 0.103 for the pure GNN. Under 20% dropout, it makes 11 predictions above 12% improvement and realizes at least half of the predicted value in 10 cases; the transformer makes 19 such predictions but falls below half in 8 cases.

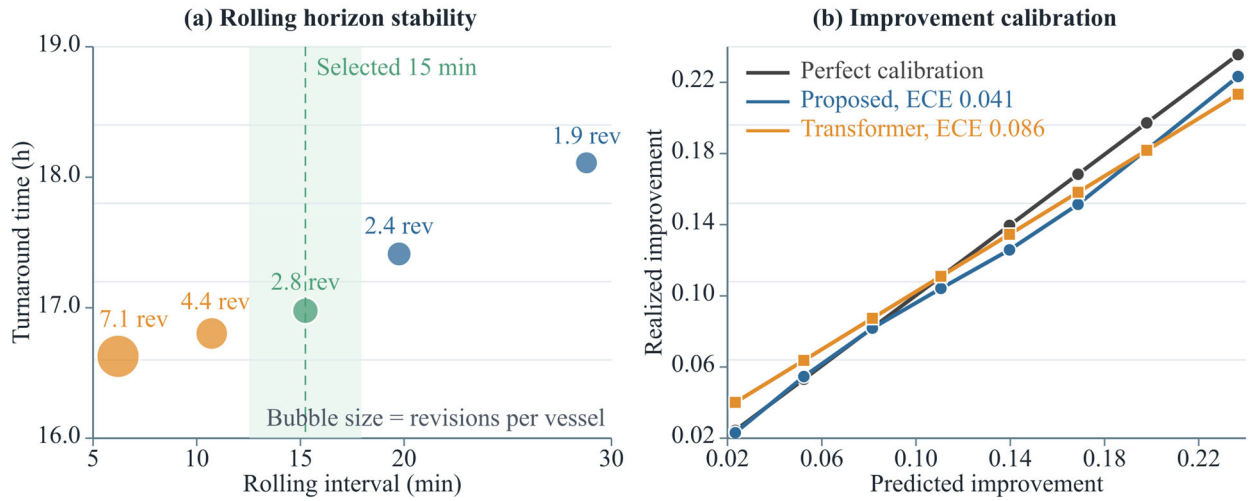


Figure 8. Deployment readiness indicators: (a) rolling horizon stability under different decision intervals; (b) calibration of predicted versus realized scheduling improvement

Inference for online twin control needs to be within the state-update cycle to some extent so that simulation and feasibility checks can still be performed [28]. The observed neural latency of 41 ms is small relative to the five-minute decision interval and leaves most of the online budget for twin rollout and deterministic feasibility checks. Reliability weighting avoids strong recommendations when the AIS and PLC streams are different, and a graph mask prevents the predicted gain from exceeding crane-spacing limits. Berth and yard operations are also in a physical connection, so crane sequencing cannot reduce transfer saturation without coordinating downstream capacity [29]. According to the test log, 31% of the residual delay is due to yard saturation, and increasing berth-side crane utilization beyond 84% during this time increases truck waiting by 18% without expediting vessel departure. A deployed decision-support interface should therefore display yard-pressure warnings, binding constraints, and candidate scores alongside crane assignments rather than presenting an isolated assignment optimum.

Commissioning studies suggest adapting multi-objective weights to the equipment, labor and service priorities of the terminal instead of using a single weight vector universally [30]. In the present sensitivity test, when the energy coefficient exceeds 0.22, the normalized crane energy is reduced by an additional 2.1%, but the mean vessel turnaround time increases by 0.7 hours. Reducing the stability coefficient below 0.08 lowers the immediate scheduling loss by 1.4% but increases revisions per vessel from 2.8 to 4.6. A 5-minute rolling window reduces the turnaround time by 0.4 hours but results in 7.1 revisions; a 30-minute window fails to cover 63% of queue shocks that dissipate in two decision cycles. The resulting commissioning region uses a 15-minute horizon, an energy coefficient of 0.18-0.22, and a stability coefficient of 0.08-0.12. These values should be recalibrated whenever crane-fleet composition, labor policy, or yard-dispatch rules change.

5. Conclusion

This paper develops a robust feature fusion Perceiver IO-GNN model for smart-port quay-crane digital-twin scheduling. The framework integrates asynchronous terminal observations, reliability-aware state construction, latent multimodal encoding, graph-based feasibility reasoning, and twin-evaluated rolling assignment. The study demonstrates how sensing uncertainty, crane interference, task precedence, energy use, schedule stability, and safety constraints can be represented within one decision-support workflow.

The study remains limited by its calibrated simulation setting. It has not yet been connected to a live terminal operating system, and dispatcher intervention, labor-shift rules, maintenance windows, contractual vessel priorities, communication failures, and emergency procedures are represented only in simplified form. The deterministic feasibility checker provides a clear safety boundary but may also exclude unconventional schedules that could prove feasible under richer operational models.

