JAXNAVISIM: A GPU-ACCELERATED, DATA-DRIVEN SIMULATOR AND LEARNING BENCHMARK FOR MARITIME NAVIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Coordinating vessels through the shared waterways of busy ports grows more challenging as vessel traffic increases. Port authorities need realistic simulations to assess how arrival schedules, anchorage assignments, and routes affect vessel throughput and navigation safety. We introduce JAXNAVISIM, a data-driven simulator and learning benchmark coupling Automatic Identification System (AIS) data with fast, controllable traffic simulation. *First*, we release a dataset of $\sim$141 million AIS records from over 30K vessels across eight high-density regions of the Singapore Strait, among the world's busiest waterways, with $\sim$6.7 million vessel-hours. Unlike several existing AIS datasets which mostly cover open seas, ours captures close-quarters vessel traffic in the narrow strait. *Second*, we formulate navigation as a goal-reaching task, and implement a scalable JAX simulator supporting historical replay, selective vessel control with batched GPU execution. We also incorporate multiple vessel dynamics models and sea surface currents. *Finally*, we benchmark imitation learning (IL), offline and online RL, with ablations of key design choices. Learned policies reach high goal completion, but collisions persist. We therefore propose Predictive Action Refinement (PAR), which refines actions using short-horizon interaction forecasts, substantially reducing collisions. Our contributions enable research on scarce maritime resource allocation, traffic modeling, digital twins, and provide large real-world testbed for IL/RL methods.

1 INTRODUCTION

Maritime traffic management in busy ports requires coordinating vessel movements through constrained, shared waters (Corry & Bierwirth, 2019; Jia et al., 2020). In a traffic separation scheme (TSS), vessels enter, follow, cross, and leave traffic streams; at anchorages, arriving vessels approach assigned anchor positions among anchored ships and passing traffic. Operational decisions such as arrival timing, anchorage assignment, and approach route shape both a vessel's path and the encounters it faces. This is particularly relevant in busy ports such as Singapore's where vessel movements must be coordinated within limited sea space (Agussurja et al., 2018).

Historical data record only one outcome of a day's traffic decisions. Port planners, however, are often interested in counterfactual questions: *how would vessel interactions, traffic throughput, navigation safety change if an arrival occurred later, a vessel were assigned to a different destination, or a different approach route were selected?* Answering them requires behavioral models that generate vessel movements under conditions not recorded historically. Automatic Identification System (AIS) data offer a large record from which realistic vessel behavior can be learned. Prior works use these data mainly for trajectory prediction (Li et al., 2023; Nguyen & Fablet, 2024), which forecasts movement under the decisions that were actually taken, whereas port planners must evaluate several alternative decisions. We therefore separate decisions from behavior: experts or traffic-management systems specify feasible operational decisions, and the simulator evaluates the behavioral consequences of these decisions rather than inferring them from AIS alone.

Prior maritime works evaluate learned models mostly in open loop: a trajectory predicted from the logged history is compared with the recorded one using displacement errors (Li et al., 2023; Nguyen & Fablet, 2024). A navigation policy, however, acts sequentially. In closed loop, each action changes

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

the vessel's subsequent position and observations, and small errors can move the policy away from the demonstrated states and compound over time (Ross et al., 2011). Autonomous-driving research has shown that open-loop scores are poorly aligned with closed-loop performance (Dauner et al., 2023), motivating closed-loop benchmarks such as nuPlan and Waymax (Karnchanachari et al., 2024; Gulino et al., 2023). The same holds at sea: low displacement error does not prevent collisions or missed destinations, and avoiding encounters by large detours fails the task. We therefore evaluate policies in closed loop on goal completion, arrival timing, trajectory fidelity, kinematics, and collisions, against both logged and reactive traffic.

Counterfactual analysis and closed-loop evaluation both require a benchmark combining curated AIS demonstrations, scalable simulation, flexible vessel control, and common behavioral metrics. Existing public resources provide these only in isolation. Large public AIS archives (Danish Maritime Authority, 2026; MarineCadastre.gov, 2026) release raw point records from which users must construct trajectories, and task splits (Ma et al., 2026). Processed resources target trajectory forecasting (Nguyen & Fablet, 2024; Ma et al., 2026) rather than closed-loop navigation. Existing maritime simulation and learning platforms (Pham et al., 2025; Vaidheeswaran et al., 2025) do not jointly provide public expert demonstrations, accelerator-native batched execution, and flexible multi-vessel control. Such fragmentation makes it difficult to train policies at scale and to verify that better goal reaching preserves realistic, safe interaction in dense traffic. We thus introduce JAXNAVISIM, a data-driven simulator and benchmark for learning and evaluating maritime navigation across anchorage and TSS settings. Our contributions are:

- **Public AIS data and reproducible navigation tasks.** We release over 141 million AIS records from more than 30K vessels across eight high-density anchorage and TSS regions in the Singapore Strait. We turn these data into goal-reaching navigation tasks with predefined training and test splits for reproducible learning and evaluation.
- **Scalable simulation with configurable vessel dynamics and interactions.** Our functional JAX (Bradbury et al., 2018) simulator supports historical replay, selective vessel control against replayed traffic, and joint self-play through a common interface. Batched GPU simulation reaches 3.1 million single-vessel and 53.6 million multi-vessel steps per second on a consumer-grade RTX 3090, supporting the many rollouts that learning and evaluation require. The simulator accounts for hull dimensions and surrounding traffic, which shape vessel behavior and collision risk (Yang et al., 2024), and offers alternative vessel dynamics models, and sea surface currents induced external effects.
- **A common evaluation protocol and comparative baselines.** We benchmark imitation learning (IL) and offline and online reinforcement learning (RL) on punctuality, trajectory fidelity, kinematics, and dimension-aware collision rates, and ablate key design choices. Learned policies can achieve high goal completion but may incur high collision rates. We therefore propose Predictive Action Refinement (PAR), which refines actions with short-horizon interaction forecasts and substantially reduces collision rates.

Beyond maritime applications, the setting offers learning researchers long episodes, local observations, and dense vessel interactions. With expert demonstrations and controllable simulation, it supports studying, through IL and offline and online RL, how policies behave over many decisions and how goal completion relates to trajectory fidelity and collision avoidance.

2 RELATED WORK

Table 1 compares JAXNAVISIM with other data-driven simulators.

Data-driven simulator for autonomous driving (AD) Early works such as nuPlan (Caesar et al., 2021) combine public driving scenarios and approximately 1,500 hours of human-driving data with closed-loop planning evaluation. Nocturne (Vinitsky et al., 2022) uses real-world driving scenes from the Waymo Open Motion Dataset (Ettinger et al., 2021) to benchmark partially observable multi-agent reinforcement learning and imitation learning. More recent works such as Waymax (Gulino et al., 2023) and GPUDrive (Kazemkhani et al., 2025) focus on GPU-accelerated simulation for scalable policy training and evaluation. Waymax provides a JAX-based simulator that supports batched execution on GPUs, whereas GPUDrive uses a CUDA-based engine for high-throughput multi-agent reinforcement learning. Many other AD simulators exist, owing to large public datasets and the need to benchmark planning and control (Li et al., 2024; Zhang et al., 2024).

	Domain	Real data	Open code/dataset	GPU acceleration	Collision check	Multi-agent	IL	Offline/online RL
nuPlan (2021)	road	✓	✓	✗	box overlap	✗	✓	✗
Nocturne (2022)	road	✓	✓	✗	box overlap	✓	✓	online
Waymax (2023)	road	✓	✓	✓	box overlap	✓	✓	online
GPUDrive (2025)	road	✓	✓	✓	box overlap	✓	✓	online
MariNav (2025)	sea	✓	code	✗	✗	✗	✗	online
ShipNaviSim (2025)	sea	✓	code	✗	proximity threshold	✗	✓	✗
Zhang et al. (2026)	sea	✓	✗	✗	proximity threshold	✗	✗	online
JAXNAVISIM (ours)	sea	✓	✓	✓	capsule overlap	✓	✓	✓

Table 1: JAXNAVISIM v.s. other simulators. "Open code/dataset": code and curated dataset are public ("code": code only). "IL": imitation learning from logged trajectories. "Offline/online RL": "online" denotes online RL only; ✓ denotes both. "Collision check": "overlap" means the vehicle or vessel shapes intersect; "proximity threshold" is a distance-based collision-risk proxy, such as the 555 m radius of ShipNaviSim.

Our focus, vessel navigation, however, differs from driving. Driving scenarios typically last seconds (9 s in Waymax), whereas a vessel's passage through one of our regions lasts tens of minutes to over an hour. A vessel cannot stop or turn quickly, and sea-surface currents affect its motion. These characteristics are not represented in driving simulators.

Data-driven simulator for maritime navigation In contrast to AD, far fewer studies have investigated data-driven simulation for maritime navigation. MariNav (Vaidheeswaran et al., 2025) uses an AIS-derived traffic graph and wind data for single-vessel routing. Zhang et al. (2026) train a single-vessel curriculum RL policy for fuel-aware routing in an Indian Ocean environment built from proprietary AIS data, but no land, traffic lanes, or anchorages, and assess collision risk by proximity criteria. The closest work, ShipNaviSim (Pham et al., 2025), learns from AIS data, but its released code runs only CPU-based single-vessel control on synthetic examples. It also measures proximity with a fixed 555 m near-miss threshold. In contrast, our collision metric detects geometric overlap from vessel dimensions and orientations. This matters most in anchorages, where vessels often operate at close quarters. Overall, no existing maritime platform provides both public AIS data and code, or AD-simulator functionality such as GPU execution and geometry-based collision detection. JAXNAVISIM combines these with flexible single- and multi-vessel control in one benchmark, supporting IL, offline and online RL in anchorage and TSS settings.

3 JAXNAVISIM

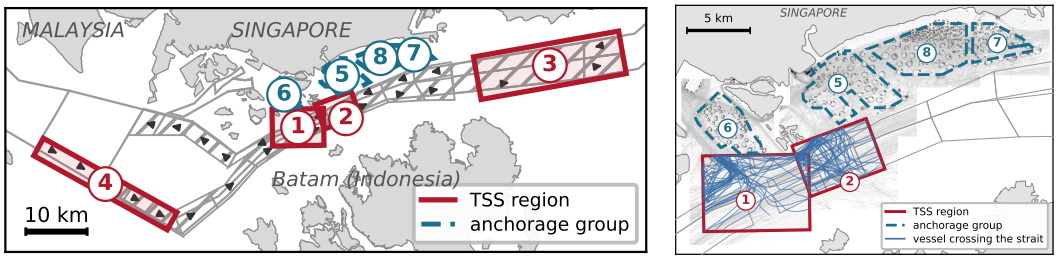
We next describe the problem setting, dataset and task construction, the simulator interface, and the metrics used to evaluate policies.

3.1 MARITIME PROBLEM SETTING

The Singapore Strait is one of the world's busiest waterways. We release eight regions of it that differ in how vessels navigate (Figure 1). The four anchorage groups (regions 5–8) are highly constrained spaces, where pilots maneuver slowly among closely anchored ships to reach an anchoring position. TSS East and TSS West (regions 3 and 4) act as a marine highway: through traffic follows the one-way lanes of the traffic separation scheme (TSS) at cruising speed. At Intersections 1 and 2 (regions 1 and 2), vessels leave or join these lanes to enter/exit port waters among crossing traffic. Together, the regions span slow maneuvering in confined water, lane following, and merging and crossing. Each task asks a vessel to reach a goal among surrounding traffic: in an anchorage region, an anchoring position or, for vessels passing through, the point where they leave the region; in a TSS region, the end of its recorded pass. Although we focus on the Singapore Strait, where our dataset is available, nothing in JAXNAVISIM is specific to it. A new setting needs only AIS dataset, which follows a standard format worldwide, and charted region outlines.

3.2 GOAL-REACHING NAVIGATION TASK

We formulate maritime navigation as a goal-reaching problem. Each recorded vessel passage defines one navigation episode. For episode m, let $\tau^m = \left(z_0^m, z_1^m, \ldots, z_{T_m}^m\right)$, and $z_t^m = (x_t^m, y_t^m, v_t^m, \psi_t^m)$ denote the recorded trajectory, where (x_t^m, y_t^m) is the vessel position, v_t^m is its speed, and ψ_t^m is its heading. The controlled vessel is initialized from z_0^m, and its recorded terminal position $g^m =$



(a) TSS lanes and the eight released regions.

(b) Anchorage density, crossing traffic.

Figure 1: **The Singapore Strait setting.** Regions 1–4: Intersection 1, Intersection 2, TSS East, TSS West. Regions 5–8: anchorage groups AEW, AWW, AEPBC, AESPA, each a cluster of contiguous anchorages, together holding 20 of Singapore's 32 anchorages and 72% of their area. (a) TSS lanes and direction of travel. (b) AIS position density on 3 June 2024 (grey, log scale). Each ring is an anchored ship swinging around its anchor; no other ship may anchor within it, which tightly constrains space. Median vessels at rest over 2024: 111, 66, 42, and 34 in regions 8, 5, 6, and 7. Blue: vessels crossing the strait through the intersections.

$\left(x^m_{T_m}, y^m_{T_m}\right)$ defines its navigation goal. In an anchorage, this is the anchoring position of an arriving vessel or the exit point of a transiting one; in a TSS or intersection region, end of recorded passage.

At each step, the policy observes the goal and recent states of the controlled vessel and nearby traffic, and selects an action (see section 3.4, action model). An episode succeeds if the vessel enters a radius (e.g., 100 m) around its destination within the evaluation window. We also evaluate arrival timing, trajectory fidelity, kinematics, and collisions. Surrounding vessels replay their recorded trajectories, or any subset can instead be controlled by learned policies.

3.3 Dataset Construction

Our released dataset covers the eight high-density regions in Figure 1 and is built from raw 2024 AIS data provided by the Maritime and Port Authority of Singapore (MPA). It contains ∼141 million AIS records from more than 30,000 vessels at around 3 minute intervals.

Static vessel information. Static records give vessel identifiers, length, beam, and AIS vessel type. Dimensions define each vessel's geometry for collision checking, and type helps selects the controlled vessels, such as cargo ships and tankers, whose size makes them hard to maneuver.

Dynamic trajectory information. Vessels broadcast AIS reports at irregular intervals of seconds to minutes. Each report gives a timestamp, lat-long, speed and course over ground, heading, and navigational status (e.g., at anchor). As simulation and learning need a uniform time step, we linearly interpolate each vessel's reports to the simulator's 10 s step, and build region-specific trajectories linked to static vessel information. Each controlled-vessel trajectory comes with the recorded states of surrounding vessels in the same region and time interval; this traffic, to which the recorded vessel responded, is replayed in simulation and used for collision checks.

Episode construction An episode begins when a vessel enters a region and ends at its first observed terminal event. In an anchorage, this event is either an *arrival*, when the vessel comes to a sustained stop inside the region, or a *transit*, when it leaves the region without one; in TSS and intersection regions, it is always a transit. For anchorage arrivals, we keep only vessels that anchored, removing those that stopped alongside another ship to service it (e.g., bunkering). We control large cargo ships and tankers, which are challenging to maneuver, with IL and RL policies, whereas the traffic context also includes vessels of other types. We train on the 273 days of January–September and test on the 92 days of October–December, which evaluates generalization to later traffic within each region (Table 2). Unlike other AIS datasets focused on open-sea passage, our benchmark covers varied close-quarters tasks, such as arrivals and transits among anchored ships, TSS lane following, merging and crossing traffic (Table 2). Appendix B documents the different preprocessing steps.

3.4 Simulator Design

The simulator is a functional, batched JAX environment that replays recorded traffic around controlled vessels (Figure 4 in Appendix C).

	Anchorage groups					Intersection/TSS regions				
	AEW	AESPA	AEPBC	AWW	*Total*	Int. 1	Int. 2	TSS E	TSS W	*Total*
Train episodes	22,258	20,396	11,985	7,404	62,043	107,243	92,763	68,906	26,152	295,064
Train transitions (millions)	1.93	2.58	1.20	0.68	6.39	18.53	11.25	28.20	10.13	68.11
Test episodes	7,600	7,068	4,219	2,454	21,341	34,836	30,778	18,958	6,538	91,110
Test arrivals	22%	76%	54%	53%	—	—	—	—	—	—

Table 2: Test arrivals give the share of test episodes that are arrivals; the rest are transits, and intersection/TSS regions have only transits. Models are trained and evaluated separately in each region.

Scenarios and closed-loop control. A scenario holds logged trajectories, vessel metadata, control masks, and navigation goals. Let s_t be the simulated scene of N vessels, including observation history, and $\mathcal{C} \subseteq \{1, \dots, N\}$ denote the controlled vessels, selected by a per-vessel control mask. Each vessel $i \in \mathcal{C}$, with local observation $o_t^i = O^i(s_t)$ and goal g^i, acts through a policy π_θ, possibly shared across vessels, and the scene advances as:

$$a_t^i \sim \pi_\theta(\cdot \mid o_t^i, g^i) \quad (\forall i \in \mathcal{C}), \qquad s_{t+1} = F\big(s_t, \{a_t^i\}_{i \in \mathcal{C}}; \ell_{t+1}\big), \tag{1}$$

where F advances the scene by one step, applying the actions under the chosen dynamics model (e.g., equation 2) and setting uncontrolled vessels to their recorded states ℓ_{t+1}. With one controlled vessel we drop i. Controlled vessels evolve in closed loop and respond to one another; replayed vessels do not react. A policy sees only local observations and vessel's goal, never future replay samples. The mask selects replay ($\mathcal{C} = \emptyset$), a single ego, mixed control, or self-play ($\mathcal{C} = \{1, \dots, N\}$). We evaluate a single ego ($|\mathcal{C}| = 1$) in the anchorage regions (Section 5.2), where most vessels are stationary, and multiple controlled vessels in the TSS regions (Section 5.3).

Observations. At step t, each controlled vessel observes from its own perspective its current and past H states, the same history of its 10 nearest vessels within 1 nautical mile, its goal, and, when enabled, the local sea current. A state holds position, speed, heading, length, beam, and AIS vessel type (Appendix C.1). Unobserved vessels stay in the scene and are still checked for collisions.

Actions and dynamics model. Following ShipNaviSim (Pham et al., 2025), we use a continuous three-dimensional delta action space (Gulino et al., 2023), $a_t = (\Delta x_t, \Delta y_t, \Delta\psi_t)$, the changes in position $p_t = (x_t, y_t)$ and heading ψ_t over one $\Delta t = 10\,\text{s}$ step:

$$p_{t+1} = p_t + (\Delta x_t, \Delta y_t), \qquad \psi_{t+1} = \text{wrap}(\psi_t + \Delta\psi_t), \tag{2}$$

where $\text{wrap}(\theta) = ((\theta + \pi) \bmod 2\pi) - \pi$ keeps the heading in $[-\pi, \pi)$; next speed is the distance traveled per step, $v_{t+1} = \sqrt{\Delta x_t^2 + \Delta y_t^2}/\Delta t$. A delta action applies to any moving object and needs no vessel-specific dynamics; only loose kinematic bounds (Appendix C.2) constrain it, without enforcing acceleration or other hydrodynamic responses. Other motion models are described later.

Rewards and termination. A shaped reward combines goal progress, a clearance penalty, and goal-arrival, collision, and region-exit terms (Appendix C.3). Episodes end on goal arrival, region exit, or an exhausted step budget (Appendix C.4).

Accelerator-native execution. Trajectories and metadata are structured JAX arrays; reset, step, observation, and reward are pure functions. `lax.scan` runs rollouts and `vmap` batches scenarios; simulation and policy inference can stay on the accelerator. Section 5.1 measures batched stepping with logged actions, a workload distinct from end-to-end policy evaluation (Appendix M).

Evaluation metrics. We evaluate each controlled vessel over its full rollout; Appendix D gives the definitions. **Goal reaching (R_G)** is the fraction of episodes that reach the goal radius within the evaluation window (Appendix C.4); it measures precise, on-time arrival. **Collision rate (C)** is the fraction of episodes in which the ego's hull, modeled from its length and beam, overlaps another vessel's. **Closest point of approach (CPA)** is the minimum clearance between the ego's hull and any other hull over the rollout, in metres; it reveals near misses that C does not count. **Tracking error** is the average (ADE) and final displacement (FDE) error from the recorded track, measured by gc-ADE, following ShipNaviSim, and by $\text{FDE}_{\neg R}$, computed *over missed goals only*. **Kinematics** are per-episode extremes of acceleration, curvature, and course–heading drift (Appendix K); they test whether the motion is plausible for a large ship.

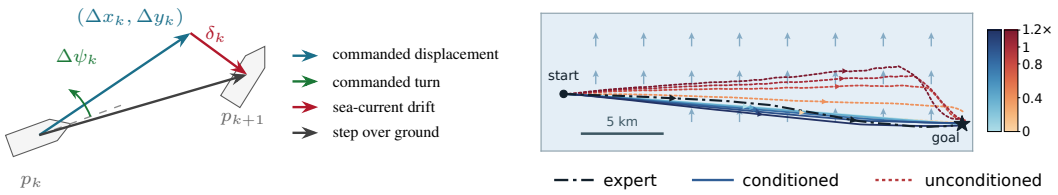


Figure 2: Delta dynamics with the external effect and sea current-conditioned policy behavior.

