AI-Driven Ship Scheduling via Single-Pass Online Reinforcement Learning

Farah Qistina binti Alnizam
School of Comp. Science
University of Nottingham
Semenyih, Selangor, Malaysia
hcyfa2@nottingham.edu.my

Jeremy Klement Jim
School of Comp. Science
University of Nottingham
Semenyih, Selangor, Malaysia
hfyjj2@nottingham.edu.my

***Abstract*—This study presents a reinforcement learning (RL) framework for real-time maritime scheduling across port rotations. Traditional approaches rely on static timetables or heuristics and often fail to respond effectively to congestion and weather disruptions. In contrast, the proposed method employs a self-comparative reward mechanism, where each ship is incentivised to improve its own historical best route completion time (RCT) rather than adhere to external schedules. An Advantage Actor-Critic (A2C) agent is trained online within an event-driven simulation environment that models storm-affected segments and berth contention across global ports. The agent continuously refines its queued ship selection and departure timing strategies throughout a single, multi-year simulation without episodic resets. Results show that the trained policy outperforms first-come-first-served (FCFS) baselines by 23%. These findings suggest that self-referenced RL offers a resilient and adaptive scheduling solution for dynamic maritime operations.**

Keywords— Maritime Logistics, Ship Scheduling, Port Rotation, Storm-Aware Routing, Event-Driven Simulation, Reinforcement Learning, Online Training, Single-Pass, Actor-Critic, Self-Comparative Reward

I. Introduction

Global shipping routes are frequently disrupted by port congestion, adverse weather, and fluctuating demand. Traditional scheduling techniques, including FCFS rules, often fail to adjust to dynamic environments. RL, by contrast, offers a data-driven, real-time decision-making paradigm that has shown promise in related tasks.

This work introduces a self-comparative RL framework for maritime scheduling. Instead of optimising against static deadlines, ships aim to improve their own best RCT across repeated voyages. An A2C agent is trained within a multi-year event-driven simulation to make two key decisions: ship prioritisation and departure timing. The environment models port queues, berth-dependent service times, and seasonal storm exposure.

To evaluate the effect of long-term learning on scheduling performance, the simulator was run afresh across progressively extended planning horizons. Each horizon was trained from scratch and benchmarked against static baselines to assess the impact of sustained learning over time. Notably, this yielded meaningful improvements without relying on episodic resets, making the approach well-suited for dynamic logistics planning in long-term maritime operations.

II. Literature review

RL has recently shown considerable potential in maritime scheduling applications due to its ability to adjust to dynamic conditions and learn from interaction. Zhen [9] demonstrated that deep RL can significantly reduce vessel delays in canal systems by learning optimised sequencing policies. Similarly, Wang [6] applied a deep Q-network (DQN) to berth allocation, reporting substantial reductions in waiting times over traditional heuristics. Zhang [8] provided a broad survey of RL applications in scheduling, noting its success in diverse areas such as manufacturing, cloud resource allocation, and logistics.

Prior efforts in maritime scheduling often rely on static optimisation methods or episodic RL frameworks that assume discrete operations with clearly defined end goals. Golias [1] introduced Just-In-Time (JIT) arrival strategies to reduce anchorage time using estimated time of arrival (ETA) planning, while Moradi [3] applied RL to route planning under weather uncertainty, learning continuous ship heading and speed adjustments.

A2C, a synchronous policy gradient method, has also been applied to dynamic scheduling problems, particularly where decision-making combines discrete and continuous elements. Zhang and Xu [7] utilised A2C to optimise traffic signal control in a logistics corridor near a seaport, demonstrating improved flow under varying load conditions.

Despite these advances, the literature lacks approaches that support continuous RL training in a persistent environment while simultaneously handling mixed discrete-continuous action spaces. In addition, few methods reward ships based on self-comparative metrics rather than external benchmarks. This gap motivates our design of a single-pass, online training framework that learns adaptively as operations unfold.

III. Environment Design

A custom event-driven simulator was developed to model ship scheduling over realistic global routes. Ships traverse unique fixed routes composed of multiple ports, known as port rotations, and complete these rotations multiple times per year. The five most centralised global ports, Antwerp, Suez, Singapore, Shanghai, and Panama, were selected for simulation [6]. Each port has two berths, with the larger berth accommodating 5 cranes and the smaller berth accommodating 3.

A total of 22 predefined routes are composed of segments extracted from a master route in Figure 1. Each route spans between 2 and 5 ports. Routes with 2 to 4 ports follow a ping-pong traversal pattern, reversing direction at each endpoint. In contrast, the 5-port route forms a continuous loop, capturing full intercontinental coverage and reflecting realistic global shipping dynamics.

