Deep Reinforcement Learning for Maritime Route Optimisation: A Hexagonal Grid Approach

From heuristic baselines to a DDQN agent on an isotropic H3 grid over the Mediterranean Sea

Author: Bilal Saleem | Supervisor: Prof. Matthew Piggott & Dr. Samaneh Mofrad | Imperial College London

IMPERIAL COLLEGE LONDON
MSC APPLIED COMPUTATIONAL SCIENCE AND ENGINEERING

RLSS 2026 · MILAN

ELLIS Unit Milan · Reinforcement Learning Summer School 2026

1 THE CHALLENGE: DECARBONISING MARITIME SHIPPING

International shipping accounts for approximately **3% of global greenhouse gas emissions**. The IMO's 2023 revised strategy targets net-zero emissions by 2050, with an interim 20–30% reduction by 2030 relative to 2008 levels. Voyage optimisation — choosing the safest, most fuel-efficient route through dynamic, weather-varying ocean conditions — is one of the most tractable near-term levers available to fleet operators today.

2 GAP IN CURRENT APPROACHES

Classical methods (A*, Dijkstra, Tabu Search) are fast and interpretable, but **static**: they plan once and cannot adapt to evolving weather mid-voyage. Reinforcement learning agents, by contrast, learn a policy that can be continuously updated online as new forecasts arrive.

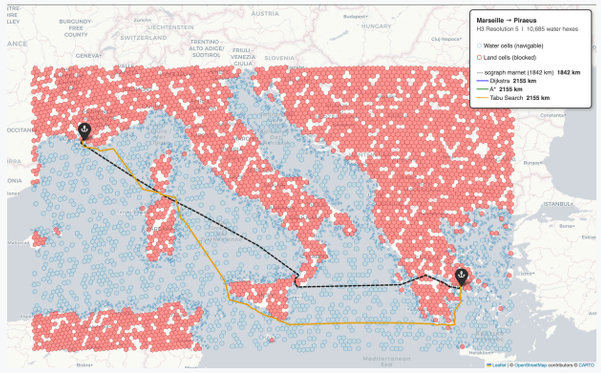
Latinopoulos et al. (JMSE, 2025) is the state of the art: DDQN and DDPG agents on a uniform 0.25° lat/lon grid, achieving 8.7–12.6% fuel savings over a shortest-path baseline on a real Mediterranean route. It is a strong result — and it was this paper that motivated this project. But it has two structural limitations:

- → The uniform **28 km grid** cannot navigate straits narrower than one cell. Canals are handled with teleportation shortcuts, not actual navigation.
- → The **rectangular grid introduces diagonal bias** (NE step = 41% longer than N step), forcing the Q-network to learn a geometry correction.

Research Question: *Can a DDQN agent on an adaptive H3 hexagonal grid — with isotropic movement and genuine strait/canal navigation — match or exceed the Latinopoulos fuel savings, validated across three Mediterranean routes of increasing difficulty?*

3 STEP 1 — HEURISTIC BASELINES

Before training any neural network, a full suite of heuristic pathfinding algorithms was implemented on the same routes, mirroring the baseline comparison in Latinopoulos et al.



H3 res-5 grid — 10,685 water/coastal hexes · Marseille → Piraeus · scgraph 1,842 km vs H3 Dijkstra 2,155 km

5 STEP 2 — RETHINKING THE GRID: RECTANGLES → ADAPTIVE HEXAGONS

LATINOPOULOS 0.25° GRID

Uniform 28 km per cell everywhere. Aligns with ERA5 data. But: diagonal step = √2 × cardinal step = 1.41× longer. 8 actions with *two distinct step lengths*. Q-values encode a geometric artefact. Straits <28 km inaccessible. Corinth Canal: teleportation shortcut.

→

ADAPTIVE H3 HEXAGONAL GRID

All 6 neighbours equidistant — always. 6 actions, all equal cost. Q(s,"move NE") = Q(s,"move N") under same progress. Reward shaping geometrically unambiguous. Coastal resolution 3.7 km. Straits and canals navigable.