3.5 Additional Dynamics and Weather Conditions

External conditions. In the delta action space (equation 2), each action is simply the change between two consecutive points in a recorded ship track (Pham et al., 2025). This is convenient, but large vessels have no actuator on the sway axis: their propeller and rudder provide surge and yaw, but no commanded sway (Do, 2010). We next model what does move a ship sideways. Surface current is sampled from the HYCOM reanalysis (Chassignet et al., 2007) at each vessel's own position and time, giving a displacement δ_k per step. Figure 9 in the appendix shows what the sea-current field looks like. The left panel of Figure 2 illustrates the forward dynamics: the commanded displacement (blue) and heading change (green) combine with current drift (red) to produce the vessel's movement over ground (grey). Expert actions are obtained by inverting these dynamics (Appendix N), solving for the command that, together with the external effect δ_k, produces the observed step. Taking that action therefore lands the vessel on the next AIS sample exactly. In the right panel of Figure 2, the blue spectrum is the current-conditioned policy and the red spectrum the unconditioned one, each shown under multipliers from zero to $1.2\times$ the reference sea current field. The conditioned policy stays close to the expert track, while the unconditioned one is pushed farther upward as the current strengthens but still turns back toward the goal. Conditioning on the sea current therefore helps the policy stay close to the expert trajectory.

Additional dynamics models. The simulator provides four models: *Delta* adds drift to Equation 2; *Port-glide* removes the sway command and uses the AIS heading; *Water-track* uses the direction of travel through the water; and *Nomoto* applies a first-order steering lag (Nomoto et al., 1957). Of the four, only Port-glide cannot reproduce a recorded track, because it has no way to produce the sideways part of the motion, so replaying its extracted actions no longer returns the expert to its own track (Table 27). We therefore use Water-track, which derives the heading from the vessel's own motion through the water. We refer the reader to Appendix N for the full derivation of each model and how it changes the demonstrations.

4 Baseline Methods

All IL and RL baselines share a late-fusion encoder architecture (Nayakanti et al., 2023) over the ego history, local neighbors, and goal, used in both the policy and value networks (Figure 6 in Appendix E). For most baselines, we also use action chunking, predicting a sequence of future actions rather than one action per step, following its success in IL (Zhao et al., 2023) and offline RL (Li et al., 2025). Appendix E gives the details of each method.

Trajectory-prediction baselines We evaluate two published, open-source trajectory-prediction models for AIS data: TrAISformer (Nguyen & Fablet, 2024), a generative transformer over discretized AIS attributes, and GLRP (Yang et al., 2025), which models ship interactions with a graph. Each model forecasts the vessel's future track once at the episode's first step, and the track is executed open-loop without re-planning.

Imitation learning. Flow-matching behavior cloning (FM-BC) trains the chunk policy with conditional flow matching (Lipman et al., 2023), as in recent flow-matching policies (Black et al., 2024). To act, it turns Gaussian noise into an action chunk with a few Euler integration steps along the learned flow. Normalizing-flow (NF) behavior cloning (NF-BC) (Ghugare & Eysenbach, 2025) replaces the flow-matching head with a conditional normalizing flow, RealNVP (Dinh et al., 2017).

Offline RL. QC-FQL (Li et al., 2025) learns an action-chunking critic and a one-step actor from the offline data; the actor maximizes the critic's value while being distilled from our frozen FM-BC policy. To further reduce the cost of training an additional distilled policy as well as sampling

from the flow-matching policy during training of QC-FQL, we adopt the DEAS (Kim et al., 2026) critic training objective and the best-of-N (BoN) sampling strategy (Stiennon et al., 2020), where we sample multiple outputs from FM-BC policy and select the action sequence with the highest Q-value. We refer to this method as DEAS-BoN.

Online RL. ReinFlow (Zhang et al., 2025) fine-tunes FM-BC in the simulator with PPO (Schulman et al., 2017) updates and a flow-matching regularizer. HR-PPO (Cornelisse & Vinitsky, 2024) trains a per-step actor with a KL penalty that keeps it close to FM-BC policy, and PPO trains the same per-step actor without that penalty.

Multi-agent RL. To test behavior in the multi-agent setting of Section 5.3, we deploy an independent copy of the trained policy on each controlled vessel. All copies share the same learned weights but act from their own local observations and goals, so their actions affect a shared scene and the vessels respond to one another in closed loop.

Predictive Action Refinement (PAR) Learned policies reach their goals at high rates but also collide frequently, and neither offline nor online policy improvement lowers their collision rates (Section 5.2). We therefore propose Predictive Action Refinement (PAR), a test-time layer that makes a frozen policy's actions collision-aware without retraining. Inspired by proposal scoring in Dauner et al. (2023) for autonomous driving and model predictive safety filtering (Wabersich & Zeilinger, 2021), PAR forecasts observed neighbours at constant velocity over a short future horizon at each step, checks the policy's proposed action sequence against these forecasts, and, if needed, replaces it with an alternative from a small set of slowdowns, turns, and goal-directed sequences. A sequence counts as feasible if the ego's predicted hull never overlaps a forecast neighbour's hull over the horizon. PAR keeps the policy's sequence when it is feasible, unless it makes little progress near the goal; otherwise, it executes the feasible alternative that makes the most progress toward the goal or, if none is feasible, the one with the largest predicted clearance. Appendix F gives the formal definition and further details.

5 Experiments

We first measure simulation speed (Section 5.1), then evaluate navigation policies in the single-agent anchorage setting (Section 5.2) and the multi-agent TSS setting (Section 5.3). We also discuss dynamics-model comparisons in Section 5.3, with details reported in Appendix N.9.

5.1 Simulation speed

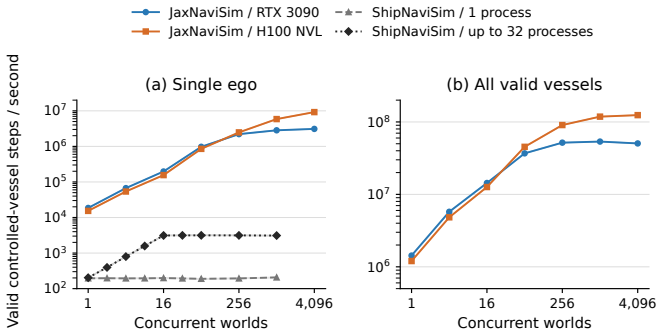


Figure 3: Comparison of simulation throughput with recorded actions: median and minimum–maximum over three repeats at each number of concurrent worlds.

We measure throughput by stepping all 29,858 AEW training and test scenarios with their recorded actions over up to 4,096 concurrent worlds. Following GPUDrive (Kazemkhani et al., 2025), we count valid controlled-vessel steps per second, excluding padded and inactive slots (Appendix M). With one controlled ego per world, JaxNaviSim reaches 3.1 million steps per second on a consumer-grade RTX 3090 and 9.2 million on an H100 NVL, both at 4,096 worlds (Figure 3a); controlling every valid vessel raises this to 53.6 and 124.2 million, respectively (Figure 3b). GPU acceleration is central to this scale. The public CPU implementation of ShipNaviSim reaches 207 ego steps per second in one process and 3,153 with 32 processes, plateauing beyond 16 processes

(Figure 3a, Table 26). A single RTX 3090 thus delivers about 985 times the throughput of 32 CPU processes, and an H100 NVL about 2,900 times.

5.2 SINGLE-AGENT SETTING IN ANCHORAGE REGIONS

Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA↑ (m)	gc-ADE ↓ (m)	FDE$_{\neg R}$ ↓ (m)
NF-BC	0.7545 ± 0.0003	0.2231 ± 0.0028	80.6	26.54 ± 0.13	490
FM-BC	0.8944 ± 0.0057	0.2036 ± 0.0036	87.8	23.18 ± 0.08	436
FM-BC + PAR	0.8955 ± 0.0027	0.0112 ± 0.0007	90.3	23.07 ± 0.13	470
ReinFlow	0.9246 ± 0.0059	0.2132 ± 0.0095	84.3	25.48 ± 0.62	439
HR-PPO	0.8796 ± 0.0094	0.2200 ± 0.0080	80.0	30.25 ± 0.86	453
PPO	0.3382 ± 0.4746	0.4329 ± 0.1714	21.1	242.01 ± 56.28	3640
QC-FQL	0.9199 ± 0.0013	0.2025 ± 0.0019	87.3	23.57 ± 0.15	455
DEAS-BoN, $N = 16$	0.9665 ± 0.0015	0.2054 ± 0.0020	87.2	31.77 ± 0.09	375
GLRP (one-shot)	0.5315 ± 0.0048	0.2439 ± 0.0024	70.8	16.84 ± 0.26	303
TrAISformer (one-shot)	0.2261 ± 0.0125	0.5075 ± 0.0069	-1.5	38.54 ± 1.00	867

Table 3: All baseline families on AEW (7,600 test episodes) under 3 seeds.

Region	Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA↑ (m)	gc-ADE ↓ (m)	FDE$_{\neg R}$ ↓ (m)
AESPA	NF-BC	0.7961 ± 0.0036	0.4897 ± 0.0041	1.8	29.52 ± 0.02	501
AESPA	FM-BC	0.9313 ± 0.0004	0.4510 ± 0.0016	8.5	26.76 ± 0.07	533
AESPA	FM-BC + PAR	0.9259 ± 0.0043	0.0336 ± 0.0025	31.6	27.05 ± 0.01	571
AESPA	ReinFlow	0.9482 ± 0.0201	0.4638 ± 0.0228	6.5	30.00 ± 0.34	540
AESPA	QC-FQL	0.9616 ± 0.0014	0.4461 ± 0.0020	9.4	27.85 ± 0.09	684
AESPA	GLRP (one-shot)	0.5961 ± 0.0647	0.5134 ± 0.0089	-2.2	22.35 ± 0.61	369
AEPBC	NF-BC	0.7621 ± 0.0031	0.3163 ± 0.0106	45.1	26.92 ± 0.23	491
AEPBC	FM-BC	0.8977 ± 0.0064	0.2779 ± 0.0018	56.0	24.16 ± 0.10	503
AEPBC	FM-BC + PAR	0.9058 ± 0.0022	0.0194 ± 0.0019	63.9	24.30 ± 0.11	565
AEPBC	ReinFlow	0.9256 ± 0.0182	0.2845 ± 0.0078	55.5	26.30 ± 0.58	530
AEPBC	QC-FQL	0.9428 ± 0.0021	0.2746 ± 0.0022	57.9	24.62 ± 0.05	644
AEPBC	GLRP (one-shot)	0.4710 ± 0.0278	0.3344 ± 0.0017	37.8	20.55 ± 0.21	344
AWW	NF-BC	0.6989 ± 0.0014	0.2569 ± 0.0059	76.7	27.56 ± 0.06	399
AWW	FM-BC	0.7774 ± 0.0123	0.2539 ± 0.0061	79.9	27.27 ± 0.10	417
AWW	FM-BC + PAR	0.8150 ± 0.0067	0.0227 ± 0.0039	84.2	27.52 ± 0.09	486
AWW	ReinFlow	0.8219 ± 0.0145	0.2547 ± 0.0133	80.6	28.40 ± 0.03	416
AWW	QC-FQL	0.8583 ± 0.0031	0.2399 ± 0.0033	84.3	26.39 ± 0.18	404
AWW	GLRP (one-shot)	0.5662 ± 0.0306	0.3010 ± 0.0103	59.8	20.35 ± 0.54	287

Table 4: Baselines on the other anchorage regions, AESPA (7,068 test episodes), AEPBC (4,219), and AWW (2,454), under 3 seeds.

We control one vessel among recorded traffic on the full test populations of AEW, AESPA, AEPBC, and AWW, training separate policies with three seeds per region and scoring them with the metrics of Section 3.4. Table 3 compares all baselines on AEW, and Table 4 the main ones on the other three regions; remaining baselines are in Appendix H. $\pm$ values are standard deviations over seeds.

Open-loop trajectory prediction is not enough. Executed open-loop, GLRP and TrAISformer reach the goal in only 53.2% and 22.6% of AEW episodes, with collision rates of 24.4% and 50.8%. GLRP nevertheless has the lowest gc-ADE in Table 3 (16.84 against 23.18 for FM-BC), and the same holds on the other regions: staying close to the logged track does not imply reaching the goal, which is why closed-loop evaluation matters. Their motion is also implausible for a large ship: the forecasts turn far more sharply than recorded vessels, reaching 5–17 times the maximum curvature of human replay, while all closed-loop policies except unregularized PPO stay below it (Table 23).

Flow-matching imitation is a strong baseline. FM-BC reaches the goal in 89.4% of AEW episodes, against 75.4% for NF-BC, and has higher R_G and lower gc-ADE and collision rate than NF-BC in every region. Its R_G ranges from 77.7% on AWW, the region with the smallest training set, to 93.1% on AESPA. Yet FM-BC collides in 20.4–45.1% of episodes; on AESPA its median CPA is 8.5 m, against 76.0 m for human replay.

Policy improvement raises goal reaching, not safety. ReinFlow and QC-FQL raise R_G over FM-BC by 1.7–4.5 and 2.6–8.1 points across the four regions, while their collision rates stay within 1.4 points of FM-BC's. QC-FQL keeps gc-ADE within 1.1 of FM-BC, whereas ReinFlow increases it (25.48 against 23.18 on AEW). HR-PPO reaches 88.0% on AEW with higher tracking error, and PPO is unstable across seeds ($33.8 \pm 47.5\%$). DEAS-BoN attains the highest AEW R_G, 96.7%, 7.2

Region	Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA↑ (m)	gc-ADE ↓	FDE$_{\neg R}$ ↓ (m)
Intersection 1	BC-NF	0.8777 ± 0.0145	0.1628 ± 0.0189	160.6 ± 12.9	65.56 ± 0.94	713
Intersection 1	BC-FM	0.9463 ± 0.0474	0.1328 ± 0.0176	198.0 ± 18.0	55.32 ± 3.17	221
Intersection 1	Independent PPO	0.9331 ± 0.1133	0.6544 ± 0.0254	-16.7 ± 1.0	266.76 ± 11.22	181
Intersection 1	HR-PPO	0.9959 ± 0.0012	0.1036 ± 0.0022	235.3 ± 2.7	53.61 ± 0.50	210
Intersection 2	BC-NF	0.7614 ± 0.0606	0.1737 ± 0.0182	160.6 ± 15.5	83.66 ± 11.17	1108
Intersection 2	BC-FM	0.9693 ± 0.0198	0.1048 ± 0.0090	236.4 ± 13.2	48.70 ± 1.22	187
Intersection 2	Independent PPO	0.6829 ± 0.2661	0.5749 ± 0.0991	-10.3 ± 16.9	213.04 ± 27.53	595
Intersection 2	HR-PPO	0.9962 ± 0.0004	0.0861 ± 0.0033	266.4 ± 2.0	44.45 ± 0.67	152
TSS East	BC-NF	0.8288 ± 0.0293	0.1818 ± 0.0163	212.0 ± 19.6	90.28 ± 3.40	1043
TSS East	BC-FM	0.9123 ± 0.0176	0.1438 ± 0.0037	253.0 ± 8.1	82.92 ± 1.45	196
TSS East	Independent PPO	0.9253 ± 0.1136	0.6825 ± 0.0261	-25.8 ± 2.1	477.94 ± 4.98	197
TSS East	HR-PPO	0.9363 ± 0.0064	0.1407 ± 0.0064	263.0 ± 12.9	82.69 ± 1.79	184
TSS West	BC-NF	0.7961 ± 0.0744	0.1302 ± 0.0074	376.2 ± 44.3	96.11 ± 10.05	1359
TSS West	BC-FM	0.9439 ± 0.0072	0.1145 ± 0.0107	395.4 ± 15.1	75.14 ± 2.61	205
TSS West	Independent PPO	0.9947 ± 0.0021	0.5618 ± 0.0075	-15.4 ± 2.0	467.38 ± 0.34	217
TSS West	HR-PPO	0.4320 ± 0.1281	0.0662 ± 0.0174	649.9 ± 12.4	89.87 ± 4.76	606

Table 5: Four-method comparison under 3 seeds on the four TSS and intersection regions, with every vessel under policy control at the same time. R_G is scored at a goal radius $r_G = 200$ m.

points above FM-BC, but its collision rate stays at 20.5% and its gc-ADE rises to 31.77. Better task completion therefore does not translate into safer navigation.

PAR cuts collisions by an order of magnitude. FM-BC + PAR lowers the collision rate to 1.1–3.4% in all four regions, a 91.1–94.5% reduction relative to FM-BC. R_G changes by $+0.1$, -0.5, $+0.8$, and $+3.8$ points on AEW, AESPA, AEPBC, and AWW, and gc-ADE by at most 0.3. Median CPA rises in every region, most on AESPA, from 8.5 to 31.6 m. On AEW, PAR removes 98% of FM-BC's collisions with anchored vessels and 76% of those with moving vessels, which account for most of the residual collisions (Appendix L).

5.3 Multi-agent setting and additional dynamics-models

To evaluate multi-agent behavior, we place every qualifying vessel in the four navigation regions under independent copies of the same trained policy. Each vessel acts from its own local observations and goal, so all vessels respond to one another in closed loop. Table 5 reports the multi-agent results. Compared with the same region single-agent navigation results in Appendix O, collision rates generally increase when all vessels are controlled simultaneously. These rates are nevertheless lower than those in the single-policy anchorage results. These results also suggest future scope for safe multiagent RL and IL methods.

Dynamics models. We also examine how the dynamics model and sea current affect policy behavior. Appendix N.9 compares the four dynamics models in TSS East and the anchorage regions, including policies trained without a sea current but tested in a current-enabled simulator and policies trained with the current. It also separates the effects of including the sea current information in the simulator, expert action labels, or policy observations, and reports R_G, collision rate, CPA, and gc-ADE. Across these comparisons, sea-current conditioning consistently lowers gc-ADE. Models whose inverse dynamics reproduce the recorded track retain high R_G, whereas Port-glide's irreducible sideways-motion error causes a substantial drop in goal reaching.

6 Conclusion

We presented JaxNaviSim, a data-driven maritime navigation benchmark that combines expert demonstrations, controllable traffic simulation, and evaluation of task completion and behavior across anchorage and TSS settings. Our results show that learned policies reach their goals at high rates but often collide, and that our Predictive Action Refinement (PAR) substantially reduces these collisions. Our work moves toward the realistic simulations port authorities need to assess how scheduling, anchorage, and routing decisions affect throughput and safety.

Reproducibility statement

Source code to reproduce our experiments is available in the Supplementary Material. The benchmark definition specifies the dataset screens, temporal split, observations, action limits, evaluation windows, and collision geometry. Appendix E gives the learning settings, and Appendices G and H report additional results. Per-run configuration records identify the data source, population, and episode protocol used by each evaluation. Reproducing stochastic comparisons requires the same checkpoints, batch plans, random streams, software versions, and hardware configuration. The current measurements use test-set checkpoint selection, as stated in Appendix E.

Data availability and ethics

The benchmark's expert AIS demonstrations are publicly available. Section 3.3 and Appendix B document the population, task construction, and temporal splits. Vessel movement histories and registry attributes can reveal operational information; removing direct identifiers alone does not establish anonymity. The benchmark is intended for research on simulated navigation and does not validate policies for operational vessel control.

AI use statement

The research ideas, methodology, and experimental design of this work were led by the authors, who also divided the work among themselves, supervised every stage, and verified the results. We used generative AI tools as assistants: to search for and collect related work, to help implement and debug code under the authors' direction and review, and to revise the manuscript for readability, including checking spelling, the consistency of mathematical notation, LaTeX errors, and the consistency of reported results between the text and tables. AI tools did not generate experimental data; all reported results come from experiments that were actually run and from publicly available published data. The authors take full responsibility for this paper, and no AI tool is an author.

References

Lucas Agussurja, Akshat Kumar, and Hoong Chuin Lau. Resource-constrained scheduling for maritime traffic management. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. doi: 10.1609/aaai.v32i1.12086. URL https://ojs.aaai.org/index.php/AAAI/article/view/12086.

Kevin Black, Noah Brown, Danny Driess, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024. URL https://arxiv.org/abs/2410.24164.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs. http://github.com/google/jax, 2018.

Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric Wolff, Alex Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. nuPlan: A closed-loop ML-based planning benchmark for autonomous vehicles. *arXiv preprint arXiv:2106.11810*, 2021. URL https://arxiv.org/abs/2106.11810.

Eric P. Chassignet, Harley E. Hurlburt, Ole Martin Smedstad, George R. Halliwell, Patrick J. Hogan, Alan J. Wallcraft, Remy Baraille, and Rainer Bleck. The HYCOM (HYbrid Coordinate Ocean Model) data assimilative system. *Journal of Marine Systems*, 65(1–4):60–83, 2007. doi: 10.1016/j.jmarsys.2005.09.016. URL https://doi.org/10.1016/j.jmarsys.2005.09.016. Data server: https://www.hycom.org/dataserver.

Daphne Cornelisse and Eugene Vinitsky. Human-compatible driving agents through data-regularized self-play reinforcement learning. *Reinforcement Learning Journal*, 5:2320–2344, 2024. URL https://rlj.cs.umass.edu/2024/papers/Paper338.html.

Paul Corry and Christian Bierwirth. The berth allocation problem with channel restrictions. *Transportation Science*, 53(3):708–727, 2019. doi: 10.1287/trsc.2018.0865.

Danish Maritime Authority. AIS data, 2026. URL https://www.dma.dk/safety-at-sea/navigational-information/ais-data. Accessed: 2026-09-22.

Daniel Dauner, Marcel Hallgarten, Andreas Geiger, and Kashyap Chitta. Parting with misconceptions about learning-based vehicle motion planning. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 1268–1281, 2023. URL https://proceedings.mlr.press/v229/dauner23a.html.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using Real NVP. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1605.08803.

K. D. Do. Practical control of underactuated ships. *Ocean Engineering*, 37(13):1111–1119, 2010. doi: 10.1016/j.oceaneng.2010.04.007. URL https://doi.org/10.1016/j.oceaneng.2010.04.007.

Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R. Qi, Yin Zhou, Zoey Yang, Aurélien Chouard, Pei Sun, Jiquan Ngiam, Vijay Vasudevan, Alexander McCauley, Jonathon Shlens, and Dragomir Anguelov. Large scale interactive motion forecasting for autonomous driving: The Waymo Open Motion Dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9710–9719, 2021. doi: 10.1109/ICCV48922.2021.00957.