Future research should evaluate the model through hardware-in-the-loop and live terminal trials with dispatcher feedback, explicit fail-safe behavior, and measured interface latency. Additional work may extend the decision graph to joint berth-yard-gate coordination, update objective weights through controlled online adaptation, and provide explanations that identify the sensor evidence, graph constraints, and objective components behind each reassignment. These steps are required before the method can progress from simulated decision support to production deployment.

6. Declarations

6.1 Author Contributions

Rintaro Soma contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision, project administration, and funding acquisition. Genta Iwamoto contributes to validation, analysis, investigation, data collection, draft preparation. All authors have read and agreed with the manuscript before its submission and publication.

6.2 Funding and Acknowledgments

This research received no specific financial support from any funding agency.

6.3 Institutional Review Board Statement

Not applicable.

6.4 Informed Consent Statement

Not applicable.

6.5 Date Availability

Not applicable.

6.6 Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

6.7 Declaration of generative AI and AI-assisted technologies

During the preparation of this manuscript, the authors utilized Grammarly for linguistic polishing, logical organization and draft revision. The generative AI tool was not involved in research design, data collection, data analysis, or the equation of core conclusions. All outputs generated by AI were reviewed, revised and verified manually by the authors. The authors take full responsibility for all content of this manuscript.

7. References

[1] Paraskevas, A., Madas, M., Zeimpekis, V., & Fouskas, K. (2024). Smart ports in Industry 4.0: A systematic literature review. Logistics, 8(1), 28. https://doi.org/10.3390/logistics8010028

[2] Abril, D., Paternina-Arboleda, C. D., & Velasquez-Bermudez, J. (2024). An integrated event-driven real-time tactical-operational optimization framework for smart port operations planning. Logistics, 8(3), 65. https://doi.org/10.3390/logistics8030065

[3] Hakimi, F., Khaled, T., Al-Kharaz, M., Foahom Gouabou, A. C., & Amzil, K. (2024). Towards a digital twin modeling method for container terminal port. Procedia Computer Science, 246, 3113-3121. https://doi.org/10.1016/j.procs.2024.09.361

[4] Li, J., Chang, D., & Wen, F. (2025). Digital twin enhanced rescheduling based on hybrid strategy in intermodal container terminal. Computers & Operations Research, 180, 107053. https://doi.org/10.1016/j.cor.2025.107053

[5] Ran, L., Cheng, Y., & Zhang, G. (2025). Two-stage robust optimization for the integrated dynamic berth allocation and quay crane scheduling problem under uncertainty. Engineering Optimization, 57(11), 3378-3400. https://doi.org/10.1080/0305215x.2024.2435539

[6] Vahdani, B., Veysmoradi, D., Abyaneh, M. B., & Rashedi, M. (2024). Robust integrated multi-mode scheduling of flexible loading and unloading operations with maintenance services in a port container terminal. Ocean & Coastal Management, 259, 107481. https://doi.org/10.1016/j.ocecoaman.2024.107481

[7] Li, Y., Li, X., Zhang, C., & Wu, T. (2024). Decomposition algorithms for the robust unidirectional quay crane scheduling problems. Computers & Operations Research, 167, 106670. https://doi.org/10.1016/j.cor.2024.106670

[8] Folco, P., Sahli, A., Belmokhtar-Berraf, S., & Bouillaut, L. (2024). A rolling horizon for rolling stock maintenance scheduling problem with cyclical activities. Computers & Industrial Engineering, 196, 110460. https://doi.org/10.1016/j.cie.2024.110460

[9] Zhao, Y., Luo, X., & Zhang, Y. (2024). The application of heterogeneous graph neural network and deep reinforcement learning in hybrid flow shop scheduling problem. Computers & Industrial Engineering, 187, 109802. https://doi.org/10.1016/j.cie.2023.109802

[10] Tong, G., Ge, Z., & Peng, D. (2024). RSMformer: An efficient multiscale transformer-based framework for long sequence time-series forecasting. Applied Intelligence, 54(2), 1275-1296. https://doi.org/10.1007/s10489-023-05250-8

[11] Xu, Z., Chen, X., Li, Y., & Xu, J. (2024). Hybrid multimodal feature fusion with multi-sensor for bearing fault diagnosis. Sensors, 24(6), 1792. https://doi.org/10.3390/s24061792

[12] Chu, L., Zhang, J., Chen, X., & Yu, Q. (2024). Optimization of integrated tugboat-berth-quay crane scheduling in container ports considering uncertainty in vessel arrival times and berthing preferences. Journal of Marine Science and Engineering, 12(9), 1541. https://doi.org/10.3390/jmse12091541

[13] Eldemir, F., & Taner, M. E. (2025). A hybrid meta-heuristic approach for solving single-vessel quay crane scheduling with double-cycling. Journal of Marine Science and Engineering, 13(2), 371. https://doi.org/10.3390/jmse13020371