The environment includes disruptions from seasonal storm zones, with storms occurring based on deterministic seasonal calendars derived from historical cyclone patterns in the Western North Pacific (WNP), Eastern North Pacific

(ENP), and North Atlantic (NA) regions [7]. These zones, along with their geographic coverage, are illustrated in **Figure 1**. Travel segments are decomposed into storm-affected and unaffected distances by aligning each ship's progress against storm-active spans, enabling context-aware penalty computation.

Figure 1. *Global shipping routes (red lines) connecting Antwerp, Suez, Singapore, Shanghai, and Panama. Shaded areas indicate seasonal storm zones in the Western North Pacific WNP, ENP, and NA regions.*

Each ship must progressively reduce its RCT across successive port rotations while balancing congestion and weather conditions. A 1 000-row shipment matrix introduces realistic loading and unloading dynamics based on the ship's maximum capacity, which varies according to its size.

PROBLEM AND TASK

The problem is formalised as follows: let S_t represent the simulator state at time t, including queue lengths, storm windows, rotation progress, and historical ship performance. At each control point, the agent selects an action, either a continuous delay or a discrete queued ship choice to optimize cumulative reward:

$$R_t = -\alpha \cdot Q + -\beta \cdot S + \gamma \cdot \Delta_{seg} + \Delta_{rot}$$

Where:

- Q is the queue length penalty,
- S is the storm departure penalty
- Δ_{seg} is a per-leg speed improvement compared to past performance,
- Δ_{rot} is the net improvement (or regression) in full-rotation duration.

The scalar weights α,β, and γ were tuned via Bayesian optimisation.

The pseudocode below outlines how the environment applies the agent's actions and incrementally accumulates rewards.

Pseudocode – Environment Step Function

```
function step(action):
  time, event, ship = next event from queue
  reward = 0
  if event == 'departure':
    delay = action['delay']
    if ship completes rotation: reset rotation timers
    advance to next port
    if completed rotation: reward += Δ_rot
    if storm detected: reward -= β * 1
    schedule arrival at time + delay + travel
  elif event == 'arrival':
    reward += γ * Δ_seg
    if berth available:
      assign ship
      schedule service_complete
    else:
      reward -= α * queue length
      add ship to queue
  elif event == 'service_complete':
    ship_choice = action['ship_choice']
    free berth and assign to next ship if queued
    schedule immediate departure
  return new_state, reward, done
```

IV. METHODOLOGY

A. *Agent Architecture*

In this work, A2C was selected for its ability to handle hybrid action spaces. The Actor outputs a continuous delay (Normal) and discrete ship choice (Categorical), while the Critic estimates state value using departure and mean-pooled features of all queued ships. Both share a two-layer MLP with ReLU activations. The Actor outputs the parameters for the delay and queue selection distributions, while the Critic outputs a scalar value representing the state. This design supports effective scheduling in mixed decision environments like maritime logistics.

B. *Training Protocol*

Training was carried out using a policy gradient method, with updates applied at each simulation step. To ensure consistency, seed control was applied by fixing the random seeds across Python, NumPy, and PyTorch. Deterministic algorithms were used in PyTorch to avoid non-deterministic behavior. The loss function included entropy regularization (0.01) to encourage exploration and gradient clipping (0.5) to stabilize training. Hyperparameter tuning was performed using Bayesian optimization to find the best values for α, β, γ, which controlled the queue penalty, storm penalty, and per-leg speed reward, respectively. The optimization aimed to maximize mean normalized improvement in rotation times, with 30 function evaluations and 5 random starts using the Expected Improvement (EI) acquisition function. The optimal hyperparameters found were α = 0.0152, β = 1.9216, γ =0.6940.

The agent then interacted with a fresh simulation environment, without episodic resets. Actions were selected based on the current state, and rewards were issued according to the four criteria previously described. Policy gradients were computed after each step or whenever a non-zero reward was received. Normalised advantages guided both actor and critic updates, with entropy regularisation used to maintain exploration throughout training.

C. *Baselines*

A heuristic baseline, using zero-delay and FCFS berth selection, is evaluated over the same durations as the trained agent, serving as a naive scheduling strategy. The agent's performance is then assessed by comparing its post-training rotation-time improvements to those of the baseline, aiming to demonstrate that it outperforms basic methods by refining its scheduling choices.

D. Cumulative Reward

A one-year simulation was executed to verify whether the reward system was functioning as intended. Cumulative reward was tracked at each environment step that returned a non-zero reward, producing a curve indexed by reward-triggering events.

IV. EXPERIMENTS AND RESULTS

This data was recorded in the agent and saved after training. Interestingly, the cumulative reward trend was increasingly negative over time. To make the learning progression intuitively interpretable, the sign was flipped prior to visualisation.