Jesse Farebrother, Jordi Orbay, Quan Vuong, Adrien Ali Taïga, Yevgen Chebotar, Ted Xiao, Alex Irpan, Sergey Levine, Pablo Samuel Castro, Aleksandra Faust, Aviral Kumar, and Rishabh Agarwal. Stop regressing: Training value functions via classification for scalable deep RL. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 2024. URL https://proceedings.mlr.press/v235/farebrother24a.html.

Thor I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, Chichester, UK, 2011. ISBN 9781119991496. doi: 10.1002/9781119994138. URL https://doi.org/10.1002/9781119994138.

Raj Ghugare and Benjamin Eysenbach. Normalizing flows are capable models for continuous control. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Cole Gulino, Justin Fu, Wenjie Luo, George Tucker, Eli Bronstein, Yiren Lu, Jean Harb, Xinlei Pan, Yan Wang, Xiangyu Chen, John D. Co-Reyes, Rishabh Agarwal, Rebecca Roelofs, Yao Lu, Nico Montali, Paul Mougin, Zoey Yang, Brandyn White, Aleksandra Faust, Rowan McAllister, Dragomir Anguelov, and Benjamin Sapp. Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1838feeb71c4b4ea524d0df2f7074245-Abstract.html.

Shuai Jia, Lingxiao Wu, and Qiang Meng. Joint scheduling of vessel traffic and pilots in seaport waters. *Transportation Science*, 54(6):1495–1515, 2020. doi: 10.1287/trsc.2020.0990.

Napat Karnchanachari, Dimitris Geromichalos, Kok Seang Tan, Nanxiang Li, Christopher Eriksen, Shakiba Yaghoubi, Noushin Mehdipour, Gianmarco Bernasconi, Whye Kit Fong, Yiluan Guo, and Holger Caesar. Towards learning-based planning: The nuPlan benchmark for real-world autonomous driving. In *IEEE International Conference on Robotics and Automation*, 2024. URL https://arxiv.org/abs/2403.04133.

Saman Kazemkhani, Aarav Pandya, Daphne Cornelisse, Brennan Shacklett, and Eugene Vinitsky. GPUDrive: Data-driven, multi-agent driving simulation at 1 million FPS. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2408.01584.

Changyeon Kim, Haeone Lee, Younggyo Seo, Kimin Lee, and Yuke Zhu. DEAS: DEtached value learning with action sequence for scalable offline RL. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/68b7f71467c83d5d32275c5acbf6b588-Abstract-Conference.html.

Huanhuan Li, Hang Jiao, and Zaili Yang. Ship trajectory prediction based on machine learning and deep learning: A systematic review and methods analysis. *Engineering Applications of Artificial Intelligence*, 126:107062, 2023. doi: 10.1016/j.engappai.2023.107062. URL https://doi.org/10.1016/j.engappai.2023.107062.

Qiyang Li, Zhiyuan Zhou, and Sergey Levine. Reinforcement learning with action chunking. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Yueyuan Li, Wei Yuan, Songan Zhang, Weihao Yan, Qiyuan Shen, Chunxiang Wang, and Ming Yang. Choose your simulator wisely: A review on open-source simulators for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 9(5):4861–4876, 2024. doi: 10.1109/TIV.2024.3374044.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.02747.

Kun Ma, Qilong Han, Chengjing Song, Jingzheng Yao, Hao Wang, and Changmao Wu. EnvShip: A unified framework for context-aware and cross-region vessel trajectory forecasting, 2026. URL https://arxiv.org/abs/2606.15240.

MarineCadastre.gov. AccessAIS, 2026. URL https://marinecadastre.gov/accessais/. Accessed: 2026-09-22.

Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S. Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. In *IEEE International Conference on Robotics and Automation*, 2023. URL https://arxiv.org/abs/2207.05844.

Duong Nguyen and Ronan Fablet. TrAISformer—a transformer network with sparse augmented data representation and cross entropy loss for AIS-based vessel trajectory prediction. *IEEE Access*, 2024. doi: 10.1109/ACCESS.2024.3349957. URL https://arxiv.org/abs/2109.03958.

K. Nomoto, T. Taguchi, K. Honda, and S. Hirano. On the steering qualities of ships. *International Shipbuilding Progress*, 4(35):354–370, 1957. doi: 10.3233/ISP-1957-43504. URL https://journals.sagepub.com/doi/abs/10.3233/ISP-1957-43504.

Seohong Park, Qiyang Li, and Sergey Levine. Flow Q-learning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48104–48127. PMLR, 2025. URL https://proceedings.mlr.press/v267/park25f.html.

Quang Anh Pham, Janaka Chathuranga Brahmanage, and Akshat Kumar. ShipNaviSim: Data-driven simulation for real-world maritime navigation. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pp. 1641–1649, 2025. URL https://www.ifaamas.org/Proceedings/aamas2025/pdfs/p1641.pdf.

Ralf Römer, Julian Balletshofer, Jakob Thumm, Marco Pavone, Angela P. Schoellig, and Matthias Althoff. From demonstrations to safe deployment: Path-consistent safety filtering for diffusion policies. In *IEEE International Conference on Robotics and Automation*, 2026. URL https://arxiv.org/abs/2511.06385.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15, pp. 627–635, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Vaishnav Vaidheeswaran, Dilith Jayakody, Samruddhi Mulay, Anand Lo, Md Mahbub Alam, and Gabriel Spadon. Goal-conditioned reinforcement learning for data-driven maritime navigation. *arXiv preprint arXiv:2509.01838*, 2025. URL https://arxiv.org/abs/2509.01838.

Eugene Vinitsky, Nathan Lichtlé, Xiaomeng Yang, Brandon Amos, and Jakob Foerster. Nocturne: A scalable driving benchmark for bringing multi-agent learning one step closer to the real world. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/191e9e721a2748a860714fb23aaf7c5d-Abstract-Datasets_and_Benchmarks.html.

Kim P. Wabersich and Melanie N. Zeilinger. A predictive safety filter for learning-based control of constrained nonlinear dynamical systems. *Automatica*, 129:109597, 2021. doi: 10.1016/j.automatica.2021.109597. URL https://arxiv.org/abs/1812.05506.

Kaisen Yang, Dong Yang, and Yuxu Lu. Enhancing risk perception by integrating ship interactions in multi-ship encounters: A graph-based learning method. *Reliability Engineering & System Safety*, 261:111150, 2025. doi:10.1016/j.ress.2025.111150. Authors' code https://github.com/KaysenWB/RESS_GLRP, commit 79e1ec6.

Ying Yang, Yang Liu, Guorong Li, Zekun Zhang, and Yanbin Liu. Harnessing the power of machine learning for AIS data-driven maritime research: A comprehensive review. *Transportation Research Part E: Logistics and Transportation Review*, 183:103426, 2024. doi: 10.1016/j.tre.2024.103426. URL https://doi.org/10.1016/j.tre.2024.103426.

Ruiqi Zhang, Jing Hou, Florian Walter, Shangding Gu, Jiayi Guan, Florian Röhrbein, Yali Du, Panpan Cai, Guang Chen, and Alois Knoll. Multi-agent reinforcement learning for autonomous driving: A survey, 2024.

Tonghe Zhang, Chao Yu, Sichang Su, and Yu Wang. ReinFlow: Fine-tuning flow matching policy with online reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/98d6f928497d9eaad9f320fb2db3040d-Abstract-Conference.html.

Xiaocai Zhang, Zhe Xiao, Maohan Liang, Tao Liu, Haijiang Li, and Wenbin Zhang. Realistic curriculum reinforcement learning for autonomous and sustainable marine vessel navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 39612–39619, 2026. doi: 10.1609/aaai.v40i46.41313.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems*, 2023. URL https://arxiv.org/abs/2304.13705.

A LIMITATION

The anchorage study uses displacement dynamics and replayed neighbours, without reciprocal responses or hydrodynamic validation. The dynamics models of Appendix N are kinematic: they assert how a vessel is steered and how the current displaces it, and no coefficient in them is fitted to the recordings. The current gains are set to unit advection rather than estimated, and the surface field is a reanalysis product, not a measurement in these waters. That comparison is a single seed on one region-day, so it separates the dynamics models but does not establish their effect sizes. PAR limits replacements but leaves nominal actions unchanged. Sampled collisions depend on metadata and observation coverage; goal reaching does not establish successful anchoring or regulatory compliance. The four regions belong to one port, with small vessels excluded. Repeated vessel identities, test-set model selection, and method-specific budgets limit broader generalization claims.

B DATASET DOCUMENTATION

Purpose and unit of analysis. The dataset supports learning and evaluating goal-conditioned vessel navigation within an anchorage group. Each episode corresponds to a trajectory in a daily ego manifest. Its key is the pair consisting of the source-day identifier and trajectory identifier; trajectory identifiers alone are not globally unique. Training and evaluation are separate for AEW, AESPA, AEPBC, and AWW. An episode specifies an initial vessel state, surrounding traffic, and a destination inferred from its observed ending.

Source products. The group-wide package contains ego trajectories and manifests, scene trajectories for surrounding vessels, an alongside-analysis table for arrivals, and path-shape statistics. Shared reference files provide vessel particulars and region and destination polygons. Length and beam support capsule geometry, while the AIS ship-type field determines the ego-type screen. Beam resolution prioritizes a quality-controlled vessel table, then the product's vessel breadth, then an AIS-type-group median. Coverage through this procedure does not imply that every dimension is directly measured or that every identifier join is correct.

Scale before benchmark filtering. The source collection covers four anchorage groups (AEW, AESPA, AEPBC, and AWW) and four strait regions (INTERSECTION1, INTERSECTION2, TSS_EAST_24KM, and TSS_WEST_24KM), with 365 daily ego products per region. The released positions have already been extracted from AIS and interpolated at a nominal 10 s resolution. A regional trajectory segment is an extracted pass through a region; one source trajectory identifier can contribute multiple passes. We take pass counts from the daily extraction metadata. The 2,920 ego files contain 622,758 segments with 447,304,145 samples and 446,681,387 transitions. For segment i with L_i samples, these counts are $N_{\text{sample}} = \sum_i L_i$ and $N_{\text{transition}} = \sum_i \max(L_i - 1, 0)$. The nominal aggregate trajectory duration is $10N_{\text{transition}}/3600 = 1{,}240{,}781.6$ vessel-hours, summing $10\max(L_i - 1, 0)$ seconds per regional segment. This uses the nominal 10 s interval between successive samples; actual timestamp intervals can differ slightly because of upstream rounding. Counts retain any overlap between regions. The separate surrounding-traffic product contains 2,747,794,743 position records and is excluded from these totals. Transitions exclude boundaries between separate passes, including passes that share a source trajectory identifier.

Vessel identifiers. The ego products contain 20,172 distinct International Maritime Organization (IMO) ship identifiers, deduplicated across all days and regions. Every ego row has a seven-digit IMO value that passes the check-digit test; this does not independently verify the identifier against registry records. The released IMO field is joined from a year-wide table keyed by MMSI, so counting IMO values cannot recover vessel distinctions already lost when different ships share an MMSI.

Episode semantics. The alongside analysis labels an arrival using the vessel's stopping behaviour and its relation to a nearby companion. The benchmark retains arrivals labelled as anchoring and applies the additional length-ratio rule to serviced arrivals. These are inferred task labels. The source documentation reports 58 manually labelled cases, with agreement on 50 of 54 cases for which a decision was available. This small audit provides limited evidence about the selection

heuristic; it is not a representative error estimate for all regions. The transit tortuosity threshold has no corresponding human-labelled validation in the saved evidence.

Episode-selection details. We retain episodes lasting 5–60 minutes, with mean speed of at least 1 knot and position-implied step speeds below 40 knots. Position-implied speed is used to identify corrupted position sequences that may coexist with plausible broadcast speed values. For arrivals, we retain records classified as anchoring by the supplied alongside analysis. For arrivals classified as the serviced vessel, we additionally require an ego-to-companion length ratio of at least 1.5 to reduce ambiguous assignments. For transits, we require path tortuosity, defined as the ratio of path length to endpoint displacement, to be at most 1.3. The anchoring and size-ratio rules apply only to arrivals, whereas the tortuosity rule applies only to transits.

Region	Episode screen train / test	After vessel-size screen train / test	Final benchmark train / test
AEW	24,829 / 8,514	23,795 / 8,141	22,258 / 7,600
AESPA	22,642 / 7,810	21,861 / 7,574	20,396 / 7,068
AEPBC	13,079 / 4,605	12,494 / 4,400	11,985 / 4,219
AWW	11,687 / 3,704	9,245 / 3,041	7,404 / 2,454
Total	72,237 / 24,633	67,395 / 23,156	62,043 / 21,341

Table 6: Population counts through dataset construction. The first column applies the terminal-event, duration, speed, anchoring, size-ratio, and tortuosity rules. The second removes small and unknown-length vessels from all scenes. The final column gives the benchmark population, restricted to cargo and tanker egos.

Temporal provenance and missing observations. The recorded split uses 273 source-day identifiers for training and 92 for testing. The identifiers select files; timestamps specify when observations occurred. The source documentation records that the file named `20240122` contains observations from the evening of 25 January through the morning of 26 January, and that `20240125` is absent. This documented case remains within the training period. Scene files also overlap neighbouring files in time, while ego files abut. Consequently, the number of daily products should not be read as a claim of complete calendar-day coverage or of timestamp-level independence between scenes. The package reports gaps on 31 days, with a median blackout duration of 47 minutes. Missing states remain subject to validity masking during evaluation.

Intended scope and access. The expert AIS demonstrations are publicly available. The anchorage benchmark represents cargo and tanker navigation among vessels of known length at least 30 m in four regions of one port. It does not cover the excluded harbour craft or establish transfer to another port. Vessel identities may recur across the temporal split.

C Simulator details

Figure 4 summarizes how recorded data, the simulator, and policy training connect. Figure 5 illustrates an anchorage episode.

C.1 AIS vessel type

The AIS ship-and-cargo type is a two-digit code whose tens digit gives the vessel class. The simulator observes the raw code; the policies receive it as a one-hot vector over the seven classes of Table 7, for the controlled vessel and for each observed neighbour. In the anchorage benchmark, the controlled vessels are cargo and tanker vessels (codes 70–89).

C.2 Action bounds

The delta action of Equation 2 is constrained to

$$\mathcal{A} = \left\{ a : \sqrt{\Delta x^2 + \Delta y^2} \leq 205.78 \text{ m}, \quad |\Delta\psi| \leq \pi/4 \right\}, \tag{3}$$

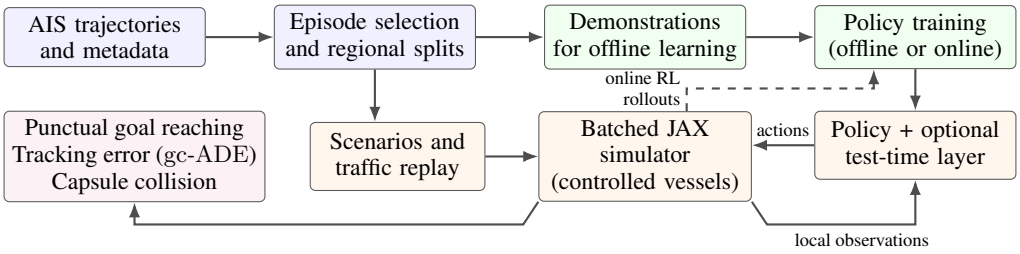


Figure 4: The JAXNAVISIM pipeline. Recorded AIS trajectories supply offline demonstrations and the replayed traffic of each scenario. Policies are trained offline from demonstrations or online from closed-loop rollouts, in which one or more controlled vessels act among replayed traffic, optionally through a test-time layer. Policies receive only current local observations; only the simulator accesses future replay samples. Section 3.4 defines the metrics.

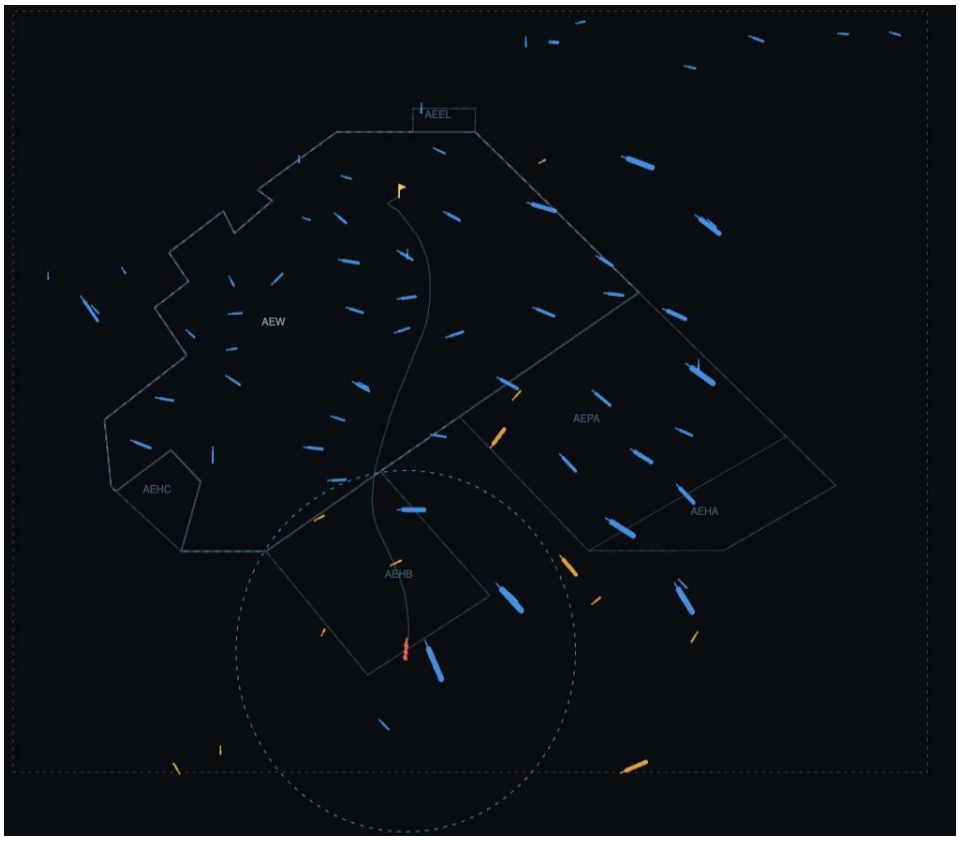


Figure 5: **Goal-reaching navigation in an anchorage region.** The controlled vessel is shown in red and its assigned destination by the yellow flag. The green curve shows the corresponding recorded reference path, while the dashed circle indicates the local traffic-observation range. The surrounding vessel glyphs and anchorage boundaries provide the traffic and geographic context in which the policy must reach the destination.

Class	AIS codes	Vessels
Unknown	0	type not available
Fishing and operations	30–39	fishing, towing, dredging, diving, military, sailing
Special	50–59	pilot, search and rescue, tug, port tender, law enforcement, medical
Passenger	60–69	passenger
Cargo	70–79	cargo
Tanker	80–89	tanker
Other	all other codes	including 1–29, 40–49, 90–99, and out-of-range codes

Table 7: Coarse vessel classes derived from the AIS ship-and-cargo type.

that is, at most 40 knots of displacement and a 45° heading change per 10 s step.

C.3 Standard shaped reward

QC-FQL, the critic used for candidate ranking, ReinFlow, PPO, and HR-PPO use one reward weighting. For a transition from s to s', let I_g, I_c, and I_o indicate goal arrival, a sampled collision under the capsule-based metric, and exit from the region, respectively. The reward is

$$r(s, s') = 0.05\big[200I_g + \gamma\Phi(s') - \Phi(s) - 20I_c - b(s') - 50I_o\big], \qquad \gamma = 0.99, \tag{4}$$

where $\Phi(s) = -\|p^e - g\|/(100\,\mathrm{m})$ and terminal potential is zero, including at the step budget. The clearance penalty uses checkable ego–neighbour pairs:

$$\begin{aligned} b(s') &= \max_j \left[\max\left(0, 1 - \frac{d_{ej}(s')}{m_{ej}(s')}\right)\right]^2, \\ m_{ej} &= \max\big(50\,\mathrm{m},\ 2\max(B_e, B_j),\ (60\,\mathrm{s})v_{ej}^{\mathrm{close}}\big). \end{aligned} \tag{5}$$

Here d_{ej} is signed capsule clearance, B is beam, and $v_{ej}^{\mathrm{close}} \geq 0$ is the closing range rate estimated from simulated history. The penalty is zero if no pair is checkable. The closing-speed margin is not capped; the separate fixed-margin reward option is unused. The clearance, collision, and exit penalties change the objective in addition to the goal-distance potential shaping.

The offline transitions were relabeled with this weighting before training QC-FQL and the regional critics; the recorded observations, actions, and episode boundaries are unchanged. The factor 0.05 is already in the stored offline reward, so QC-FQL's additional reward multiplier is 1. Simulator-trained methods apply the factor during training.

C.4 Evaluation windows and termination by region

In the anchorage regions, goal reaching allows a grace period of G steps after the logged demonstration ends: stopping at the last logged sample would treat every delayed arrival as a failure, while an unrestricted horizon can reward excessive delay. The rollout is not stopped at this deadline. It continues for up to E steps past the demonstration, so that a vessel arriving late or not at all is still checked for collisions and its arrival time remains observable:

$$E = \min(300, T), \qquad G = \min(10, \lfloor 0.1T \rfloor). \tag{6}$$

Let τ_g be the first step at which the rollout enters the goal radius, with $\tau_g = \infty$ if it never does. An episode counts as a success for R_G if

$$R_G = \mathbb{1}[\tau_g < T + G]. \tag{7}$$

The grace period is a benchmark tolerance for relative timing, not a port scheduling requirement. Success is position-based: arrivals are not required to stop, align their heading, or keep station, so R_G does not certify physical anchoring. The rollout stops earlier at goal arrival, region exit, or the global 1,000-step budget. After the demonstration ends, surrounding traffic is still replayed from the recorded day, and the collision indicator of Equation 10 covers the entire executed rollout. Missing neighbour samples are marked invalid and cannot be collision-checked (Appendix D).