RESOLUTION ZONES

ZONE	RESOLUTION	CELL SIZE	PURPOSE
Open ocean	Res-5	~9.85 km edge / ~28 km c-to-c	Matches Latinopoulos. Efficient BFS.
Within 40 km of coast	Res-6	~3.7 km edge / ~10 km c-to-c	Straits, canal approaches, harbours become navigable corridors

The adaptive boundary is computed automatically: any res-5 water cell adjacent to a land cell is replaced by its 7 res-6 children, each independently classified by polygon land-fraction (threshold ≥ 0.20 → excluded). Polygon-level check, not centroid. **Result: 10,937 water/coastal cells** across the Mediterranean.

6 MDP FORMULATION & STATE SPACE

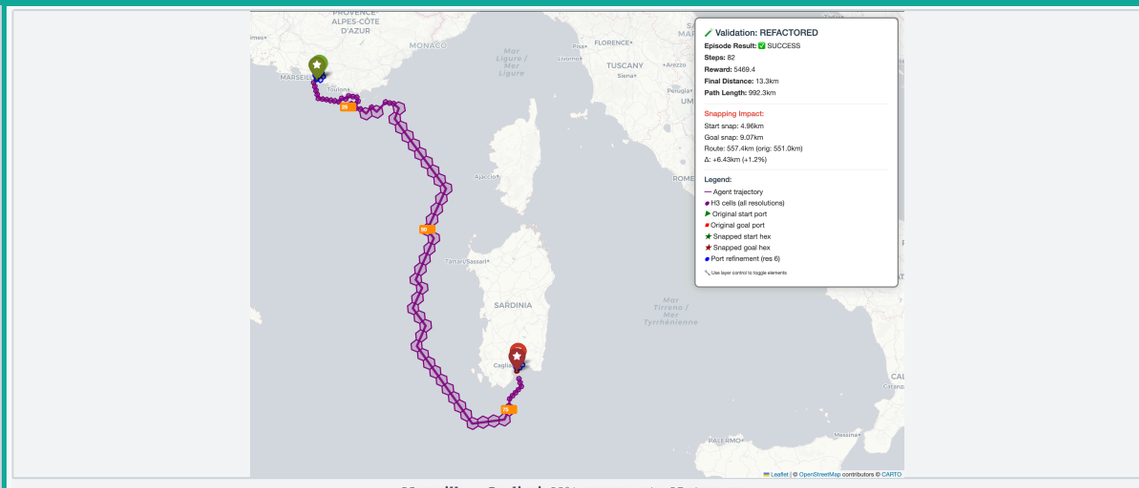
MDP: ⟨S, A, P, R, Γ⟩

```
S : ℝ⁸ observation vector (below)      γ = 0.99
A : 7 discrete actions                 P : deterministic graph transitions
R : potential-based shaped reward      Episodic, finite-horizon
```

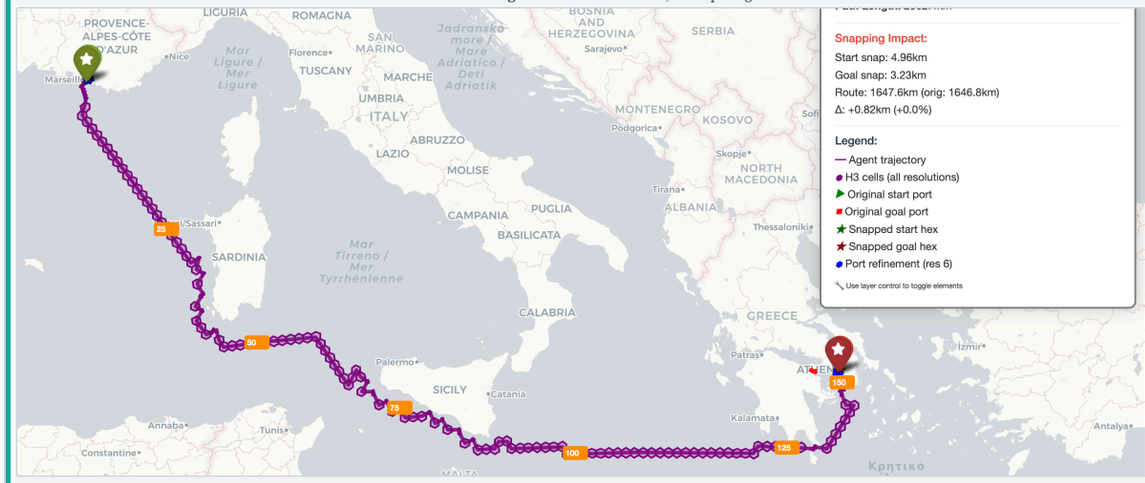
STATE VECTOR S ∈ ℝ⁸ (ALL NORMALISED)