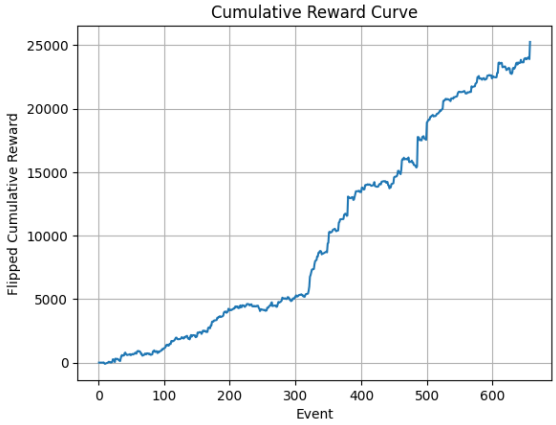

Figure 2. *Flipped cumulative reward during one-year training.*

E. Multi-Year Horizons

Ten independent experiments were conducted, each simulating an environment spanning 1 to 10 years. For each duration, two runs were executed: one with the baseline policy and one with the trained A2C agent. These runs were used to compare improvement metrics, calculated for each ship as the difference between its first completed rotation and its best rotation achieved by the end of the simulation. The mean improvement percentage across all ships in a single run was then calculated as:

$$\frac{1}{N}\sum_{i=1}^{N}\left(\frac{Improvement_i}{First\ Rotation_i}\times 100\right)$$

Where N is the total number of ships.

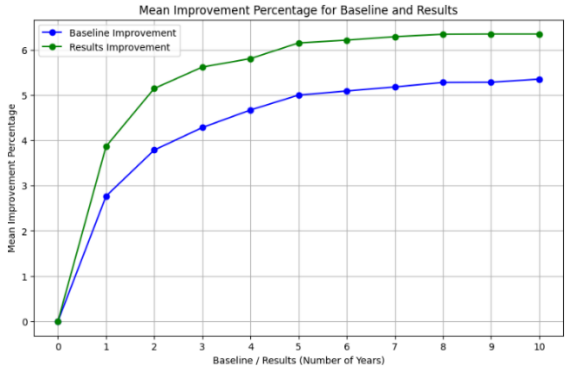

Figure 3. *Trends in mean improvement percentage from year 1 to year 10 for the baseline and the trained agent.*

F. Improvement Ratio Comparison

We selected the five-year simulation results for comparison, as the previous experiment showed marginal gains beyond that point. For each ship, the improvement percentage was extracted from both result sets and their differences calculated. The final table was sorted by this difference to highlight ships that benefited most from training.

Ship ID	Baseline %	Agent %	Difference
1	2.9	9.4	6.4
2	3.8	10.0	6.2
3	3.3	7.9	4.6
6	0.7	5.1	4.4
13	2.9	5.9	3.1
8	6.8	9.3	2.5
19	9.9	11.5	1.7
4	0.4	1.5	1.0
17	4.8	5.6	0.9
5	3.4	4.1	0.6
16	3.3	3.7	0.4
11	1.0	1.5	0.4
14	7.2	7.2	0.0
9	1.9	1.9	0.0
10	8.0	8.0	0.0
18	6.7	6.7	0.0
12	2.2	2.1	-0.0
20	11.3	10.3	-0.9
15	6.9	5.8	-1.0
22	7.4	6.1	-1.3
21	9.1	7.2	-1.9

Table 1. *Per-ship improvement percentages for baseline and agent, and their differences (sorted by agent gain).*

Additionally, we compared the average improvement per across all ships as a percentage difference.

```
Mean baseline improvement: 5.00%
Mean results   improvement: 6.15%
Percentage change: +23.06%
```

G. Runtime Metrics Comparison

The same five-year simulation results were used to evaluate the operational efficiency of the learned policy. Total hours spent in queues, service, and storm conditions were summed across all ships to capture fleet-wide operational time under each condition.

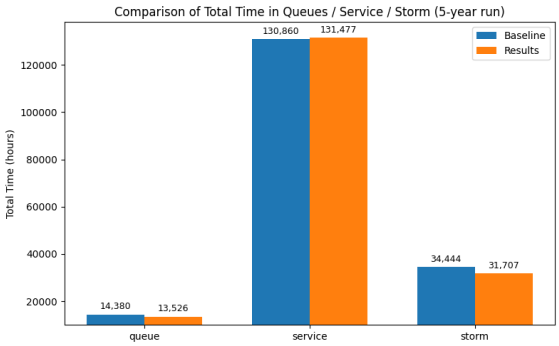

Figure 4. *Total time spent in queues, service, and storm conditions under baseline and trained policy over a five-year run.*

V. DISCUSSION

The results show that the A2C agent consistently outperforms the baseline across evaluation metrics and time horizons. Figure 2 confirms steady learning through an upward cumulative reward trajectory, reflecting effective

penalty reduction across queueing, storm exposure, and inefficiencies.