Goal radius. Every run uses a 100 m goal radius, which defines *reached* for the shaped reward, the termination, and the evaluator alike. Anchorage results are reported at that radius. The navigation regions of Appendix N are reported at 200 m. Those regions cover far more water than the anchorage basins, spanning some 24 km end to end, and vessels cross them at service speed instead of manoeuvring to a stop at an assigned berth, so a fixed radius is a much tighter target there: a transiting vessel covers an appreciable fraction of 100 m within a single 10 s step of Equation 3, which makes arrival at that radius sensitive to sampling rather than to navigation. A 200 m radius is also the value ShipNaviSim (Pham et al., 2025) uses in this TSS setting, so the reported figure remains comparable with prior work. This needs no second run, because reach is radius-free after the fact:

$$\mathrm{reach}(r_G, W) = \mathbb{1}\left[\min_{t<W} \|p_t - g\| < r_G\right], \tag{8}$$

so Equation 7 is the case $r_G = 100\,\mathrm{m}$, and the 200 m figure is an exact re-scoring of the closest approach stored per episode.

The navigation-region horizon. The windows above follow a single ego's logged demonstration. The navigation regions instead use a fixed horizon of 1,000 steps from each ego's own entry, cut at region exit, and do not stop when that ego's logged demonstration ends. Under joint control the egos enter and leave at different times, so there is no common T to key a window to; using the same fixed horizon for both regimes keeps the two measured on one population. This replaces Equation 6 for the navigation regions only.

D METRIC DEFINITIONS AND IMPLEMENTATION CONVENTIONS

Goal-reaching rate (R_G). This metric measures the rate at which the ego vessel successfully reaches its goal. We use a goal radius r_G of 100–200 m, depending on the region; an episode counts as a success if the vessel comes within this radius of its goal within the evaluation window (Equation 7, Appendix C.4). Compared with the length of tankers and cargo ships (up to around 300 m), r_G is relatively small.

Goal-reaching index convention. For a reference with T samples indexed $0, \ldots, T-1$, the last index counted by R_G is $T+G-1$, with G from Equation 6. Goal arrival, region exit, and the global step budget can end the rollout sooner. The saved `reached_step` is the first goal-hit index, or -1 on failure. The saved arrival-delay convention is `reached_step` $- T$, so arrival at the last reference sample has value -1. This index convention differs by one step from elapsed delay relative to that last sample.

Collision rate. We approximate a vessel of length L_i and beam B_i by a capsule: a segment S_i of length $\max(L_i - B_i, 0)$ along its heading, expanded by radius $B_i/2$. The signed clearance between vessels i and j is

$$d_{ij}(t) = \mathrm{dist}(S_i(t), S_j(t)) - \frac{B_i + B_j}{2}, \tag{9}$$

and an episode has a collision if

$$C = \mathbb{1}\left[\min_{(t,j)\in\mathcal{V}} d_{ej}(t) \le 0\right], \tag{10}$$

where e is the ego and $\mathcal{V}$ contains the valid ego–neighbour pairs over the entire executed rollout. Collisions between two replayed neighbours are not attributed to the policy. The metric checks overlap of approximate footprints at 10 s samples, so it can miss contacts between samples and does not measure regulatory responsibility.

Closest point of approach (CPA). The CPA of an episode is its minimum signed clearance $\min_{(t,j)\in\mathcal{V}} d_{ej}(t)$, measured along the executed rollout rather than predicted from initial velocities. We report its median over episodes in metres.

Collision-evaluation window and observation access. The headline collision rate covers the full executed rollout, including the initial sample and any extension beyond the demonstration. It uses valid ego–neighbour pairs in the simulated scene, which can contain vessels outside the policy's ten-neighbour observation. This distinction matters for the controller: its forecast uses only locally observed neighbours, whereas the evaluator can detect collisions with other valid scene vessels. No interpolation is applied between native 10 s samples. A missing neighbour sample cannot contribute a checked pair at that step. These conventions limit the interpretation of a zero collision indicator.

Tracking error (gc-ADE, FDE$_{\neg R}$). Following ShipNaviSim, gc-ADE compares the simulated and logged ego positions $\hat{p}_t$ and p_t over the $n = |\mathcal{I}|$ valid steps $\mathcal{I}$ of the logged demonstration, evaluated only while a valid reference comparison exists:

$$\text{gc-ADE} = \frac{\sqrt{\sum_{t\in\mathcal{I}} \|\hat{p}_t - p_t\|_2^2}}{n} = \frac{\text{RMSE}}{\sqrt{n}}. \tag{11}$$

Thus, equal RMSE can produce different gc-ADE values when the comparison length changes. The main tables retain this legacy quantity for continuity with the updated benchmark. FDE is measured at the final compared sample; FDE$_{\neg R}$ averages it over episodes that miss the goal deadline ($R_G = 0$). If a rollout terminates early, its tracking score does not by itself penalize the missing remainder; goal reaching and termination must be considered alongside tracking.

Kinematics. We report per-episode extrema of acceleration, heading-based curvature, and course–heading drift; Appendix K gives their conventions.

Aggregation. Each per-seed rate is the mean of binary episode indicators over the complete regional test set. Tables report the arithmetic mean across three seeds, with sample standard deviation where shown. The same episodes occur under each seed, so this standard deviation is not uncertainty over independent traffic populations. We do not pool the four regions into one leaderboard score.

All FM-BC rows use the checkpoint-sweep evaluation, and FM-BC + PAR runs the same selected FM-BC checkpoints on the same test episodes. Deterministic XLA settings improve repeatability on a fixed configuration; exact identity across GPU models is not assumed. Standard deviations across three policy seeds describe variation between the trained policies and their stochastic evaluations; they are not confidence intervals over independent real-world deployments. DEAS-BoN shares one regional critic across these seeds.

E Learning settings and data access

Network architecture. Figure 6 shows the FM-BC actor and the DEAS-BoN critic; each has its own encoder of the same architecture (Nayakanti et al., 2023). *Encoder.* The observation o is split into three token groups: the ego vessel, its ten nearest neighbours (each with its current and previous state), and the goal. A token MLP embeds the tokens of each group. Each group then passes separately through two blocks, each consisting of cross-attention from 8 learned latent vectors to the group's tokens, a feed-forward layer, self-attention among the latents, and a second feed-forward layer. The latents of each group are averaged into a 128-dimensional vector, and the three group vectors are concatenated into the representation $e \in \mathbb{R}^{384}$. *Actor.* A linear layer maps e to $h \in \mathbb{R}^{128}$. The actor generates a chunk A of $k = 4$ actions $(\Delta x, \Delta y, \Delta\psi)$ (Equation 2). Let $\text{vec}(A) \in \mathbb{R}^{12}$ flatten the chunk, and let D_r be the diagonal matrix of the componentwise action normalizers of region r (Table 9), repeated over the four actions; the flow then operates on the normalized chunk $z = D_r^{-1}\,\text{vec}(A)$. The velocity network $v_\theta(h, z, u)$, with parameters θ, is an MLP on the concatenation $[h; z; u]$, where $u \in [0, 1]$ is the flow time. It has four hidden layers of width 1,024, the first three with layer normalization before the GELU activation, and a 12-dimensional output layer (Dense 12). Because h is computed from o, Equation 12 writes the network as $v_\theta(o, z, u)$. Sampling starts from Gaussian noise $z_0 \sim \mathcal{N}(0, I)$ and applies $K = 3$ Euler steps, $z \leftarrow z + v_\theta(h, z, u)/K$ at $u = 0, 1/K, 2/K$. The result is denormalized to $D_r z$ and reshaped into four actions, each of which is projected into the bounded action set $\mathcal{A}$ (Equation 3). *Critic.* The critic, with parameters ϕ, estimates the value $Q_\phi(o, A)$ of executing chunk A after observation o. Its input is the concatenation $[e;\, D_r^{-1}\,\text{vec}(A)]$ of its encoder output and the normalized chunk.

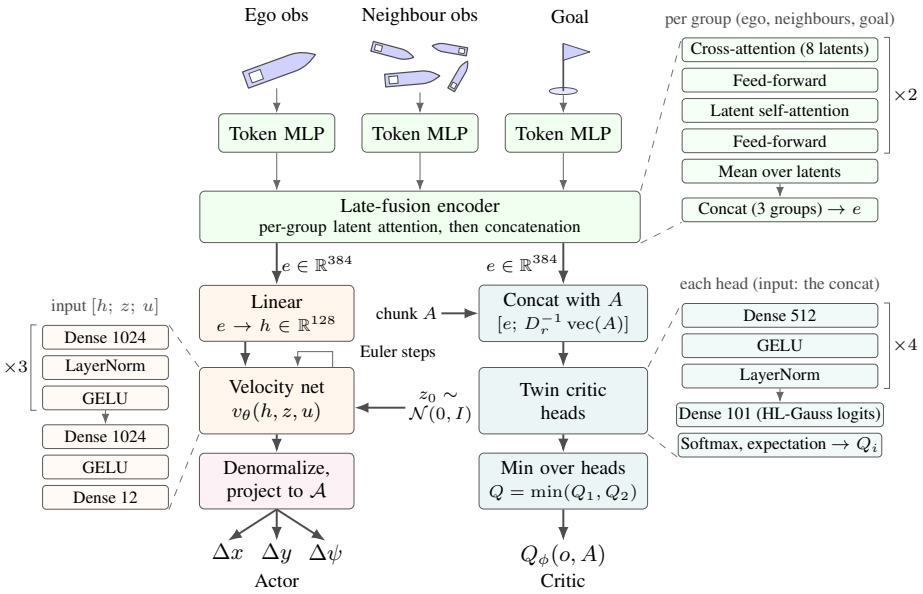


Figure 6: Actor (FM-BC, left) and critic (DEAS-BoN, right) networks. Both use the same late-fusion encoder architecture, with separate parameters. Dashed lines expand a block into its layers, and $\times n$ marks a group of layers repeated n times. The symbols are defined in the paragraph "Network architecture".

Two heads process this input separately. Each applies four blocks of a 512-unit dense layer, GELU activation, and layer normalization, followed by a dense layer with 101 outputs: the logits of a distribution over 101 fixed value bins (the HL-Gauss head described under DEAS-BoN below). The expectation of this distribution is the head's value Q_i, $i \in \{1, 2\}$, and the critic returns the smaller of the two, $Q_\phi(o, A) = \min(Q_1, Q_2)$, which limits value overestimation.

FM-BC objective and sampling. The FM-BC policy predicts a chunk $A = (a_t, \ldots, a_{t+k-1})$ with $k = 4$. With D_r and vec as above, let $z_1 = D_r^{-1}\,\mathrm{vec}(A)$ be a normalized expert chunk. Conditional flow matching (Lipman et al., 2023) trains a velocity network with

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{\substack{(o,A)\sim\mathcal{D}_r,\ z_0\sim\mathcal{N}(0,I) \\ u\sim\mathcal{U}[0,1]}} \left[\left\| v_\theta(o, (1-u)z_0 + uz_1, u) - (z_1 - z_0) \right\|_2^2 \right]. \tag{12}$$

Sampling uses three Euler updates from Gaussian noise, followed by denormalization and projection into $\mathcal{A}^k$. Four actions execute before resampling.

The three Euler steps in Table 8 are the evaluation setting used for the reported FM-BC sweeps. They override the generic sampler default stored in the training configuration; reproducing a result requires the evaluation arguments as well as the training configuration.

DEAS-BoN. A critic for each region is trained on reward-relabeled expert transitions, with the policy frozen, using expectile 0.9, discount 0.99, batch size 512, and one million gradient updates. The critic uses an HL-Gauss value head (Farebrother et al., 2024): each scalar TD target is smoothed into a Gaussian-weighted histogram over 101 fixed bins and fitted by cross-entropy, and the predicted value is the expectation over bin centres. At test time, best-of-N selection draws N candidate chunks from the policy and executes the one with highest predicted value:

$$A^\star = \arg \max_{A^{(i)},\ i=1,\ldots,N} Q_\phi(o, A^{(i)}). \tag{13}$$

NF-BC and QC-FQL. NF-BC uses a conditional RealNVP head with six coupling layers, the common late-fusion encoder, the same action chunks, and two million gradient updates at batch size 512. QC-FQL uses one million updates at batch size 512; actor and critic learning rates are 10^{-4}

FM-BC setting	Value
Encoder	Wayformer-style late fusion
Representation width	128
Observed history	Current and one preceding state
Observed neighbours	Ten nearest valid vessels within 1 nautical mile
Additional vessel features	AIS type and beam
Action chunk / execution	Four actions / four steps
Velocity network	Four hidden layers of width 1,024
Hidden-layer normalization	Layer normalization before activation
Training flow time	Uniform on $[0, 1]$
Sampling	Three Euler updates from Gaussian noise
Gradient updates / batch size	2,000,000 / 512
Learning rate	10^{-4}, cosine decay to the training horizon
L_2 regularization	10^{-6}
Exponential moving average	0.999
Input normalization	Running statistics
Action normalization	Regional training-action absolute p99.9
Action projection	40-knot displacement disc; 45° heading increment
Seeds	0, 1, 2
Retained checkpoints	1.91M–2.00M updates, spaced by 10,000
Selection	Highest test R_G; lower gc-ADE breaks ties

Table 8: Common FM-BC settings. The same recipe is applied separately to each regional dataset. A shared recipe does not imply equal numbers of passes through datasets of different sizes.

Region	$\lvert\Delta x\rvert$ (m)	$\lvert\Delta y\rvert$ (m)	$\lvert\Delta\psi\rvert$ (rad)
AEW	105.17	93.093	0.21136
AESPA	106.02	76.033	0.24734
AEPBC	108.56	57.422	0.22246
AWW	94.994	81.945	0.16107

Table 9: The componentwise 99.9th-percentile action normalizers. These quantities scale learning targets; the physical action bound is applied after denormalization.

and 3×10^{-4}, respectively. The frozen teacher is the selected FM-BC model of the same region and seed. The one-step actor is trained against the teacher's three-step Euler output at a shared Gaussian latent, with distillation weight 100 and an inverse-mean-absolute-Q normalization on the value term. Two critics are trained jointly with the actor, using their minimum for the actor objective and target. The target uses the learned actor's next chunk, discounted by 0.99^4, and a Polyak update coefficient of 0.005. Our implementation is an adaptation rather than the published QC-FQL or FQL recipe (Li et al., 2025; Park et al., 2025): the FM-BC teacher stays frozen rather than being trained jointly, training is purely offline, the value head is HL-Gauss, and the distillation weight and reward relabeling are our choices. Evaluation samples fresh latent noise; it is not zero-noise evaluation. The one-seed $\alpha = 1000$ and frozen-critic pilots are not additional seeds of the reported method.

Simulator-trained methods. Our ReinFlow adaptation is identified as R0 in the saved run configurations. It updates the flow policy using a stochastic sampling chain and an expert flow-matching regularizer; its reported evaluation uses the deterministic Euler integrator with a sampled initial latent. HR-PPO trains a per-step actor with a teacher regularizer of weight 0.1: a KL penalty toward FM-BC, estimated as the actor's negative log-likelihood of actions sampled from FM-BC. It uses zero additional action noise at evaluation. Each method uses the corresponding seed's FM-BC teacher. The teacher remains a four-action model even when the HR-PPO adaptation queries its first action at each step. PPO uses the same per-step actor recipe with the teacher penalty removed and zero additional evaluation noise. These methods differ in policy parameterization and execution frequency as well as optimization, and are not a single-factor algorithm comparison. We evaluate all three on every region, with method-specific checkpoint schedules.

Interaction budget and population. The saved AEW runs complete 1,525 ReinFlow iterations and 762 HR-PPO iterations, with their selected iterations shown in Appendix H. Simulator training draws from training-period scenario pools with a reference-length eligibility cap of 256 samples and an extended storage horizon of 512 samples. This population restriction differs from offline BC's full expert dataset. Table 10 records their different update schedules; equal checkpoint counts do not imply equal computation or interaction budgets.

Trajectory-prediction baselines. Each forecaster predicts a 360-step (60 min) track, which is converted to displacement actions and projected into $\mathcal{A}$ before execution. Both forecasters are run from the authors' code and read the same 180-step (30 min) history, repeat-padded because a full history exists for only 12.5% of AEW episodes. The 360-step horizon is longer than either model was published at. TrAISformer uses our vocabulary rather than the upstream latitude–longitude box: region-local metres on a 25 m grid, 0.25 m/s of speed, and 5° of heading. Its goal enters as two extra token positions at the head of the causal sequence, one per coordinate bin. At 58.5M parameters it is $11.7\times$ our policy and is run as published and unshrunk; the executed plan is its greedy decode. GLRP's scene is the ego and its ten exported neighbours, and only the ego is scored. Its standardization statistics are computed over the observation window only, rather than over past and future as upstream, and its checkpoint is selected on our held-out validation days. The goal enters GLRP as two extra channels on the ego node's input features and as a learned projection added to the decoder's initial hidden state; the executed plan is the deterministic latent-mean decode.

F Predictive Action Refinement details

The reported PAR configuration, labeled C3 in the saved parameter study (Table 18), uses capsule geometry, a 16-step (160 s) forecast horizon, zero clearance margin, and per-step monitoring of the remaining four-action queue. All candidate generation and traffic forecasting use the policy's current observation. The zero-margin feasibility predicate is a prediction under this model.

Feasibility test and selection. At step t, let A_t^{nom} be the four-action sequence proposed by the policy and $\mathcal{K}_t$ the candidate set: A_t^{nom} together with the slowdowns, turns, and goal-directed sequences constructed below. For each candidate $A \in \mathcal{K}_t$, PAR rolls the ego vessel e forward under A and forecasts each neighbour j observed at step t at constant velocity. Let $\hat{d}_{ej}(t+h; A)$ be the signed clearance of Equation 9 between the predicted capsules of e and j at h steps ahead, $h = 1, \ldots, H$

AEW setting	ReinFlow	HR-PPO
Parallel environments	256	1,024
Rollout per environment	64 four-action chunks	128 individual steps
Training iterations	1,525	762
PPO epochs per iteration	4	4
Minibatches per epoch	16	32
Actor learning rate	3×10^{-6}	3×10^{-4}
Critic learning rate	3×10^{-4}	3×10^{-4}
Discount / GAE parameter	0.99 / 0.95	0.99 / 0.95
PPO clipping parameter	0.2	0.2
Expert regularization	FM loss, weight 1.0	Teacher penalty, weight 0.1
Goal bonus / collision penalty	200 / 20	200 / 20
Overall reward scale	0.05	0.05
Retained checkpoint interval	152 iterations	76 iterations

Table 10: Recorded AEW RL settings. The final training iteration is also saved. Rollout budgets describe simulator sampling, not independent new AIS demonstrations. The two regularization weights multiply different losses.

with $H = 16$. The predicted clearance of A and the feasible set are

$$c_t(A) = \min_{(h,j)\in\widehat{\mathcal{V}}_t(A)} \hat{d}_{ej}(t+h;A), \qquad \mathcal{F}_t = \{A \in \mathcal{K}_t : c_t(A) \geq 0\}, \tag{14}$$

where $\widehat{\mathcal{V}}_t(A)$ contains the checked pairs of a horizon step h and an observed neighbour j; near the goal, it ends at A's predicted goal arrival (Goal handling below). A candidate is therefore feasible if its predicted hull overlaps no forecast neighbour over the checked horizon. PAR then executes

$$A_t^\star = \begin{cases} A_t^{\mathrm{nom}}, & \text{if } A_t^{\mathrm{nom}} \in \mathcal{F}_t \text{ and it is not stalled near the goal,} \\ \arg\max_{A\in\mathcal{F}_t} P_t(A), & \text{else if } \mathcal{F}_t \neq \emptyset, \\ \arg\max_{A\in\mathcal{K}_t} c_t(A), & \text{otherwise,} \end{cases} \tag{15}$$

where $P_t(A)$ is the predicted goal progress and the stalled condition is defined under Goal handling. The last case is a fallback, not a certified safe action.

Traffic forecast and ego rollout. Neighbour velocity is estimated by finite differences from its two observed positions. Positions are extrapolated at constant velocity; heading, length, and beam are held fixed. Candidate ego displacements and heading increments are rolled using Equation 2. Beyond the fourth action, the final displacement is repeated and the final heading is held. Predicted capsule separation uses the evaluator's geometric representation. The controller observes only ten neighbours, while the evaluator can check all valid scene vessels.

Candidate construction and the rate envelope. The 15 candidates are the unchanged nominal chunk, four slowdowns with target speed factors $0.75, 0.5, 0.25, 0$, six turns, and four goal-directed candidates. Following the path-reparameterization idea of PACS (Römer et al., 2026), slowdowns begin by resampling the nominal polyline at cumulative arc lengths. PAR sets a symmetric speed-rate limit $a_{\max} = 0.010\,\mathrm{m/s}^2$. Starting from the observed ego speed u_{-1}, each desired speed $\bar{u}_j$ is limited as

$$u_j = \max\{0, \mathrm{clip}(\bar{u}_j, u_{j-1} - a_{\max}\Delta t, u_{j-1} + a_{\max}\Delta t)\}. \tag{16}$$

A second projection applies this limit to the displacement magnitude divided by Δt, because a chord across a polyline corner can be shorter than the arc used for resampling. A degenerate segment inherits the last available direction, or the observed heading. This projection can change the sampled path, so exact path preservation and the reachability guarantees of PACS are not claimed.