DIM	FEATURE	RANGE
$\hat{\phi}$	Normalised latitude	[0, 1]
$\hat{\lambda}$	Normalised longitude	[0, 1]
$\hat{d}_{BFS}$	BFS hops to goal / initial hops	[0, 1]
$\hat{d}_{hav}$	Haversine dist / initial dist	[0, 1]
t_{frac}	Step count / max steps	[0, 1]
$\hat{\psi}$	Last action bearing / 360°	[0, 1]
$\Delta\hat{d}$	Progress this step (normalised)	[−1, 1]
$\hat{v}$	Visit count at current hex (norm)	[0, 1]

Action space: 7 discrete actions — 6 H3 ring-1 neighbours sorted clockwise from North + STAY. Binary action mask $m \in \{0,1\}^7$ prevents stepping onto land; computed once and cached per hex.



Marseille → Cagliari. 99% success rate, 85 steps avg.



Marseille → Piraeus. 100% success rate, 154 steps avg.

7 STEP 3 — REWARD ENGINEERING

REFACTORED REWARD — POTENTIAL-BASED SHAPING (NG ET AL., 1999)

```
r(s,a,s') = α·[Φ(s') − Φ(s)]  +  r_terminal  +  Σ r_guardrails

Φ(s) = −BFS_dist(s, goal) / initial_BFS_dist     α = 25.0
r_terminal = +10.0  iff  BFS_dist(s', goal) ≤ 2 hops (sparse)

Guardrails (λ = min(0.4, BFS_hops/100)):
  time: −0.05·λ   land: −1.0·λ   ping-pong: −0.1
  3-cycle: −0.2   revisit: −0.05·count (cap −1.0)
```

8 DOUBLE DQN ARCHITECTURE

```
Target:  y_t = r_t + γ · Q_θ⁻(s_{t+1}, argmax_{a'}: m_{a'}=1) Q_θ(s_{t+1},a'))
Loss:    ℒ(θ) = E_𝒟 [SmoothL1(y_t − Q_θ(s_t, a_t))]
δ=1.0

Network: ℝ⁸ → [Linear(8→400) + LayerNorm + ReLU]
             → [Linear(400→300) + LayerNorm + ReLU]
             → Linear(300→7)
LayerNorm (not BatchNorm): no dependency on batch statistics.
```

HYPERPARAMETER	VALUE
Optimizer	Adam, lr = 5×10⁻⁴
Replay buffer	1,000,000 transitions (numpy ring buffer)
Mini-batch	128 transitions, sampled uniformly
Gradient clip	$\|\nabla\theta\|_2 \leq 1.0$
Exploration	ε-greedy, ε: 1.0 → 0.01 (×0.995/ep)
Soft update	τ = 0.001, every 5 gradient steps
Discount γ	0.99

9 LONG-ROUTE EXPLORATION — PIRAEUS (1,647 KM)

Cold-start problem: random-walk exploration covers ≈√500 ≈ 22 hops in a 500-step episode; the goal is 107 BFS hops away — effectively invisible without structure.