Figure 3 shows higher mean rotation-time improvements for the agent at every horizon, with the largest gains in the first two years and diminishing returns by year five. Table 1 further quantifies this, with the agent achieving a 6.15% improvement versus 5.00% for the baseline, representing a 23.06% relative gain. This highlights the agent's ability to generalise delay and ship prioritisation strategies. Not all ships improved equally, which points to variation in route structure and storm exposure.

Figure 4 presents operational breakdowns over five years. The agent reduced queue time by 854 hours and storm exposure by 2,737 hours, indicating better timing and weather avoidance. A minor increase of 617 hours in service time indicates improved berth utilisation.

Ship ID	Improvement %	Queue Time	Service Time	Storm Time	Rotations
1	9	609	5944	0	100
2	10	575	5687	0	95
3	8	783	9246	2650	155
4	1	279	2297	4513	40
5	4	291	3128	1732	48
6	5	337	4691	0	71
7	4	632	6021	0	48
8	9	790	7299	1223	58
9	2	247	3244	3610	27
10	8	142	4534	2504	36
11	1	297	3174	3159	25
12	2	216	3309	1485	29
13	6	421	5275	0	41
14	7	1035	10500	815	33
15	6	1106	7003	1849	24
16	4	887	5915	903	19
17	6	765	5896	3610	18
18	7	504	6865	1106	22
19	12	663	6894	990	20
20	10	969	8849	0	26
21	7	894	7825	903	30
22	6	1083	7882	655	30

Table 2. *Five-Year Training Horizon Performance.*

Table 2 highlights ship-level diversity: ships like 1, 2, and 6 faced no storms, while others such as 4, 10, and 17 endured thousands of storm-hours. Queue and service times varied significantly, as did the number of rotations completed, reflecting both the complexity of the environment and the agent's adaptability across routes and conditions.

Overall, these findings validate reinforcement learning as a viable approach to maritime scheduling. The hybrid action space and structured reward signal enabled meaningful efficiency gains under realistic constraints. Future work could incorporate sustainability metrics such as fuel and emissions, and extend the simulation to include terminal-side operations involving dockside machinery operations for end-to-end optimisation.

VI. CONCLUSION

This work presents a reinforcement learning framework for real-time maritime scheduling using a self-comparative reward strategy. The A2C agent, trained in a storm-aware, event-driven simulator, achieves a 23.06% relative reduction in rotation time compared to the FCFS baseline. By optimising both ship delay and prioritisation, the agent reduces queueing and storm exposure while increasing berth efficiency. The study shows that most efficiency gains occur within the first five years of training, after which further improvements diminish.

REFERENCES

[1] M. Golias, M. Boile, and S. Theofanis, "The berth-scheduling problem: Maximizing berth productivity and vessel service level," *Transportation Research Record*, vol. 2100, pp. 35–41, 2009. [Online].

[2] J. K. Jim and F. Q. Alnizam, "AI-Ship-Scheduler: A reinforcement learning-based ship scheduling simulator," GitHub repository, 2025. [Online]. Available: https://github.com/Jeremoot/AI-Ship-Scheduler

[3] B. Moradi, R. Tavakkoli-Moghaddam, and M. Khalilzadeh, "Reinforcement learning in maritime routing under uncertain weather," *Ocean Engineering*, vol. 257, p. 111530, 2022. [Online]. Available: https://doi.org/10.1016/j.oceaneng.2022.111530

[4] J. Notteboom, "The relationship between seaports and the intermodal hinterland in light of global supply chains," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 365, no. 1551, pp. 477–490, 2010. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2880080/

[5] *Tropical cyclone: Location and patterns of tropical cyclones*, *Encyclopedia Britannica*. [Online]. Available: https://www.britannica.com/science/tropical-cyclone/Location-and-patterns-of-tropical-cyclones

[6] T. Wang, H. Zhao, and Q. Liu, "Deep Q-learning for berth allocation optimisation," *Journal of Marine Science and Technology*, vol. 29, no. 4, pp. 301–312, 2024.K. Elissa, unpublished.

[7] Y. Zhang and H. Xu, "Advantage actor-critic for traffic signal control in port logistics corridors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 1, pp. 142–153, 2024. [Online]. Available: https://doi.org/10.1109/TITS.2023.3321456

[8] J. Zhang, C. Zhang, and Y. Zhang, "Reinforcement learning for logistics scheduling: A comprehensive review," *Expert Systems with Applications*, vol. 236, p. 120205, 2024. [Online]. Available: https://doi.org/10.1016/j.eswa.2023.120205

[9] S. Zhen, Y. Li, and K. Yang, "Deep reinforcement learning for canal transit scheduling," *Transportation Research Part E: Logistics and Transportation Review*, vol. 156, p. 102525, 2021. [Online]. Available: https://doi.org/10.1016/j.tre.2021.102525J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–7