Turn targets are $\pm 8°, \pm 16°, \pm 24°$, with offset ϕ applied through the schedule $\phi \min((j+1)/8, 1)$. Only the four generated actions are committed; an eight-step schedule does not mean that eight actions are executed before replanning. Replacement heading increments are clipped to $\pm 3°$. Goal-directed candidates also use the speed ramp and spread heading changes, approaching the goal without an instantaneous stop. The PAR setting disables the wide-turn and combination expansion pool. All candidates retain the simulator's displacement bound.

These limits apply to replacement actions. A geometrically feasible nominal chunk can still be kept when it violates the rate envelope. The legacy E0 ablation retains that nominal rule, uses instantaneous speed scaling and turns, and enables an 18-candidate expansion pool; its ordinary action bounds remain active. Neither PAR's heading limit nor its speed ramp establishes course-change limits or vessel-specific hydrodynamic feasibility under displacement control.

Selection and queue monitoring. The progress in Equation 15 is $P_t(A) = \|g - p_t\|_2 - \|g - \hat{p}_{t+4}(A)\|_2$, where $\hat{p}_{t+4}(A)$ is the predicted ego position after the four actions of A. Among feasible candidates, PAR prefers those on the passing side of its previous replacement when any is feasible, to avoid oscillating between sides. The nominal-deviation penalty is zero in the reported configuration. Before the next physical step, a new forecast checks the remaining queue and can trigger replanning.

Goal handling. Within 400 m of the goal, a candidate's clearance horizon ends at its own predicted first entry into the 100 m goal radius, including that arrival sample. At least one checked forecast sample remains. In this region, a feasible nominal chunk with less than 10.4 m predicted progress over the four committed actions no longer automatically bypasses selection. Chunk substitutions therefore include low-progress handling as well as collision avoidance; a substitution count is not a count of distinct avoidance maneuvers.

Interpreting failure. An empty feasible candidate set does not establish that no feasible action exists. Passing the finite-horizon test also does not preclude a later collision, an unobserved neighbour, a forecast error, or overlap between samples. The measured outcome is the full-rollout sampled collision rate under the specified population and dynamics model. Exact tangency can pass $c_t(A) \geq 0$ in Equation 14 but counts as a collision under $d_{ej} \leq 0$.

G FOUR-REGION BEHAVIOR-CLONING AND CONTROLLER RESULTS

The four regions contain 7,600, 7,068, 4,219, and 2,454 test episodes, respectively. Tables 11 and 12 give individual-seed FM-BC and PAR results. The FM-BC seed table uses the checkpoint-sweep evaluation of Tables 3 and 4.

Region	Seed	Step	R_G	C	gc-ADE
AEW	0	1920000	0.8880	0.2007	23.19
AEW	1	1930000	0.8962	0.2076	23.25
AEW	2	2000000	0.8991	0.2025	23.09
AESPA	0	1940000	0.9317	0.4502	26.83
AESPA	1	1920000	0.9310	0.4529	26.77
AESPA	2	2000000	0.9314	0.4501	26.69
AEPBC	0	1990000	0.8978	0.2764	24.21
AEPBC	1	2000000	0.8912	0.2773	24.05
AEPBC	2	1930000	0.9040	0.2799	24.22
AWW	0	1930000	0.7632	0.2514	27.16
AWW	1	1970000	0.7861	0.2608	27.30
AWW	2	1910000	0.7828	0.2494	27.35

Table 11: FM-BC results at the selected checkpoint for each seed. These are the FM-BC evaluations of Tables 3 and 4.

H FOUR-REGION LEARNING COMPARISONS

Tables 13 and 14 report every learned baseline in all four regions, including HR-PPO and PPO beyond AEW. Their FM-BC rows use the checkpoint-sweep evaluation, as in the main anchorage tables. Table 14 also reports DEAS-BoN in every region. Table 15 specifies selected checkpoints. All methods have three policy seeds per region; critic selection shares one critic across those seeds.

Region	Seed	R_G	C	Subst./ep
AEW	0	0.8925	0.0114	4.32
AEW	1	0.8962	0.0117	3.57
AEW	2	0.8978	0.0104	3.57
AESPA	0	0.9250	0.0352	4.93
AESPA	1	0.9220	0.0349	4.50
AESPA	2	0.9305	0.0307	4.42
AEPBC	0	0.9068	0.0194	6.18
AEPBC	1	0.9033	0.0175	6.17
AEPBC	2	0.9073	0.0213	6.90
AWW	0	0.8073	0.0257	7.49
AWW	1	0.8187	0.0183	6.81
AWW	2	0.8191	0.0240	6.16

Table 12: PAR results by seed. Substitutions count replaced action chunks, including near-goal low-progress handling, rather than distinct avoidance maneuvers.

PPO's instability is visible within regions as well as between them. On AEW its three seeds have punctuality approximately 0.883, 0.120, and 0.012. On AWW, high punctuality coexists with a gc-ADE of 293.18. Including these runs avoids concealing the behavioral cost and seed variation of the unregularized objective. ReinFlow and HR-PPO also differ from the offline methods in interaction budgets, scenario eligibility, and execution frequency, as described in Appendix E.

Trajectory-prediction baselines in all regions. Table 16 reports both one-shot forecasters in every anchorage region. GLRP reaches the goal punctually in 47.1–59.6% of episodes and TrAISformer in 14.0–22.6%, with collision rates of 24.4–51.3% and 42.1–76.4%, respectively, below the punctuality and above the collision rate of FM-BC in every region.

Region	Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA (m)	gc-ADE $\downarrow$	FDE$_{\neg R}$ (m)
AEW	FM-BC	0.8944 ± 0.0057	0.2036 ± 0.0036	87.8	23.18 ± 0.08	436
AEW	HR-PPO	0.8796 ± 0.0094	0.2200 ± 0.0080	80.0	30.25 ± 0.86	453
AEW	QC-FQL	0.9199 ± 0.0013	0.2025 ± 0.0019	87.3	23.57 ± 0.15	455
AESPA	FM-BC	0.9313 ± 0.0004	0.4510 ± 0.0016	8.5	26.76 ± 0.07	533
AESPA	HR-PPO	0.9466 ± 0.0015	0.4576 ± 0.0041	7.8	32.78 ± 0.12	752
AESPA	QC-FQL	0.9616 ± 0.0014	0.4461 ± 0.0020	9.4	27.85 ± 0.09	684
AEPBC	FM-BC	0.8977 ± 0.0064	0.2779 ± 0.0018	56.0	24.16 ± 0.10	503
AEPBC	HR-PPO	0.9049 ± 0.0049	0.3070 ± 0.0097	46.4	30.56 ± 0.50	590
AEPBC	QC-FQL	0.9428 ± 0.0021	0.2746 ± 0.0022	57.9	24.62 ± 0.05	644
AWW	FM-BC	0.7774 ± 0.0123	0.2539 ± 0.0061	79.9	27.27 ± 0.10	417
AWW	HR-PPO	0.8370 ± 0.0371	0.2680 ± 0.0120	73.1	32.04 ± 0.51	381
AWW	QC-FQL	0.8583 ± 0.0031	0.2399 ± 0.0033	84.3	26.39 ± 0.18	404

Table 13: Learning baselines on the four-region test population. Rates and gc-ADE are mean $\pm$ sample standard deviation across three seeds; CPA and FDE$_{\neg R}$ are means of per-seed summaries. FDE$_{\neg R}$ conditions on missing the punctual deadline. Checkpoints are test-selected. HR-PPO uses KL weight 0.1. Expert replay has $R_G = 1$ by construction; its regional CPA is 119.8, 76.0, 106.0, and 119.0 m. Further baselines appear in Appendix H.

Region	Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA (m)	gc-ADE $\downarrow$	FDE$_{\neg R}$ (m)
AEW	NF-BC	0.7545 ± 0.0003	0.2231 ± 0.0028	80.6	26.54 ± 0.13	490
AEW	ReinFlow	0.9246 ± 0.0059	0.2132 ± 0.0095	84.3	25.48 ± 0.62	439
AEW	PPO	0.3382 ± 0.4746	0.4329 ± 0.1714	21.1	242.01 ± 56.28	3640
AEW	DEAS-BoN	0.9665 ± 0.0015	0.2054 ± 0.0020	87.2	31.77 ± 0.09	375
AESPA	NF-BC	0.7961 ± 0.0036	0.4897 ± 0.0041	1.8	29.52 ± 0.02	501
AESPA	ReinFlow	0.9482 ± 0.0201	0.4638 ± 0.0228	6.5	30.00 ± 0.34	540
AESPA	PPO	0.0075 ± 0.0034	0.4883 ± 0.1432	10.4	185.71 ± 13.65	3358
AESPA	DEAS-BoN	0.9768 ± 0.0016	0.4505 ± 0.0041	8.4	36.60 ± 0.21	465
AEPBC	NF-BC	0.7621 ± 0.0031	0.3163 ± 0.0106	45.1	26.92 ± 0.23	491
AEPBC	ReinFlow	0.9256 ± 0.0182	0.2845 ± 0.0078	55.5	26.30 ± 0.58	530
AEPBC	PPO	0.6672 ± 0.3032	0.3362 ± 0.0546	26.8	246.84 ± 59.39	2076
AEPBC	DEAS-BoN	0.9534 ± 0.0021	0.2780 ± 0.0047	56.2	28.90 ± 0.06	450
AWW	NF-BC	0.6989 ± 0.0014	0.2569 ± 0.0059	76.7	27.56 ± 0.06	399
AWW	ReinFlow	0.8219 ± 0.0145	0.2547 ± 0.0133	80.6	28.40 ± 0.03	416
AWW	PPO	0.9849 ± 0.0195	0.2691 ± 0.0349	45.0	293.18 ± 26.13	2993
AWW	DEAS-BoN	0.8350 ± 0.0070	0.2498 ± 0.0011	79.2	28.47 ± 0.21	380

Table 14: Additional four-region learning baselines, using the same aggregation and metric definitions as Table 13. PPO has no teacher penalty. DEAS-BoN uses $N = 16$ candidates, a batch-64 plan, and one regional critic shared across policy seeds. Each row includes all three seeds, including poorly performing runs.

Region	Method	Seed 0	Seed 1	Seed 2
AEW	FM-BC	1920000	1930000	2000000
AEW	NF-BC	1940000	1980000	1950000
AEW	ReinFlow	1520	304	1520
AEW	HR-PPO	760	762	762
AEW	PPO	762	760	760
AEW	QC-FQL	100000	200000	300000
AESPA	FM-BC	1940000	1920000	2000000
AESPA	NF-BC	1910000	1910000	1930000
AESPA	ReinFlow	1520	1525	304
AESPA	HR-PPO	762	762	760
AESPA	PPO	152	532	380
AESPA	QC-FQL	800000	1000000	300000
AEPBC	FM-BC	1990000	2000000	1930000
AEPBC	NF-BC	1970000	1980000	1930000
AEPBC	ReinFlow	304	1520	1520
AEPBC	HR-PPO	760	760	762
AEPBC	PPO	760	760	762
AEPBC	QC-FQL	100000	200000	400000
AWW	FM-BC	1930000	1970000	1910000
AWW	NF-BC	1920000	1930000	1930000
AWW	ReinFlow	1525	304	1368
AWW	HR-PPO	760	762	762
AWW	PPO	762	684	762
AWW	QC-FQL	100000	100000	200000

Table 15: Selected checkpoints. FM-BC, NF-BC, and QC-FQL values are gradient updates; ReinFlow, HR-PPO, and PPO values are rollout/update iterations. Each checkpoint is selected using test R_G, with gc-ADE breaking ties. Controller and critic evaluations use the selected FM-BC checkpoint of the same seed.

Region	Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA (m)	gc-ADE $\downarrow$	FDE$_{\neg R}$ (m)
AEW	GLRP (one-shot)	0.5315 ± 0.0048	0.2439 ± 0.0024	70.8	16.84 ± 0.26	303
AEW	TrAISformer (one-shot)	0.2261 ± 0.0125	0.5075 ± 0.0069	-1.5	38.54 ± 1.00	867
AESPA	GLRP (one-shot)	0.5961 ± 0.0647	0.5134 ± 0.0089	-2.2	22.35 ± 0.61	369
AESPA	TrAISformer (one-shot)	0.1401 ± 0.0070	0.7644 ± 0.0091	-15.7	58.92 ± 1.20	1430
AEPBC	GLRP (one-shot)	0.4710 ± 0.0278	0.3344 ± 0.0017	37.8	20.55 ± 0.21	344
AEPBC	TrAISformer (one-shot)	0.1655 ± 0.0110	0.5134 ± 0.0002	-1.9	47.78 ± 0.70	1046
AWW	GLRP (one-shot)	0.5662 ± 0.0306	0.3010 ± 0.0103	59.8	20.35 ± 0.54	287
AWW	TrAISformer (one-shot)	0.2009 ± 0.0170	0.4211 ± 0.0049	18.4	52.09 ± 0.57	1175

Table 16: One-shot trajectory-prediction baselines in all four anchorage regions, each queried once at the first step and executed open-loop; three seeds each. Aggregation and metrics follow Table 13.

I CONTROLLER ENVELOPE AND CANDIDATE-FILTERING STUDIES

PAR versus the earlier candidate family. Table 17 compares the reported PAR controller with the earlier E0 family on all four regions. PAR increases residual collision rates relative to E0 but lowers the mean per-episode maximum of the legacy curvature statistic. PAR substitutes 3.8–6.8 chunks per episode, compared with E0's 0.6–1.3. These substitutions include the low-progress rule near the goal. In the seed-0 AEW diagnostic, approximately 84% of PAR substitutions arise from that rule, so the increase cannot be interpreted as an increase in distinct avoidance maneuvers.

Region	Method	R_G	C	max $\tilde{\kappa}$
AEW	FM-BC	0.8944	0.2036	0.0055
AEW	+ E0	0.9054	0.0026	0.0502
AEW	+ PAR	0.8955	0.0112	0.0205
AESPA	FM-BC	0.9313	0.4510	0.0150
AESPA	+ E0	0.9439	0.0086	0.0854
AESPA	+ PAR	0.9259	0.0336	0.0342
AEPBC	FM-BC	0.8977	0.2779	0.0135
AEPBC	+ E0	0.9223	0.0061	0.0913
AEPBC	+ PAR	0.9058	0.0194	0.0316
AWW	FM-BC	0.7774	0.2539	0.0229
AWW	+ E0	0.8499	0.0042	0.1639
AWW	+ PAR	0.8150	0.0227	0.0365

Table 17: The earlier E0 candidate family and PAR (configuration C3), three-seed means. E0 retains the simulator's displacement and heading bounds but has no additional rate envelope. The curvature column is the legacy heading-based statistic defined in Appendix K; it is not curvature in rad/m.

Sensitivity to rate limits. The seed-0 study in Table 18 varies heading limits, speed ramps, turn-spread schedules, and forecast horizon. Stricter settings generally leave more residual collisions, although the ordering is not monotonic and several factors change together. For example, AEW collision rates are 0.0072 at C2, 0.0114 at C3, and 0.0246 at C5. Configuration C3, used for PAR, was chosen from this test-set study before the remaining seeds and regions were evaluated. This selection is disclosed rather than treating the subsequent table as evaluation of an independently fixed hyperparameter.

The stored executed-action counters for the AEW/AESPA seed-0 PAR study report no violations beyond numerical tolerance on replacement steps. Whole-rollout counters also include unmodified nominal actions and can exceed the PAR limits. The envelope settings are configurable rate constraints; assigning them merchant-vessel class names would require vessel-specific validation that is absent here. All envelope results used here come from the corrected speed-ramp implementation.

Filtering policy samples. A complementary seed-0 study first removes policy samples that fail the capsule forecast, then ranks feasible samples by the standard-reward critic; if none passes, it uses the highest-value sample. Tables 19 and 20 compare 16, 64, and 256 samples on the same batch-32 evaluation plan, alongside FM-BC (seed 0) in each region. The samples are drawn from the frozen policy and do not receive PAR's replacement-action limits.

At matched sample counts, filtering reduces collision rates by 10.6–27.2%, while punctuality changes by at most 0.12 percentage points. The relative reduction grows from 10.6–13.6% at 16 samples to 23.4–27.2% at 256 samples. Using 256 instead of 16 unfiltered samples increases both punctuality and gc-ADE in each region. Filtering slightly lowers that tracking error at every matched count. These results separate the effect of geometric screening from the effects of drawing more policy proposals.

Two controls examine selection and fallback choices. At 64 samples, choosing the greatest predicted minimum clearance when no sample passes gives lower collision rates than the default value-based fallback in every region. A second control samples uniformly from feasible proposals without a critic, or uniformly from all 64 when none passes. It reduces collision rates by 16.3–21.5% relative to

Setting	$\lvert\Delta\psi\rvert$ (°)	$\lvert a\rvert$ (m/s^2)	Spread	Horizon	AEW C	AEW R_G	AESPA C	AESPA R_G
E0	—	—	1	16	0.0028	0.9014	0.0083	0.9427
E1	17	0.150	1	16	0.0047	0.9016	0.0144	0.9403
E2	10	0.050	4	16	0.0034	0.8979	0.0170	0.9288
E3	10	0.050	4	32	0.0041	0.8959	0.0151	0.9290
E4	17	0.150	1	24	0.0054	0.9029	0.0117	0.9393
E5	20	0.100	2	16	0.0037	0.9001	0.0112	0.9393
C1	5	0.030	4	16	0.0082	0.8976	0.0235	0.9336
C2	5	0.015	6	16	0.0072	0.8989	0.0235	0.9329
C3	3	0.010	8	16	0.0114	0.8925	0.0352	0.9250
C4	2	0.005	8	16	0.0229	0.8891	0.0565	0.9138
C5	2	0.000	8	16	0.0246	0.8875	0.0603	0.9085

Table 18: Envelope study on FM-BC seed 0, with all settings for a region evaluated on one GPU after the speed-ramp correction. PAR uses configuration C3. Spread is the schedule length used to construct a turn; only four actions are committed before replanning. E5 retains combinations of bounded turns and slowdowns; other rate-limited settings disable expansion. Settings change multiple candidate-construction choices and are not a single-factor sweep. Their names are configuration identifiers, not validated vessel classes.

FM-BC (seed 0), with punctuality changes between -0.16 and $+0.34$ percentage points. Geometric screening thus yields a collision reduction even without value ranking in this study. Substantial residual collision rates remain; these findings concern the evaluated candidate pools and forecasts rather than the capacity of all policies or sampling schemes.

Region	Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA (m)	gc-ADE $\downarrow$	FDE$_{\neg R}$ (m)
AEW	FM-BC	0.8880	0.2007	88.3	23.19	429
AEW	DEAS-BoN, $N = 16$	0.9670	0.2078	86.7	31.76	377
AEW	+ filter	0.9663	0.1795	86.9	31.50	378
AEW	DEAS-BoN, $N = 64$	0.9741	0.2067	84.5	38.46	312
AEW	+ filter	0.9746	0.1687	84.7	37.98	319
AEW	DEAS-BoN, $N = 256$	0.9746	0.2126	83.6	47.02	268
AEW	+ filter	0.9738	0.1549	83.8	46.15	277
AEW	Filter 64, least violation	0.9742	0.1537	84.7	37.41	320
AEW	Random feasible, 64	0.8914	0.1576	87.6	23.10	434
AESPA	FM-BC	0.9317	0.4502	8.7	26.83	552
AESPA	DEAS-BoN, $N = 16$	0.9775	0.4498	8.1	37.00	457
AESPA	+ filter	0.9779	0.4012	10.6	36.44	478
AESPA	DEAS-BoN, $N = 64$	0.9803	0.4537	8.1	44.29	369
AESPA	+ filter	0.9803	0.3779	11.3	43.31	381
AESPA	DEAS-BoN, $N = 256$	0.9781	0.4614	6.8	52.70	297
AESPA	+ filter	0.9779	0.3441	11.0	51.09	304
AESPA	Filter 64, least violation	0.9803	0.3476	11.7	41.85	415
AESPA	Random feasible, 64	0.9301	0.3769	12.4	26.82	547

Table 19: Candidate filtering on AEW and AESPA, seed 0. The DEAS-BoN, filter, and control arms use batch 32; the FM-BC row is the seed-0 evaluation of Table 11. Each + filter row uses the same sample count as the DEAS-BoN row above it. All methods share the regional FM-BC checkpoint; DEAS-BoN arms share the standard-reward critic. Rates and errors summarize the full regional test population, and CPA is the median episode minimum clearance. FDE$_{\neg R}$ conditions on missing the punctual deadline. Least violation changes the fallback when no sample passes; random feasible removes value ranking. These samples do not receive PAR's replacement-action limits.

Region	Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA (m)	gc-ADE $\downarrow$	FDE$_{\neg R}$ (m)
AEPBC	FM-BC	0.8978	0.2764	56.7	24.21	502
AEPBC	DEAS-BoN, $N = 16$	0.9562	0.2747	57.3	28.94	458
AEPBC	+ filter	0.9559	0.2456	57.2	28.77	459
AEPBC	DEAS-BoN, $N = 64$	0.9647	0.2830	53.6	31.98	410
AEPBC	+ filter	0.9635	0.2280	53.4	31.65	402
AEPBC	DEAS-BoN, $N = 256$	0.9708	0.2868	52.5	35.22	383
AEPBC	+ filter	0.9713	0.2088	52.9	34.76	383
AEPBC	Filter 64, least violation	0.9602	0.2074	53.5	31.16	401
AEPBC	Random feasible, 64	0.8962	0.2292	55.0	24.18	503
AWW	FM-BC	0.7632	0.2514	80.8	27.16	401
AWW	DEAS-BoN, $N = 16$	0.8264	0.2539	79.3	28.23	360
AWW	+ filter	0.8268	0.2221	79.4	28.20	361
AWW	DEAS-BoN, $N = 64$	0.8488	0.2498	81.6	29.08	359
AWW	+ filter	0.8480	0.2115	81.8	28.98	361
AWW	DEAS-BoN, $N = 256$	0.8606	0.2559	80.6	30.49	327
AWW	+ filter	0.8598	0.1960	81.1	30.37	330
AWW	Filter 64, least violation	0.8476	0.2025	81.7	28.93	366
AWW	Random feasible, 64	0.7637	0.2033	78.8	27.31	402

Table 20: Candidate filtering on AEPBC and AWW, with the same seed-0 protocol, metric definitions, and arm order as Table 19. Each filtered DEAS-BoN arm is compared with its corresponding unfiltered sample count.