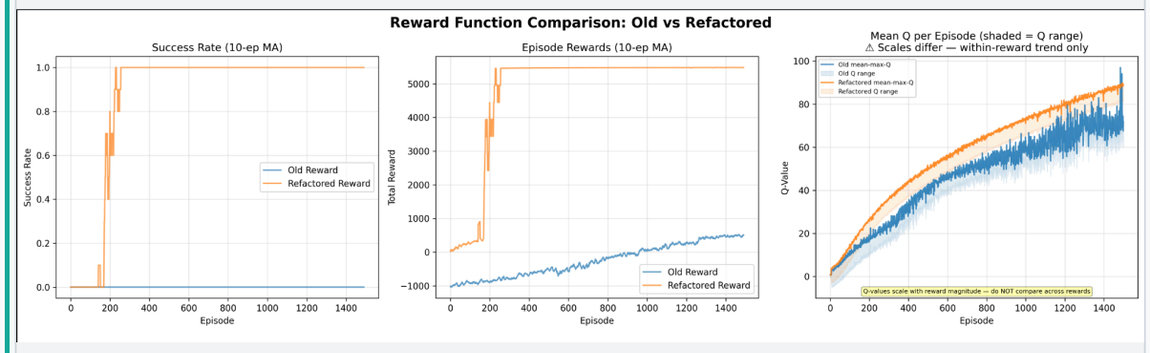
BFS-SEEDED REPLAY BUFFER

The replay buffer is pre-populated with BFS-optimal trajectories before training begins. The agent sees a complete successful route in its first batch — bootstrapping Q-values toward positive territory before ε-greedy exploration even starts. Without seeding, Q-values for all actions converge to a uniform ~9.5–10 (no gradient signal), and the agent never leaves its starting region.

Why BFS, not random seeding? *A random trajectory rarely reaches the goal on a 107-hop route. BFS guarantees the agent sees r_terminal = +10.0 in its very first replay batch — the only way to break the cold-start deadlock on long routes.*

10 CONCLUSIONS

- → **Uniform grids introduce a geometric artefact** — diagonal steps are 41% longer, forcing Q-values to correct for grid geometry rather than encode navigational value. H3 eliminates this: all 6 moves are equidistant, so Q(s, a) reflects route merit alone.
- → **Adaptive resolution is a prerequisite for correctness, not an optimisation** — res-6 refinement makes the Strait of Bonifacio traversable; an explicit res-6 corridor makes the Corinth Canal navigable end-to-end. No teleportation. No special-case reward logic.
- → **Reward hacking is diagnosable and structurally fixable** — on Piraeus, episode reward climbed from −773 to +165 over 1,500 episodes while success stayed at 0%: the agent was farming a non-Markovian signal. Potential-based shaping (Ng et al., 1999) removes the incentive by aligning Φ(s) = −BFS_dist/initial_dist with the actual distance-to-goal.
- → **All three routes solved** — Port Torres 100%, Cagliari 99%, Piraeus 100%.



Reward refactoring. Refactored policy: 100% success; old policy: 49.7%.

REFERENCES

[1] Latinopoulos, C. et al. (2025). Deep Reinforcement Learning for Ship Routing Optimisation in Dynamic Maritime Environments. *J. Mar. Sci. Eng.* 13, 902. — Primary benchmark: DDQN+DDPG on orthogonal grid, 8.7–12.6% fuel savings.

[2] Van Hasselt, H., Guez, A., Silver, D. (2016). Deep Reinforcement Learning with Double Q-Learning. *AAAI*, 30, 2094–2100. — Double DQN eliminates Q-value overestimation bias.

[3] Ng, A.Y., Harada, D., Russell, S. (1999). Policy Invariance Under Reward Transformations: Theory and Application to Reward Shaping. *ICML*. — Theoretical basis for potential-based reward shaping.

[4] Uber Technologies, Inc. (2018). H3 — Hexagonal Hierarchical Geospatial Indexing System. San Francisco, CA. — The hexagonal grid system used throughout this work.

[5] Du, Y., Chen, Y., Li, X., Schönborn, A., Sun, Z. (2022). Data fusion and machine learning for ship fuel efficiency modeling. *Commun. Transp. Res.* 2. — Motivates the proposed neural FOC model.

[6] IMO (2023). MEPC.80(WP.12). 2023 IMO Strategy on Reduction of GHG Emissions from Ships. International Maritime Organization, London. — Regulatory context and 2050 net-zero target.