[14] Ran, L., Liu, B., Zhang, G., & Cheng, Y. (2025). An improved particle swarm optimization algorithm for berth allocation and time-variant quay crane scheduling problem during an emergency. Expert Systems with Applications, 269, 126406. https://doi.org/10.1016/j.eswa.2025.126406

[15] Wang, Q., Tong, X., Li, Y., Wang, C., & Zhang, C. (2025). Integrated scheduling optimization for automated container terminal: A reinforcement learning-based approach. IEEE Transactions on Intelligent Transportation Systems, 26(7), 10019-10035. https://doi.org/10.1109/tits.2025.3556882

[16] Huang, L., Ruan, S., Decazes, P., & Denœux, T. (2025). Deep evidential fusion with uncertainty quantification and reliability learning for multimodal medical image segmentation. Information Fusion, 113, 102648. https://doi.org/10.1016/j.inffus.2024.102648

[17] Lebonvallet, G., Salazar-Zendeja, L. A., Hnaien, F., Snoussi, H., & Nélain, B. (2025). Perceiver IO model for efficient vibration signal compression and anomaly detection. IEEE Transactions on Instrumentation and Measurement, 74, 1-12. https://doi.org/10.1109/tim.2025.3604141

[18] Zhao, S., & Gu, S. (2024). A neural network algorithm framework based on graph structure for general combinatorial optimization. Neurocomputing, 587, 127670. https://doi.org/10.1016/j.neucom.2024.127670

[19] Guo, X., Zhang, P., Cai, Q., & Zhang, Y. (2025). Learning to solve combinatorial optimization problems with heterophily. Neural Networks, 189, 107554. https://doi.org/10.1016/j.neunet.2025.107554

[20] Moszyk, K., Janowczyk, O., & Deja, M. (2024). Simulation-based approach to a more efficient handling strategy for a container terminal storage yard: Case study of the Baltic Hub Container Terminal. Polish Maritime Research, 31(2), 121-131. https://doi.org/10.2478/pomr-2024-0028

[21] Wang, F., Wang, P., Hao, X., Yang, R., & Xu, R. (2025). Collaborative rescheduling of train timetables to relieve passenger congestions in an urban rail transit network: A rolling horizon approach. International Journal of Transportation Science and Technology, 19, 193-210. https://doi.org/10.1016/j.ijtst.2024.09.001

[22] Wang, T., Gao, G., Wang, K., & Shi, J. (2024). Integrated operation models with quay crane maintenance in a container terminal. Ocean & Coastal Management, 251, 107101. https://doi.org/10.1016/j.ocecoaman.2024.107101

[23] Shao, Z., Wang, H., Cai, Y., Chen, L., & Li, Y. (2025). UA-Fusion: Uncertainty-aware multimodal data fusion framework for 3-D object detection of autonomous vehicles. IEEE Transactions on Instrumentation and Measurement, 74, 1-16. https://doi.org/10.1109/tim.2025.3548184

[24] Tang, H., & Dong, J. (2024). Solving flexible job-shop scheduling problem with heterogeneous graph neural network based on relation and deep reinforcement learning. Machines, 12(8), 584. https://doi.org/10.3390/machines12080584

[25] Luo, Y., Liang, X., Zhang, Y., Tang, K., & Li, W. (2024). Energy-aware integrated scheduling for quay crane and IGV in automated container terminal. Journal of Marine Science and Engineering, 12(3), 376. https://doi.org/10.3390/jmse12030376

[26] Zheng, H., Wang, Z., & Wu, L. (2026). Disruption-responsive berth allocation and quay crane scheduling with inter-terminal collaboration. Expert Systems with Applications, 298, 129776. https://doi.org/10.1016/j.eswa.2025.129776

[27] Sobrie, L., & Verschelde, M. (2024). Real-time decision support for human-machine interaction in digital railway control rooms. Decision Support Systems, 181, 114216. https://doi.org/10.1016/j.dss.2024.114216

[28] Alsakka, F., Yu, H., El-Chami, I., Hamzeh, F., & Al-Hussein, M. (2024). Digital twin for production estimation, scheduling and real-time monitoring in offsite construction. Computers & Industrial Engineering, 191, 110173. https://doi.org/10.1016/j.cie.2024.110173

[29] Guo, L., Zheng, J., Du, J., Gao, Z., & Fagerholt, K. (2024). Integrated planning of berth allocation, quay crane assignment and yard assignment in multiple cooperative terminals. Transportation Research Part E: Logistics and Transportation Review, 183, 103456. https://doi.org/10.1016/j.tre.2024.103456

[30] Yang, Z., Wang, M., Wang, S., & Zhen, L. (2025). Optimizing continuous-time berth allocation, time-variant quay crane and yard assignment. Transportation Research Part B: Methodological, 200, 103317. https://doi.org/10.1016/j.trb.2025.103317