J ARRIVAL TIME AND CONDITIONAL FINAL DISPLACEMENT

Aligning the arrival event. Let τ_g^{expert} be the human replay's first entry into the same 100 m goal region used for the policy, on the same episode. We report

$$\Delta_{\text{arrival}} = 100\frac{\tau_g - \tau_g^{\text{expert}}}{\tau_g^{\text{expert}}}. \tag{17}$$

Negative values indicate earlier policy arrival and zero denotes matching human arrival. Comparing τ_g with the reference length T instead measures a different event: the human can enter the goal before its final logged sample. The definition-matched quantity removes that systematic offset.

Table 21 averages each seed's summaries, while Figure 7 displays the saved distributions pooled across seeds. Each method includes its own episodes that reach the goal within the extended window $T+E$ (Appendix C.4). Non-reaching episodes have unobserved arrival times and are excluded, rather than assigned their stopping time as an arrival. These are human-matched comparisons, not policy-versus-controller comparisons restricted to their common reaching subset. The reported median changes of at most 0.7 percentage points therefore describe the reaching populations and are not an individual-level causal timing effect.

Region	Method	Reaching n	Mean (%)	Median (%)	p_{10} (%)	p_{90} (%)	Earlier
AEW	FM-BC	7554	−4.01	−5.17	−25.33	16.68	0.610
AEW	+ PAR	7575	−3.96	−4.85	−25.19	17.11	0.604
AESPA	FM-BC	7021	−6.85	−8.56	−31.39	17.20	0.671
AESPA	+ PAR	7037	−6.27	−7.96	−31.17	18.75	0.658
AEPBC	FM-BC	4170	−3.95	−6.00	−33.61	23.85	0.606
AEPBC	+ PAR	4202	−5.07	−6.22	−33.99	22.91	0.614
AWW	FM-BC	2337	3.83	−1.99	−35.38	45.81	0.495
AWW	+ PAR	2408	1.09	−2.68	−36.44	41.41	0.518

Table 21: Arrival time relative to human replay on the same episode: $100(\tau_g - \tau_g^{\text{expert}})/\tau_g^{\text{expert}}$. Each cell is the mean of three per-seed summaries; counts are rounded. Non-reaching episodes are excluded separately for each method, so the populations differ. This pairs each policy with the human reference, not with the other policy on a common subset.

What the conditional FDE means. The main-table $\text{FDE}_{\neg R}$ is computed from the stored reference-window FDE on episodes with $R_G = 0$. It differs from conditioning on failure within $T + E$, and from measuring goal distance at the final executed sample. From FM-BC to FM-BC + PAR (Tables 3 and 4) it rises from 436 to 470 m on AEW, 533 to 571 m on AESPA, 503 to 565 m on AEPBC, and 417 to 486 m on AWW. Because the methods change which episodes miss the deadline, this increase alone does not establish deterioration on the same failures.

For DEAS-BoN with 16 candidates, conditional FDE is 375, 465, 450, and 380 m on AEW, AESPA, AEPBC, and AWW, respectively (Tables 3 and 14). These values are below those of FM-BC (436, 533, 503, and 417 m; Table 13), but likewise average each method's own missed-deadline episodes. The filtering tables use the same conditional definition. Comparing the error on a shared set of failures would answer a different question and is not available from these aggregate summaries.

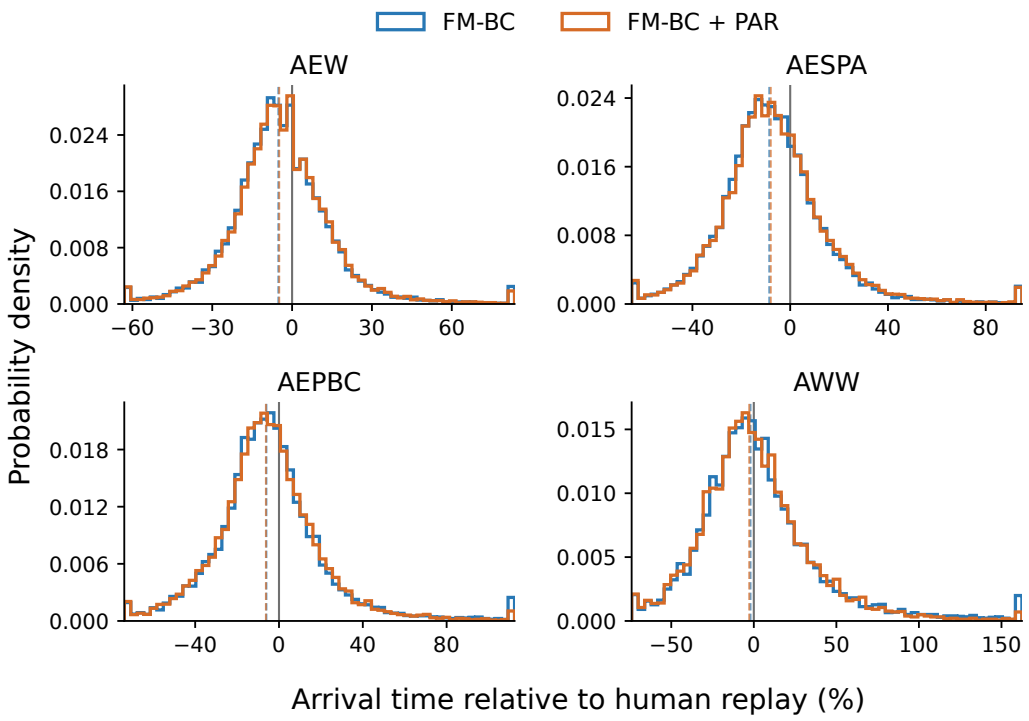


Figure 7: Arrival-time distributions drawn from saved timing records, pooled across three policy seeds. Each arm includes its own episodes reaching the goal within $T + E$; negative values mean earlier arrival than human replay. The grey line marks the human reference and dashed lines mark policy medians. Display tails are clipped at pooled 0.5th/99.5th-percentile bounds; Table 21 summarizes the untrimmed values.

K KINEMATIC METRICS AND THEIR INTERPRETATION

The saved kinematic statistics follow the ShipNaviSim implementation (Pham et al., 2025). For displacement $d_t = p_t - p_{t-1}$, speed is $v_t = \|d_t\|/\Delta t$, acceleration is $(v_t - v_{t-1})/\Delta t$, and drift is the wrapped angle between $\mathrm{atan2}(d_{t,y}, d_{t,x})$ and the recorded heading ψ_t. The first speed difference uses the observed initial speed. The implementation computes these extrema over the executed portion of the reference window; the extended collision-evaluation window uses a different mask.

The legacy parametric expression uses a per-step heading increment where a heading rate would normally occur. Above its $v_t^2 \geq 10^{-3}$ threshold it simplifies to

$$\tilde{\kappa}_t = \frac{\mathrm{wrap}(\psi_t - \psi_{t-1})}{v_t}, \qquad \kappa_{\psi,t} = \frac{\tilde{\kappa}_t}{\Delta t} = \frac{\mathrm{wrap}(\psi_t - \psi_{t-1})}{\|d_t\|}, \tag{18}$$

where $\mathrm{wrap}(\theta) = ((\theta + \pi) \bmod 2\pi) - \pi$ maps an angle difference to $[-\pi, \pi)$. Both are zeroed below that speed threshold. Tables retain $\tilde{\kappa}$ for continuity with the saved results and do not label it rad/m. Dividing by 10 s gives heading change per travelled metre, which is still distinct from curvature derived from successive courses over ground when heading and displacement direction differ.

Tables 22 and 23 average per-episode extrema, then average seeds. PAR raises the mean maximum $\tilde{\kappa}$ relative to FM-BC but remains below human replay in all four regions. Acceleration extrema remain close to FM-BC's because most actions are unchanged; this is compatible with limiting only replacements. AIS jitter, heading discontinuities, and low speeds affect the extrema, so proximity to human extrema alone does not validate maneuverability or establish a cause of collisions.

Region	Method	min $\tilde{\kappa}$	max $\tilde{\kappa}$	min a	max a	min drift	max drift
AEW	Human	−0.0310	0.0266	−0.5584	0.4901	−0.2304	0.4621
AEW	FM-BC	−0.0076	0.0055	−0.1705	0.1526	−0.0520	0.2820
AEW	+ PAR	−0.0136	0.0205	−0.1748	0.1588	−0.0636	0.2564
AESPA	Human	−0.0579	0.0547	−0.5468	0.4752	−0.5720	0.5659
AESPA	FM-BC	−0.0132	0.0150	−0.1592	0.1422	−0.2988	0.2408
AESPA	+ PAR	−0.0281	0.0342	−0.1676	0.1532	−0.2963	0.2291
AEPBC	Human	−0.0517	0.0451	−0.4771	0.4135	−0.4578	0.5285
AEPBC	FM-BC	−0.0138	0.0135	−0.1338	0.1181	−0.2554	0.2952
AEPBC	+ PAR	−0.0250	0.0316	−0.1359	0.1217	−0.2285	0.2539
AWW	Human	−0.0500	0.0433	−0.4595	0.3979	−0.4864	0.5834
AWW	FM-BC	−0.0329	0.0229	−0.1600	0.1420	−0.4495	0.5170
AWW	+ PAR	−0.0349	0.0365	−0.1579	0.1428	−0.3536	0.4009

Table 22: Human replay, FM-BC, and PAR kinematics. Each cell averages per-episode extrema, then averages policy seeds (one replay for the human). Acceleration is in m/s^2 and drift in radians. $\tilde{\kappa}$ is the legacy heading-based statistic; divide by the 10 s timestep to obtain heading change per travelled metre above its speed threshold. These reference-window summaries include unchanged nominal actions.

Table 24 adds the kinematics of DEAS-BoN with 16 candidates and filtered DEAS-BoN with 256 candidates, alongside FM-BC. Both selection schemes increase the magnitudes of mean acceleration and deceleration extrema in every region. Their higher punctuality therefore accompanies changes in executed motion, even though they select proposals from the frozen policy. These summaries describe the resulting rollouts; they do not establish vessel-specific maneuverability.

Region	Method	min $\tilde{\kappa}$	max $\tilde{\kappa}$	min a	max a	min drift	max drift
AEW	FM-BC	−0.0076	0.0055	−0.1705	0.1526	−0.0520	0.2820
AEW	NF-BC	−0.0198	0.0159	−0.3447	0.3031	−0.1615	0.3908
AEW	ReinFlow	−0.0117	0.0088	−0.1641	0.1487	−0.1984	0.4094
AEW	HR-PPO	−0.0136	0.0090	−0.0515	0.0631	−0.1062	0.2922
AEW	PPO	−0.1093	0.1378	−0.2958	0.6312	−2.1146	2.0384
AEW	QC-FQL	−0.0054	0.0041	−0.1620	0.1455	−0.0322	0.2407
AEW	GLRP (one-shot)	−0.4580	0.4638	−0.3939	0.4145	−0.2352	0.2544
AEW	TrAISformer (one-shot)	−0.2433	0.2274	−0.3109	0.3117	−0.2436	0.3660
AESPA	FM-BC	−0.0132	0.0150	−0.1592	0.1422	−0.2988	0.2408
AESPA	NF-BC	−0.0338	0.0318	−0.3270	0.2883	−0.4602	0.4250
AESPA	ReinFlow	−0.0169	0.0202	−0.1624	0.1460	−0.4285	0.3890
AESPA	HR-PPO	−0.0180	0.0118	−0.0413	0.0577	−0.2244	0.2498
AESPA	PPO	−0.1462	0.1094	−0.2111	0.0959	−1.8930	2.1720
AESPA	QC-FQL	−0.0096	0.0128	−0.1559	0.1398	−0.2048	0.1671
AESPA	GLRP (one-shot)	−0.4956	0.4764	−0.4734	0.5027	−0.4450	0.5490
AESPA	TrAISformer (one-shot)	−0.2748	0.2573	−0.3659	0.3655	−0.3366	0.3601
AEPBC	FM-BC	−0.0138	0.0135	−0.1338	0.1181	−0.2554	0.2952
AEPBC	NF-BC	−0.0379	0.0325	−0.3201	0.2787	−0.4401	0.5057
AEPBC	ReinFlow	−0.0179	0.0175	−0.1353	0.1237	−0.3747	0.4224
AEPBC	HR-PPO	−0.0187	0.0134	−0.0457	0.0591	−0.2109	0.2701
AEPBC	PPO	−0.0861	0.1265	−0.2967	0.8809	−1.6912	1.7310
AEPBC	QC-FQL	−0.0073	0.0100	−0.1228	0.1096	−0.1586	0.1840
AEPBC	GLRP (one-shot)	−0.4600	0.4756	−0.4763	0.5121	−0.4061	0.4375
AEPBC	TrAISformer (one-shot)	−0.2384	0.2271	−0.3848	0.3798	−0.3105	0.3385
AWW	FM-BC	−0.0329	0.0229	−0.1600	0.1420	−0.4495	0.5170
AWW	NF-BC	−0.0562	0.0423	−0.4261	0.3729	−0.7420	0.7532
AWW	ReinFlow	−0.0368	0.0256	−0.1648	0.1503	−0.5916	0.6521
AWW	HR-PPO	−0.0230	0.0120	−0.0764	0.0777	−0.3460	0.3877
AWW	PPO	−0.0042	0.0199	−0.2388	1.3943	−1.1691	0.9326
AWW	QC-FQL	−0.0177	0.0160	−0.1228	0.1129	−0.3311	0.3858
AWW	GLRP (one-shot)	−0.4759	0.4844	−0.5297	0.5771	−0.4437	0.5117
AWW	TrAISformer (one-shot)	−0.2724	0.2806	−0.4608	0.4602	−0.6113	0.5994

Table 23: Kinematic summaries for all learning baselines, with the same conventions as Table 22. Three-seed means of per-episode extrema; values near low speed and AIS heading discontinuities can dominate these statistics. They do not establish hydrodynamic realism. Trajectory-prediction rows are separate evaluations, three seeds each, executed open-loop from the first step.

Region	Method	min $\tilde{\kappa}$	max $\tilde{\kappa}$	min a	max a	min drift	max drift
AEW	FM-BC	-0.0076	0.0055	-0.1705	0.1526	-0.0520	0.2820
AEW	DEAS-BoN, $N = 16$	-0.0079	0.0061	-0.3069	0.2728	-0.0568	0.2788
AEW	FM-BC (seed 0)	-0.0075	0.0055	-0.1732	0.1545	-0.0481	0.2846
AEW	Filtered DEAS-BoN, $N = 256$	-0.0146	0.0126	-0.5192	0.4754	-0.1288	0.3518
AESPA	FM-BC	-0.0132	0.0150	-0.1592	0.1422	-0.2988	0.2408
AESPA	DEAS-BoN, $N = 16$	-0.0149	0.0164	-0.2848	0.2488	-0.2768	0.2419
AESPA	FM-BC (seed 0)	-0.0130	0.0147	-0.1576	0.1410	-0.2914	0.2336
AESPA	Filtered DEAS-BoN, $N = 256$	-0.0281	0.0278	-0.4768	0.4315	-0.3733	0.3460
AEPBC	FM-BC	-0.0138	0.0135	-0.1338	0.1181	-0.2554	0.2952
AEPBC	DEAS-BoN, $N = 16$	-0.0142	0.0145	-0.2239	0.1935	-0.2306	0.2883
AEPBC	FM-BC (seed 0)	-0.0135	0.0139	-0.1339	0.1189	-0.2401	0.2913
AEPBC	Filtered DEAS-BoN, $N = 256$	-0.0238	0.0221	-0.4051	0.3517	-0.2882	0.3484
AWW	FM-BC	-0.0329	0.0229	-0.1600	0.1420	-0.4495	0.5170
AWW	DEAS-BoN, $N = 16$	-0.0295	0.0224	-0.2302	0.2049	-0.4358	0.5102
AWW	FM-BC (seed 0)	-0.0351	0.0243	-0.1597	0.1439	-0.4562	0.5323
AWW	Filtered DEAS-BoN, $N = 256$	-0.0324	0.0237	-0.3655	0.3248	-0.4874	0.5634

Table 24: Kinematics of value selection and candidate filtering. The FM-BC and DEAS-BoN rows average three policy seeds; the FM-BC (seed 0) row accompanies filtered DEAS-BoN, which uses seed 0. Each value first averages per-episode extrema. Units and low-speed limitations follow Table 22.

L DESCRIPTIVE COLLISION-CASE ANALYSIS

An AEW seed-0 scan attributes collisions to partner class and motion on all 7,600 test episodes, using PAR rollouts that reproduce the seed-0 PAR evaluation exactly and a separate FM-BC rollout of the same checkpoint. The FM-BC scan contains 2,069 episode–partner collision pairs, compared with 94 under PAR. Of the FM-BC pairs, 1,837 (88.8%) involve partners classified as anchored from their recorded speed. Under PAR, 38 residual pairs involve anchored partners and 56 involve moving partners (Table 25). Exposure is dominated by anchored vessels, so pair shares should be read alongside exposure counts.

The reduction in pair counts is larger for anchored partners than for moving partners. This is consistent with the relevance of stationary obstacles in the anchorage task, but it does not isolate forecast error, candidate limitations, observation truncation, or collision responsibility. A speed threshold also does not imply perfectly constant position or heading throughout a forecast horizon. The main rates are those of Tables 3 and 4.

Partner	Exposed vessel-episodes	FM-BC pairs	PAR pairs
Cargo	80041	785	30
Tanker	164941	1059	55
Service	3321	10	1
Fishing operations	6766	49	0
Passenger	457	4	0
Other	10301	87	5
Unknown	12190	75	3
Moving	45700	232	56
Anchored	232317	1837	38

Table 25: AEW seed-0 collision-case scan, 7,600 episodes. A pair is an episode and partner vessel with at least one sampled collision. Exposure counts a vessel within 1 nautical mile at least once along the FM-BC trajectory. The motion split is a separate partition of the same population. Anchored means maximum recorded episode speed below 0.26 m/s. FM-BC counts come from a separate rollout of the same seed-0 checkpoint.

M REPLAY THROUGHPUT AND COMPUTATIONAL SCOPE

Workload and counting. The replay microbenchmark uses the AEW population of 22,258 training and 7,600 test scenarios. Every JAX repeat consumes all 29,858 scenarios in the same longest-first order. Each scenario is padded to 128 vessel slots, a configurable setting of this benchmark, with a 10 s timestep. The single-control mode supplies recorded actions to the ego and replays its neighbours; all-vessel control supplies recorded actions to every valid vessel. Neither mode samples a learned policy or measures reactive multi-agent behavior.

Let n_{bt} be the number of valid controlled vessels in live world b at simulated step t. We report

$$\mathrm{CASPS} = \frac{\sum_{b,t} n_{bt}}{\Delta T}, \tag{19}$$

where ΔT is measured elapsed stepping time. The single-control numerator is 2,874,927 live ego transitions per full pass; the all-vessel numerator is 253,423,912 valid vessel transitions, including stationary vessels. Invalid slots and worlds that have finished their final episode do not contribute. Multiplying allocated worlds by scan length would overcount this final drain; at 4,096 worlds the saved padding fraction is 3.72%.

Control	Simulator / hardware	Worlds	Median	Minimum	Maximum
Single	ShipNaviSim, 1 process	1,024	207	207	207
Single	ShipNaviSim, up to 32 processes	64	3,153	3,139	3,157
Single	JAXNAVISIM, RTX 3090	4,096	3,105,403	3,103,203	3,108,024
Single	JAXNAVISIM, H100 NVL	4,096	9,215,237	9,199,142	9,219,120
All vessels	JAXNAVISIM, RTX 3090	1,024	53,591,266	53,503,247	53,639,737
All vessels	JAXNAVISIM, H100 NVL	4,096	124,156,972	124,109,573	124,192,629

Table 26: Peak observed replay throughput across the tested world counts, in valid controlled-vessel steps per second. Each entry summarizes three timing repeats at the world count with the highest median. Single control counts the ego only; all-vessel control counts every valid vessel, including stationary ones. GPU measurements omit learned policy inference, reward, and full benchmark metrics; CPU steps include built-in metrics. Termination and sampled populations also differ (Appendix M).

GPU runner and timing. A ragged device-resident pool stores logged trajectories without padding every episode to the longest duration. The runner maintains a rolling 12-state window and replaces a finished scenario at the end of its logged window. It sweeps 1, 4, 16, 64, 256, 1,024, and 4,096 concurrent worlds on an RTX 3090 (24 GB) and H100 NVL (approximately 94 GB). Data loading, assembly, upload, and initial compilation/warm-up occur before the three timed repeats. In-loop history updates and gathering and reseeding completed worlds are inside the timed program. Both GPUs complete the largest tested batch; the 3090's all-vessel rate peaks earlier.

The program calls the logged-action actor, dynamics, observation function, and ego goal-distance check. It computes neither the learned policy nor rewards, the PAR controller, or the full collision and tracking metrics. The observation history configuration also differs from the behavioral experiments. Only the last-time x-coordinate slices of the ego and neighbour observation blocks feed an accumulated checksum. Unconsumed observation features may therefore be pruned by compilation; these timings do not establish the cost of materializing every policy input. The timer waits on the returned live-step counter rather than explicitly on the full returned output tuple. The effect of these choices on a full observation-consuming workload has not been measured. We consequently report a stepping microbenchmark rather than end-to-end policy throughput.

CPU reference and comparison limits. The public ShipNaviSim implementation runs on an AMD EPYC 7702P, either in one process with round-robin environments or in up to 32 processes over 32 allocated logical CPUs. Each worker steps a shuffled shard for a 120-second budget. Numerators sum actual ego steps; the multiprocess denominator is the largest accumulated worker step-loop time. Loading and resets outside those loops are excluded. Each repeat therefore covers a time-budgeted prefix of the population, unlike the GPU's complete pass. A reported 91 scenarios (0.30%) fail the CPU action-bound check; JAX instead projects actions into its bounds.

The CPU step also computes built-in kinematic, tracking, and proximity metrics. It terminates on reaching a 200 m goal radius, whereas the JAX runner records a 100 m goal flag but continues through the logged window. Thus the methods differ in metric work, termination, action handling, and population coverage.

Relation to behavioral experiments. The timing runner leaves deterministic XLA restrictions disabled. The behavioral evaluation campaign uses deterministic options, including disabled GPU autotuning, across RTX 3090, RTX A5000, L40/L40S, and H100 NVL devices. Exact numerical agreement across these devices is not assumed. The candidate-filtering arms share one batch plan because stochastic draws can change with batching. End-to-end FM-BC/PAR throughput, a matched CPU/GPU workload with common metrics and termination, and a total training GPU-hour count remain unmeasured by the evidence reported here.

N Dynamics models and the external-effect term

Equation 2 defines the delta action space used throughout the main results. This appendix states the three further dynamics models the benchmark supports, together with the sea-current term that couples them to an external forcing field. Each model is given by its forward law and its inverse, because the inverse is what extracts expert actions from a recorded track. A model whose inverse cannot reproduce a recording cannot be trained on that recording by behavior cloning.

N.1 Notation

Positions $p_k = (x_k, y_k)$ are in the local metric frame of Equation 2 and headings ψ_k follow the mathematical convention (0 east, counter-clockwise), with step $\Delta t = 10\,\mathrm{s}$ and step index k. We write

$$f(\psi) = (\cos\psi,\ \sin\psi), \qquad n(\psi) = (-\sin\psi,\ \cos\psi), \qquad \Delta p_k = p_{k+1} - p_k, \tag{20}$$

so $f(\psi)$ is the forward (bow) unit vector and $n(\psi)$ the port unit vector, 90° to the left of the bow. The operator $\mathrm{wrap}(\theta) = ((\theta + \pi) \bmod 2\pi) - \pi$ maps an angle to $(-\pi, \pi]$, and $\mathrm{atan2}(y, x)$ is the two-argument arctangent. The admissible action set $\mathcal{A}$ is Equation 3.

Reproducing a recording. Each model below is stated as a pair. The inverse reads two consecutive recorded states and returns the action that should join them; the forward law takes a state and an action and returns the next state. Running the two in sequence answers a simple question: if we extract the action from a recorded step and then execute it in the simulator, do we arrive at the position the vessel actually reached?

When the answer is yes at every step, the model reproduces the recording exactly, and the demonstrations it extracts are consistent with the simulator that will execute them. When the answer is no, the extracted action does not take the vessel where the recording says it went. A BC policy trained on such data inherits that gap before it makes any error of its own, so the gap limits what the policy can achieve. Table 27 reports, for each model, whether it reproduces a recording exactly.

N.2 The external-effect term

Let $U_k = U(p_k, t_k)$ be the surface current, a two-dimensional velocity sampled at the vessel's own position and wall-clock time at the *start* of the step. We take it from the HYCOM global ocean reanalysis (Chassignet et al., 2007), whose surface velocity fields are interpolated in space and time to the query. The drift is decomposed on the vessel's own axes,

$$g_k^f = \Delta t\, c_f \langle U_k, f(\psi_k)\rangle, \qquad g_k^p = \Delta t\, c_p \langle U_k, n(\psi_k)\rangle, \qquad \delta_k = g_k^f f(\psi_k) + g_k^p n(\psi_k), \tag{21}$$

where g_k^f and g_k^p are the displacements in metres contributed along the bow and to port, δ_k is their vector sum, and the dimensionless gains c_f, c_p give the fraction of the water's motion taken up along each axis. Because f and n are orthonormal, the isotropic case collapses to a heading-free expression,

$$c_f = c_p = c \quad \Longrightarrow \quad \delta_k = \Delta t\, c\, U_k. \tag{22}$$

We use $c_f = c_p = 1$, so that the vessel is advected by the surface current. This is a kinematic assertion rather than a fitted hull coefficient. The port gain c_p cannot be fitted from these recordings, because it never enters the extracted action of any model below, so asserting it is preferable to reporting a value the data does not constrain.

The field is queried with a coverage weight. Outside its spatial or temporal domain we set $\delta_k = 0$, which asserts *no push* rather than *no current*. The two are different claims and only the first is supported.

N.3 Querying the current field

The reanalysis supplies eastward and northward surface velocity on a regular longitude–latitude grid at a nominal $1/12^\circ$ spacing, about 9 km at this latitude, with three-hourly output. A query (p_k, t_k) is answered in three steps. The position is converted from the region's local metric frame back to geographic coordinates; the two velocity components are interpolated trilinearly, bilinearly over the four surrounding grid nodes and linearly between the two bracketing snapshots in time; and the result is rotated into the local frame to give U_k in metres per second. Interpolation is applied to the velocity components separately, so speed and direction are both derived quantities rather than interpolated directly.

Each query also returns a coverage weight $w_k \in [0, 1]$, the product of the in-domain indicators for the spatial and temporal axes and the validity of the surrounding nodes, and the drift of Equation 21 is scaled by it. Land and out-of-domain nodes are invalid, so w_k falls to zero near the boundary rather than stepping, and a vessel leaving coverage is not given a discontinuous shove.

Figure 8 places all eight benchmark regions in the field at one instant. It is drawn by sampling `HycomCurrentField` exactly as the dynamics query it, so the plotted field is the one the vessels are advected by rather than a separate rendering of the source product. Over the full extent shown the speed averages 0.317 kn and peaks at 1.651 kn. The four navigation regions lie along the strait, where the through-flow is strongest and most nearly aligned with the traffic lanes, while the four anchorage groups sit in weaker water to the north. The current term therefore matters most in exactly the regions the dynamics study uses.

Figure 9 shows the resulting field over TSS East. The interpolation is what makes the field continuous at vessel scale; it does not add information. The region is roughly 24 km along by 7 km across, so it spans only a few native cells, and the field it supplies is best read as a slowly varying regional set rather than resolved local structure. This is the same reason the port gain c_p is asserted rather than fitted: neither the recordings nor the field constrain circulation at the scale of a single vessel's manoeuvre.

N.4 Delta dynamics with current

The action of Equation 2, inherited from ShipNaviSim (Pham et al., 2025) and advected by δ_k:

$$\text{forward:}\quad p_{k+1} = p_k + (\Delta x_k, \Delta y_k) + \delta_k, \qquad \psi_{k+1} = \mathrm{wrap}(\psi_k + \Delta\psi_k), \tag{23}$$

$$\text{inverse:}\quad (\Delta x_k, \Delta y_k) = \Delta p_k - \delta_k, \qquad \Delta\psi_k = \mathrm{wrap}(\psi_{k+1} - \psi_k). \tag{24}$$

Three action components act against three state targets, so the extracted action always reproduces the recorded step exactly. With $U \equiv 0$ the model is bit-identical to Equation 2, which we verify numerically on every region frame, so the current-free results remain directly comparable to the main text.

N.5 Port-glide dynamics

Forward thrust and turn, $a_k = (D_k, \Delta\psi_k)$, where D_k is the commanded forward displacement in metres along the bow. The delta model of the previous subsection commands displacement on all three degrees of freedom, but a conventional surface ship carries no actuator on the sway axis: propeller and rudder produce surge and yaw, and sway follows from them rather than being commanded, which is what makes such a vessel underactuated in the sense of Do (2010). The two-component models below take that constraint seriously, and they differ only in what they then do with the heading channel. We define port-glide here rather than adopting it from prior work; it is the most

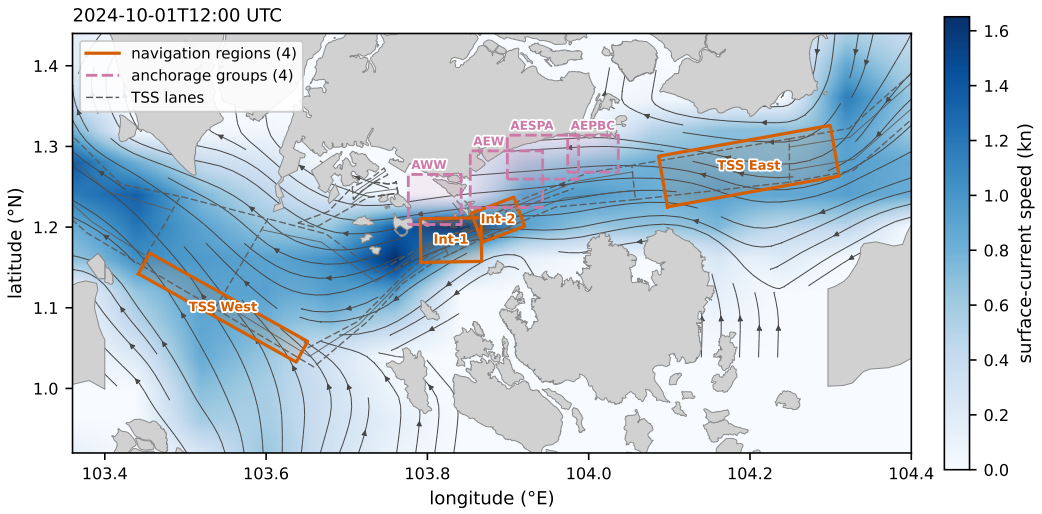


Figure 8: All eight benchmark regions in the Singapore Strait, over the HYCOM surface current at one instant. Shading gives current speed in knots and streamlines give direction, sampled over 103.36–104.40°E and 0.92–1.44°N through the same query path the dynamics use. Solid vermillion outlines mark the four navigation regions (TSS West, TSS East, Int-1, Int-2), drawn from each configuration's `region_wkt` so the rotations are the real ones rather than axis-aligned approximations; dashed purple outlines mark the four anchorage groups (AWW, AEW, AESPA, AEPBC). The two families differ in both hue and line style so they remain distinguishable in grayscale and under colour-vision deficiency. The neutral basemap behind them is the traffic separation scheme: 16 lanes, 21 separation lines, and 4 separation zones, taken from OpenStreetMap `seamark:*` data (© OpenStreetMap contributors, ODbL). The strongest flow follows the strait through the navigation regions, while the anchorage groups lie in weaker water to the north.

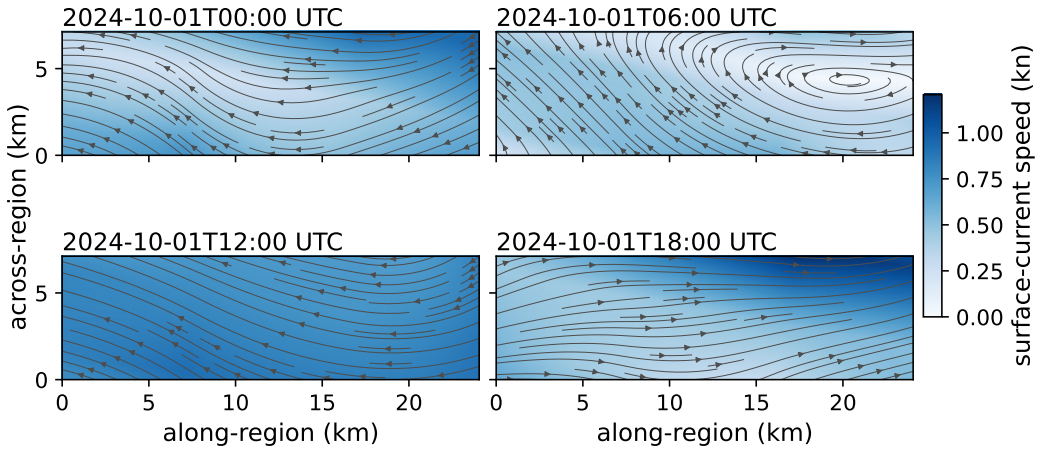


Figure 9: HYCOM surface current over the TSS East region on 1 October 2024, at four times through the day. Shading gives current speed in knots and streamlines give direction, both in the region's local metric frame. The flow reverses with the tide over the day and is not uniform across the region; the hatched wedge marks positions outside the field's coverage, where the drift term is set to zero. Vessel tracks in this region are advected by the field sampled at their own position and time, as described in Appendix N.3.

direct way to drop the sway command while keeping the recorded heading, which is read from the

recording:

$$\text{forward:}\quad \hat{p}_{k+1} = p_k + D_k f(\psi_k) + \delta_k, \qquad \psi_{k+1} = \mathrm{wrap}(\psi_k + \Delta\psi_k), \tag{25}$$

$$\text{inverse:}\quad D_k = \langle \Delta p_k, f(\psi_k) \rangle - g_k^f, \qquad \Delta\psi_k = \mathrm{wrap}(\psi_{k+1} - \psi_k). \tag{26}$$

Two action components act against three state targets, and the model's own output $\hat{p}_{k+1}$ is written separately from the recorded p_{k+1} because here the two differ. Only motion along the bow can be commanded, so any sideways motion in the recording that the current does not account for cannot be reproduced. Executing the extracted action therefore leaves $\hat{p}_{k+1} \neq p_{k+1}$, short of the recording by the sideways distance

$$\left| \langle \Delta p_k, n(\psi_k) \rangle - g_k^p \right|, \tag{27}$$

which is generally not zero. The model is asserting that a vessel cannot be *commanded* sideways, which is correct, and in addition that it cannot *move* sideways, which is not: an underactuated hull still acquires sway through drift, leeway, and the yaw–sway coupling of its own turn (Do, 2010). Conflating the two is a modelling choice rather than a defect, but the residual accumulates over a trajectory, so under this model the expert does not follow its own recorded track.

N.6 Water-track dynamics

The missing sideways motion of Equation 27 disappears if the heading is *defined* as the direction of motion through the water rather than read from the recording. Assuming $c_f = c_p$ so that Equation 22 applies and the drift is the heading-free δ_k, let

$$w_k = \Delta p_k - \delta_k \tag{28}$$

be the through-water displacement, that is, the part of the motion not attributable to the current. The action is again $(D_k, \Delta\psi_k)$, but the model turns before it translates:

$$\text{forward:}\quad \psi_{k+1} = \mathrm{wrap}(\psi_k + \Delta\psi_k), \qquad p_{k+1} = p_k + D_k f(\psi_{k+1}) + \delta_k, \tag{29}$$

$$\text{inverse:}\quad D_k = \|w_k\| \geq 0, \qquad \Delta\psi_k = \mathrm{wrap}\big(\operatorname{atan2}(w_{k,y}, w_{k,x}) - \psi_k\big). \tag{30}$$

Exactness is then an identity rather than an approximation, because $D_k f(\psi_{k+1}) \equiv w_k$ by construction:

$$p_k + D_k f(\psi_{k+1}) + \delta_k = p_k + w_k + \delta_k = p_k + \Delta p_k = p_{k+1}, \tag{31}$$

so any recorded track is reproduced to floating-point precision, whatever sideways motion it contains.

Three consequences follow. First, this is the zero-sideslip assumption: a hull with no leeway points along its own water track, and the angle between heading and course over ground, the drift angle in the sense of Fossen (2011), is then produced by the current rather than asserted. Water-track keeps the same two-component, no-sway-actuator command set as port-glide (Do, 2010), but recovers the missing sideways motion by letting the heading follow the through-water displacement instead of fixing it from the recording, so underactuation costs nothing in reconstruction. Second, ψ_k becomes the course of the step just taken, a post-step quantity, whereas the models above carry a pre-step one. Third, the model rewrites the heading channel of the data and not only the action column, so the observation, the hull orientation used for clearance geometry, and the curvature statistics all follow the redefined heading. Water-track results are therefore comparable to the others on position-based metrics such as goal reaching, but not on clearance metrics, which depend on hull orientation.

Two degenerate cases are handled explicitly. A through-water step shorter than 0.5 m has no defined course and holds the previous heading, and the final step of a window inherits the heading of its predecessor.

N.7 Nomoto dynamics

Water-track with the heading command passed through a first-order lag, in the spirit of the classical first-order steering model of Nomoto et al. (1957); see Fossen (2011) for a modern treatment. The action set remains surge and yaw only (Do, 2010), and the lag is what the two available actuators impose on the yaw channel: a rudder order changes the turn rate over several steps rather than

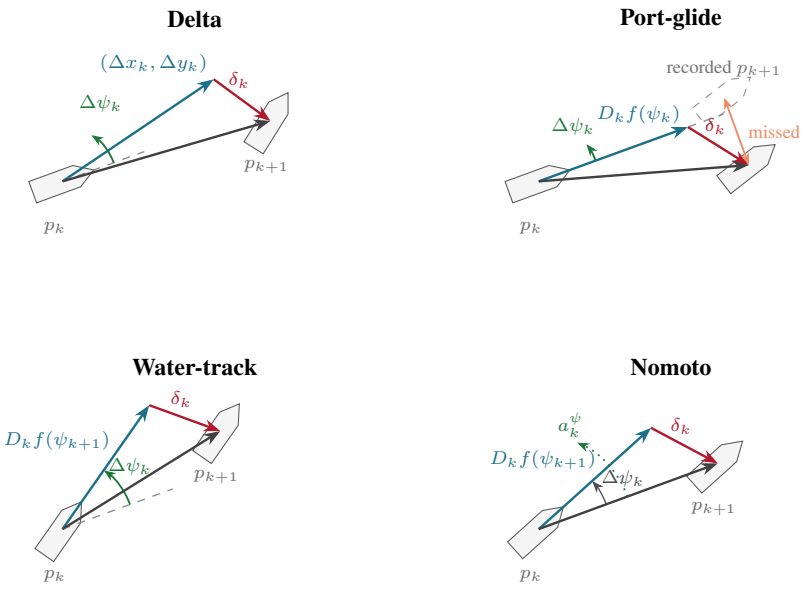


Figure 10: The forward law of each dynamics model, as a vector sum. From the position p_k, the commanded displacement is blue and the commanded turn green, while the drift δ_k contributed by the sea current is red. Their result, grey, is the step over ground the model produces, ending at the solid hull; the dashed ray is the bow $f(\psi_k)$. Delta commands displacement on both axes and turns independently of it. Port-glide instead takes its heading from the AIS recording rather than choosing it, and can only push along that bow. Any sideways motion in the recording is therefore out of reach, so the vessel ends up somewhere other than the recorded p_{k+1}, drawn as a dashed hull, indicated in pale red (Equation 27). The other three reproduce the recorded step $\Delta p_k = p_{k+1} - p_k$ exactly. Water-track turns first and defines the bow as the direction of travel through the water, and Nomoto does the same with a first-order lag, so that its commanded turn a_k^ψ (dotted) and the turn $\Delta\psi_k$ actually applied (grey) differ.

instantly. The position law, D_k, and the drift are unchanged, and only the heading channel differs. With momentum parameter $\alpha \in [0, 1)$ and action $a_k = (D_k, a_k^\psi)$, where a_k^ψ is the commanded turn,

$$\text{forward:} \quad \Delta\psi_k = \alpha\, \Delta\psi_{k-1} + (1 - \alpha)\, a_k^\psi, \tag{32}$$

$$\text{inverse:} \quad a_k^\psi = \frac{\Delta\psi_k - \alpha\, \Delta\psi_{k-1}}{1 - \alpha}, \tag{33}$$

with $\Delta\psi_{-1} = 0$, and with $\Delta\psi_k$ obtained from the water-track inverse above. At $\alpha = 0$ the model reduces exactly to water-track. For $\alpha > 0$ a held command settles onto a constant turn rate, which is the defining behavior of a first-order steering response. Recordings are reproduced exactly for every α in range. The inverse needs only the previous heading increment $\Delta\psi_{k-1}$, so no additional state variable is carried.

We use this discrete lag rather than a discretization of the continuous first-order equation $T\dot{r} + r = K\delta^{\text{rud}}$ of Nomoto et al. (1957), in which r is the yaw rate, δ^{rud} the rudder angle, T a time constant and K a gain. That equation carries the instantaneous yaw rate, which two headings cannot recover, since $(\psi_k - \psi_{k-1})/\Delta t$ is the average rate over the step and the two differ substantially during transients. More importantly, K and T are not identifiable from these recordings, because the rudder is never observed and every parameter pair reproduces the heading sequence exactly. There is consequently no true time constant that the discrete form fails to honor. The parameter α plays the role of $e^{-\Delta t/T}$ and is swept rather than fitted.

The inverse divides by $1 - \alpha$, so the admissible range of a^ψ must exceed the turn limit of $\mathcal{A}$ by a factor $(1+\alpha)/(1-\alpha)$. Retaining the bound of Equation 3 unchanged clips the extracted action and silently destroys the reconstruction.

N.8 SUMMARY

Model	Action	Dim.	Heading	Reproduces a recording
Delta	$(\Delta x, \Delta y, \Delta\psi)$	3	taken from the recording, controlled independently	exactly
Port-glide	$(D, \Delta\psi)$	2	taken from the recording	no, misses sideways motion
Water-track	$(D, \Delta\psi)$	2	estimated from the movement	exactly
Nomoto	(D, a^{ψ})	2	estimated from the movement, then filtered	exactly

Table 27: The four dynamics models. The last column asks whether executing the extracted action lands the vessel where the recording says it went. Delta achieves this with a third degree of freedom; water-track achieves it instead by assuming the vessel cannot slip sideways. Port-glide cannot, and the motion it fails to reproduce limits what a BC policy trained on it can achieve.

These models support two comparisons: whether a vessel's heading is an independent control or is determined by where the vessel is going, and what an external forcing field costs a policy under either assumption.

N.9 DYNAMICS MODELS AND THE EXTERNAL EFFECT

We compare the dynamics model and an unmodelled current in the anchorage regions AEW and AESPA with FM-BC under the anchorage protocol of Section 5.2, scored with the metrics of Section 3.4 on the full test populations of 7,600 and 7,068 episodes. The current comparison also carries the TSS East rows, so Table 29 reports both settings together. Each row is a separately trained FM-BC policy whose demonstrations were extracted through, and whose actions are integrated by, the named model of Appendix N; its checkpoint is the best of the last ten by R_G, with ties broken by gc-ADE, as in Appendix E. The forward displacement of the two-component models is bounded at 300 m on AEW and 280 m on AESPA, and the current is not observed by the policy.

Region	Dynamics model	$R_G \uparrow$	Collision rate $C \downarrow$	CPA (m)	gc-ADE $\downarrow$	FDE$_{\neg R}$ (m)	Replay R_G	Replay C
AEW	Delta	0.8880	0.2007	88.3	23.19	429	1.0000	0.0000
AEW	Port-glide	0.4433	0.2849	61.6	42.79	945	0.3986	0.1993
AEW	Water-track	0.8817	0.2332	64.7	25.32	474	1.0000	0.0171
AEW	Nomoto	0.8782	0.2371	64.9	25.54	463	1.0000	0.0170
AESPA	Delta	0.9317	0.4502	8.7	26.83	552	1.0000	0.0000
AESPA	Port-glide	0.2591	0.5074	−1.0	44.65	809	0.2292	0.4428
AESPA	Water-track	0.9380	0.4690	5.8	28.31	693	1.0000	0.0204
AESPA	Nomoto	0.9345	0.4713	5.1	28.78	711	1.0000	0.0199

Table 28: Dynamics models with no current. The Delta rows correspond to Table 11. FDE$_{\neg R}$ is as in Section 3.4. The replay columns replay the human tracks open-loop through the same model; they are a reference, not an upper bound.

Without a current, Delta, Water-track and Nomoto reach similar R_G in both regions (0.878–0.888 on AEW and 0.932–0.938 on AESPA), while Port-glide drops to 0.443 on AEW and 0.259 on AESPA (Table 28). Human replay through Port-glide reaches only 0.399 and 0.229, because the model cannot reproduce the recording's sideways motion (Equation 27); the Port-glide policy exceeds its replay, so the replay is a reference rather than an upper bound. Under the current, training with it raises R_G by $+0.077$/$+0.124$ for Water-track, $+0.055$/$+0.123$ for Nomoto, $+0.012$/$+0.074$ for Delta and $+0.001$/$+0.053$ for Port-glide on AEW/AESPA (Table 29), so the recovery depends on the model and the region. TSS East, in the same table, moves far less: $+0.090$ for Nomoto, $+0.029$ for Port-glide, $+0.010$ for Delta and -0.014 for Water-track. The anchorages are the harder case for an unmodelled current, since a vessel manoeuvring slowly among moored hulls accumulates drift over a long approach, whereas a transiting vessel crosses the field quickly. Three caveats apply. Water-track and Nomoto orient the hull by the through-water heading (Section 3.5), so their C and CPA are not directly comparable with the Delta rows: replay through them already has $C = 0.017$

		Trained without current					Trained with current				
Region	Dynamics model	$R_G \uparrow$	$C \downarrow$	CPA (m)	gc-ADE ↓	$\text{FDE}_{\neg R}$ (m)	$R_G \uparrow$	$C \downarrow$	CPA (m)	gc-ADE ↓	$\text{FDE}_{\neg R}$ (m)
AEW	Delta	0.6563	0.3149	49.6	32.62	525	0.6682	0.3175	48.9	31.31	495
AEW	Port-glide	0.3639	0.3471	41.8	48.88	969	0.3647	0.3325	46.3	47.23	914
AEW	Water-track	0.7300	0.2337	70.2	33.43	518	0.8067	0.2180	74.5	27.66	475
AEW	Nomoto	0.7539	0.2351	70.4	33.83	505	0.8092	0.2175	74.0	27.56	471
AESPA	Delta	0.6142	0.5545	−8.2	35.32	508	0.6882	0.5447	−6.2	34.02	516
AESPA	Port-glide	0.2204	0.5904	−11.0	50.94	977	0.2732	0.5576	−8.0	50.48	993
AESPA	Water-track	0.7600	0.4707	5.0	33.85	519	0.8838	0.4431	8.6	30.10	565
AESPA	Nomoto	0.7586	0.4641	5.9	34.01	518	0.8820	0.4440	8.9	30.45	583
TSS-East	Delta	0.9108	0.1423	262.6	119.06	3712	0.9203	0.1332	285.1	114.45	3562
TSS-East	Port-glide	0.3758	0.1255	290.8	111.57	3500	0.4052	0.1251	295.2	106.90	3368
TSS-East	Water-track	0.9313	0.1192	307.9	107.51	3408	0.9176	0.1205	316.4	103.15	3270
TSS-East	Nomoto	0.7472	0.1183	298.2	109.06	3475	0.8373	0.1169	309.4	103.51	3267

Table 29: An unmodelled current. Every cell is evaluated with the HYCOM current; the two column groups differ only in whether the policy was trained without or with it. Metrics as in Table 28.

	Sea current in			Goal rate at radius			gc-ADE
Motion model	simulator	expert labels	policy input	200 m	500 m	1000 m	(m)
Delta-current (3-D)	✓	✓	✓	**93.57**	**97.46**	**98.94**	**79.7**
	✓	✓	×	92.03	95.88	97.50	114.5
	✓	×	×	91.08	95.90	97.73	119.1
Water-track (2-D)	✓	✓	✓	72.15	90.52	96.10	**80.9**
	✓	✓	×	91.76	95.89	97.59	103.2
	✓	×	×	**93.13**	**96.99**	**98.22**	107.5
Port-glide (2-D)	✓	✓	✓	26.80	62.59	91.09	**82.9**
	✓	✓	×	**40.52**	**79.00**	94.65	106.9
	✓	×	×	37.58	77.69	**95.17**	111.6
Nomoto (2-D)	✓	✓	✓	47.36	82.67	93.27	**83.2**
	✓	✓	×	**83.73**	**94.27**	96.55	103.5
	✓	×	×	74.72	94.21	**96.58**	109.1

Table 30: Goal rate against the radius it is scored at, on the 92-day held-out split (23,566 episodes per cell). Goal rate is the percentage of episodes whose closest approach fell within the stated radius. Every row is evaluated in the current; the three check columns say where else it enters. Best per column within each dynamics model in bold.

on AEW and 0.020 on AESPA, against zero under Delta. Finally, the HYCOM record joins two products on 5 September 2024, so the anchorage training window is mostly non-tidal and the test window tidal; train–test differences under the current carry this confound.

The current can enter in three independent places, and Table 30 separates them. It is in the *simulator* for every row, so every policy is evaluated in it. It is in the *expert labels* when the vault was built with it, so that the imitation targets come from inverting the dynamics under the current; for the through-water models the heading is itself defined as the through-water course, so the current changes what the policy is asked to reproduce and not only the physics it meets. It is in the *policy input* when the observation carries the local current as world-frame components. The bottom row of each block therefore learned from still-water labels and was then placed in a current, a transfer result whose absolute value is not directly comparable with the two matched rows above it. Rollouts ran at the 100 m radius with the 1,000 extra steps the protocol grants, and the wider radii are recomputed from each episode's closest approach, which is not truncated at arrival.

Conditioning the policy on the current cuts gc-ADE for every dynamics model, from 114.5 to 79.7 m for Delta, 103.2 to 80.9 for Water-track, 106.9 to 82.9 for Port-glide and 103.5 to 83.2 for Nomoto. Goal rate moves the other way for the two-component models, falling from 91.8% to 72.2% for Water-track and from 83.7% to 47.4% for Nomoto at a 200 m radius, and the gap closes as the radius is relaxed.

Dynamics	$R_G \uparrow$	Collision rate $C \downarrow$	CPA (m)	gc-ADE $\downarrow$	FDE$_{\neg R} \downarrow$ (m)
Delta	0.9495	0.1014	359.5	78.54	2452
Port-glide	0.4160	0.1071	329.6	80.77	2517
Water-track	0.9596	0.0985	358.7	78.27	2452
Nomoto	0.8817	0.1017	349.1	80.22	2510

Table 31: Results of FM-BC with different dynamics models on TSS East, solo regime, with no sea current. Columns follow Table 32.

The right panel of Figure 2 shows why the two measures disagree. The goal signal is strong enough that an unconditioned policy still arrives. The current carries it off the demonstrated line for most of the transit, and it turns back towards the goal only at the end, so goal rate records an arrival while gc-ADE records the detour that preceded it. A policy that cannot observe the current is not failing to reach its destination; it is reaching it by a route the expert never took.

The TSS East dynamics comparison trains one policy per model while keeping the other settings fixed. Port-glide, the only model that cannot reproduce the recording, reaches 41.6% of goals; the three exact-replay models reach 88.2–96.0%. Collision rate, closest approach, and tracking error differ little, so the main effect of the dynamics choice is whether the vessel arrives.

O SINGLE-AGENT RESULTS IN THE NAVIGATION REGIONS

Section 5.3 drives every ego at once in the four navigation regions of Figure 8. This appendix reports the matched single-agent runs, in which one ego is under policy control and all other traffic log-replays, so that the two can be read against each other: the single-agent rows say how much of any failure belongs to the policy rather than to the multi-agent setting. Both regimes use the same fixed 1,000-step horizon of Appendix C.4 and the same 200 m re-scoring, and policies are trained separately per region.

Region	Method	$R_G \uparrow$	Collision rate $C \downarrow$	CPA (m)	gc-ADE $\downarrow$	FDE$_{\neg R}$ (m)
Intersection 1	BC-NF	0.9984 ± 0.0001	0.0742 ± 0.0012	269.9 ± 1.1	43.95 ± 0.05	852
Intersection 1	BC-FM	0.9967 ± 0.0018	0.0666 ± 0.0009	283.9 ± 0.8	46.61 ± 0.31	909
Intersection 1	Independent PPO	0.9325 ± 0.1144	0.1378 ± 0.0539	174.0 ± 23.2	572.04 ± 0.92	6083
Intersection 1	HR-PPO	0.9962 ± 0.0010	0.0729 ± 0.0020	280.4 ± 2.9	52.62 ± 0.53	1036
Intersection 2	BC-NF	0.9975 ± 0.0006	0.0654 ± 0.0020	290.2 ± 2.7	37.07 ± 0.25	601
Intersection 2	BC-FM	0.9982 ± 0.0006	0.0544 ± 0.0005	312.0 ± 0.7	37.96 ± 0.22	619
Intersection 2	Independent PPO	0.6999 ± 0.2739	0.2804 ± 0.1940	97.8 ± 96.9	362.97 ± 140.96	3352
Intersection 2	HR-PPO	0.9963 ± 0.0004	0.0585 ± 0.0009	311.5 ± 0.8	45.39 ± 0.72	707
TSS East	BC-NF	0.9095 ± 0.0029	0.1206 ± 0.0030	313.6 ± 7.8	75.73 ± 0.47	2319
TSS East	BC-FM	0.9456 ± 0.0078	0.1043 ± 0.0021	345.2 ± 2.1	79.08 ± 0.63	2457
TSS East	Independent PPO	0.9912 ± 0.0018	0.0920 ± 0.0079	444.1 ± 1.4	912.39 ± 2.92	17004
TSS East	HR-PPO	0.9367 ± 0.0067	0.1100 ± 0.0010	341.2 ± 2.9	84.98 ± 2.01	2607
TSS West	BC-NF	0.9465 ± 0.0015	0.0887 ± 0.0032	483.0 ± 4.5	72.06 ± 0.80	2053
TSS West	BC-FM	0.9514 ± 0.0047	0.0752 ± 0.0016	503.5 ± 3.8	73.25 ± 0.30	2113
TSS West	Independent PPO	0.9967 ± 0.0008	0.0493 ± 0.0135	729.0 ± 82.1	894.47 ± 2.24	16703
TSS West	HR-PPO	0.4196 ± 0.1274	0.0345 ± 0.0038	755.6 ± 50.6	94.84 ± 5.55	2218

Table 32: Four-method comparison on the four TSS and intersection regions, with one vessel under policy control and all other traffic replayed from the logs. R_G is scored at a goal radius $r_G = 200$ m. Each cell is the mean over seeds, with $\pm$ the standard deviation where all three seeds were evaluated; a dash marks a cell that has not been evaluated.

Two patterns are visible. First, imitation transfers to these regions: BC-NF and BC-FM reach 90.4–99.8% of goals, and the two intersections are easier than the straits, where episodes are longer and goals further away (FDE 607–898 m at the intersections against 2,017–2,423 m in the straits). Second, the reinforcement learning baselines are the weakest rows and fail region by region rather than uniformly: Independent PPO reaches 93.5% of goals in TSS West but 13.5% in Intersection 2, and its TSS West gc-ADE of 899.43 shows it reaching goals by a route unlike the demonstrated one.

Tables 33 and 34 report the kinematic results under the two regimes. BC-FM is the smoothest method in almost every cell, and the reinforcement learning rows carry the largest accelerations and

Region	Method	min $\tilde{\kappa}$	max $\tilde{\kappa}$	min a	max a	min drift	max drift
Intersection 1	BC-NF	-0.0079 ± 0.0001	0.0082 ± 0.0001	-0.5000 ± 0.0061	0.4498 ± 0.0050	-0.1150 ± 0.0050	0.1297 ± 0.0007
Intersection 1	BC-FM	-0.0041 ± 0.0001	0.0046 ± 0.0000	-0.2675 ± 0.0006	0.2395 ± 0.0012	-0.0961 ± 0.0054	0.1062 ± 0.0043
Intersection 1	Independent PPO	-0.0070 ± 0.0023	0.0326 ± 0.0271	-0.1304 ± 0.1166	1.5542 ± 0.0551	-1.7279 ± 0.3083	1.2727 ± 0.4292
Intersection 1	HR-PPO	-0.0023 ± 0.0001	0.0026 ± 0.0001	-0.0569 ± 0.0019	0.0767 ± 0.0026	-0.1128 ± 0.0017	0.1100 ± 0.0145
Intersection 2	BC-NF	-0.0056 ± 0.0001	0.0058 ± 0.0002	-0.5000 ± 0.0023	0.4411 ± 0.0005	-0.0861 ± 0.0047	0.1110 ± 0.0064
Intersection 2	BC-FM	-0.0026 ± 0.0001	0.0031 ± 0.0002	-0.2159 ± 0.0017	0.1910 ± 0.0013	-0.0561 ± 0.0015	0.0746 ± 0.0030
Intersection 2	Independent PPO	-0.0171 ± 0.0140	0.0761 ± 0.1022	-0.4865 ± 0.3944	1.6399 ± 0.0653	-1.9463 ± 0.4360	2.1608 ± 0.6974
Intersection 2	HR-PPO	-0.0013 ± 0.0001	0.0013 ± 0.0001	-0.0583 ± 0.0028	0.0819 ± 0.0024	-0.0601 ± 0.0039	0.0615 ± 0.0054
TSS East	BC-NF	-0.0057 ± 0.0001	0.0062 ± 0.0005	-0.7632 ± 0.0161	0.6450 ± 0.0217	-0.1589 ± 0.0408	0.1121 ± 0.0053
TSS East	BC-FM	-0.0019 ± 0.0001	0.0021 ± 0.0001	-0.2745 ± 0.0029	0.2444 ± 0.0028	-0.0844 ± 0.0097	0.0718 ± 0.0003
TSS East	Independent PPO	-0.0244 ± 0.0048	0.0120 ± 0.0082	-0.1093 ± 0.0203	1.5097 ± 0.0014	-2.5067 ± 0.6779	2.6987 ± 0.3925
TSS East	HR-PPO	-0.0013 ± 0.0005	0.0014 ± 0.0002	-0.0563 ± 0.0004	0.0860 ± 0.0005	-0.1094 ± 0.0023	0.0588 ± 0.0112
TSS West	BC-NF	-0.0031 ± 0.0001	0.0076 ± 0.0001	-0.8140 ± 0.0281	0.7218 ± 0.0248	-0.1182 ± 0.0008	0.0832 ± 0.0032
TSS West	BC-FM	-0.0009 ± 0.0002	0.0053 ± 0.0010	-0.3223 ± 0.0185	0.2785 ± 0.0150	-0.0985 ± 0.0131	0.0527 ± 0.0047
TSS West	Independent PPO	-0.0064 ± 0.0074	0.0175 ± 0.0089	-0.1725 ± 0.0489	1.4932 ± 0.0066	-3.0202 ± 0.0401	3.0037 ± 0.0732
TSS West	HR-PPO	-0.0455 ± 0.0077	0.0140 ± 0.0013	-0.0922 ± 0.0104	0.1115 ± 0.0099	-0.2964 ± 0.0873	0.2503 ± 0.0422

Table 33: Smallest and largest curvature, acceleration and drift reached during an episode, with one vessel under policy control and all other traffic replayed from the logs. Each cell averages over episodes, then over seeds; a dash marks a cell that has not been evaluated.

Region	Method	min $\tilde{\kappa}$	max $\tilde{\kappa}$	min a	max a	min drift	max drift
Intersection 1	BC-NF	-0.0445 ± 0.0052	0.0611 ± 0.0088	-0.9731 ± 0.0464	0.8198 ± 0.0335	-0.9462 ± 0.0589	1.0831 ± 0.0957
Intersection 1	BC-FM	-0.0196 ± 0.0093	0.0342 ± 0.0250	-0.4109 ± 0.0125	0.2443 ± 0.0027	-0.9511 ± 0.1289	0.8168 ± 0.0409
Intersection 1	Independent PPO	-0.0235 ± 0.0134	0.2130 ± 0.1449	-1.7383 ± 0.1230	1.7489 ± 0.1290	-2.8694 ± 0.2526	2.8628 ± 0.2823
Intersection 1	HR-PPO	-0.0032 ± 0.0005	0.0028 ± 0.0001	-0.2641 ± 0.0049	0.0714 ± 0.0014	-0.1210 ± 0.0042	0.1254 ± 0.0135
Intersection 2	BC-NF	-0.0583 ± 0.0108	0.0634 ± 0.0101	-1.2678 ± 0.2012	1.0525 ± 0.1628	-1.0948 ± 0.0168	1.1418 ± 0.0351
Intersection 2	BC-FM	-0.0271 ± 0.0137	0.0941 ± 0.0128	-0.3757 ± 0.0074	0.2027 ± 0.0056	-0.6663 ± 0.0809	1.1049 ± 0.1136
Intersection 2	Independent PPO	-0.0642 ± 0.0335	0.0806 ± 0.0940	-1.5510 ± 0.3307	1.7143 ± 0.0918	-3.0145 ± 0.0207	3.0026 ± 0.0672
Intersection 2	HR-PPO	-0.0047 ± 0.0020	0.0031 ± 0.0012	-0.1883 ± 0.0110	0.0832 ± 0.0036	-0.1044 ± 0.0557	0.2106 ± 0.1160
TSS East	BC-NF	-0.0296 ± 0.0050	0.0462 ± 0.0130	-1.2675 ± 0.0205	1.0861 ± 0.0233	-1.0671 ± 0.1083	1.1866 ± 0.1353
TSS East	BC-FM	-0.0123 ± 0.0004	0.0231 ± 0.0139	-0.5099 ± 0.0115	0.2696 ± 0.0026	-0.9031 ± 0.1376	0.5572 ± 0.2056
TSS East	Independent PPO	-0.1938 ± 0.2005	0.0442 ± 0.0463	-1.7619 ± 0.0647	1.6548 ± 0.0370	-2.8474 ± 0.3188	2.9369 ± 0.1939
TSS East	HR-PPO	-0.0013 ± 0.0006	0.0013 ± 0.0003	-0.3992 ± 0.0067	0.0849 ± 0.0015	-0.1147 ± 0.0033	0.0618 ± 0.0104
TSS West	BC-NF	-0.0166 ± 0.0075	0.0236 ± 0.0035	-1.3468 ± 0.0160	1.1532 ± 0.0380	-0.5837 ± 0.1109	1.0453 ± 0.2205
TSS West	BC-FM	-0.0083 ± 0.0066	0.0077 ± 0.0006	-0.4430 ± 0.0237	0.3131 ± 0.0027	-0.4533 ± 0.3729	0.8552 ± 0.5272
TSS West	Independent PPO	-0.0111 ± 0.0054	0.0314 ± 0.0145	-1.5944 ± 0.1408	1.5012 ± 0.0118	-3.0516 ± 0.0280	3.0342 ± 0.0573
TSS West	HR-PPO	-0.0435 ± 0.0075	0.0149 ± 0.0010	-0.2520 ± 0.0375	0.1112 ± 0.0090	-0.3052 ± 0.0813	0.2557 ± 0.0426

Table 34: Smallest and largest curvature, acceleration and drift reached during an episode, with every vessel under policy control at the same time. Each cell averages over episodes, then over seeds; a dash marks a cell that has not been evaluated.

drift angles, consistent with their tracking errors. Curvature, acceleration and drift all reach larger values under joint control for every method, which is expected when neighbours manoeuvre rather than replay.

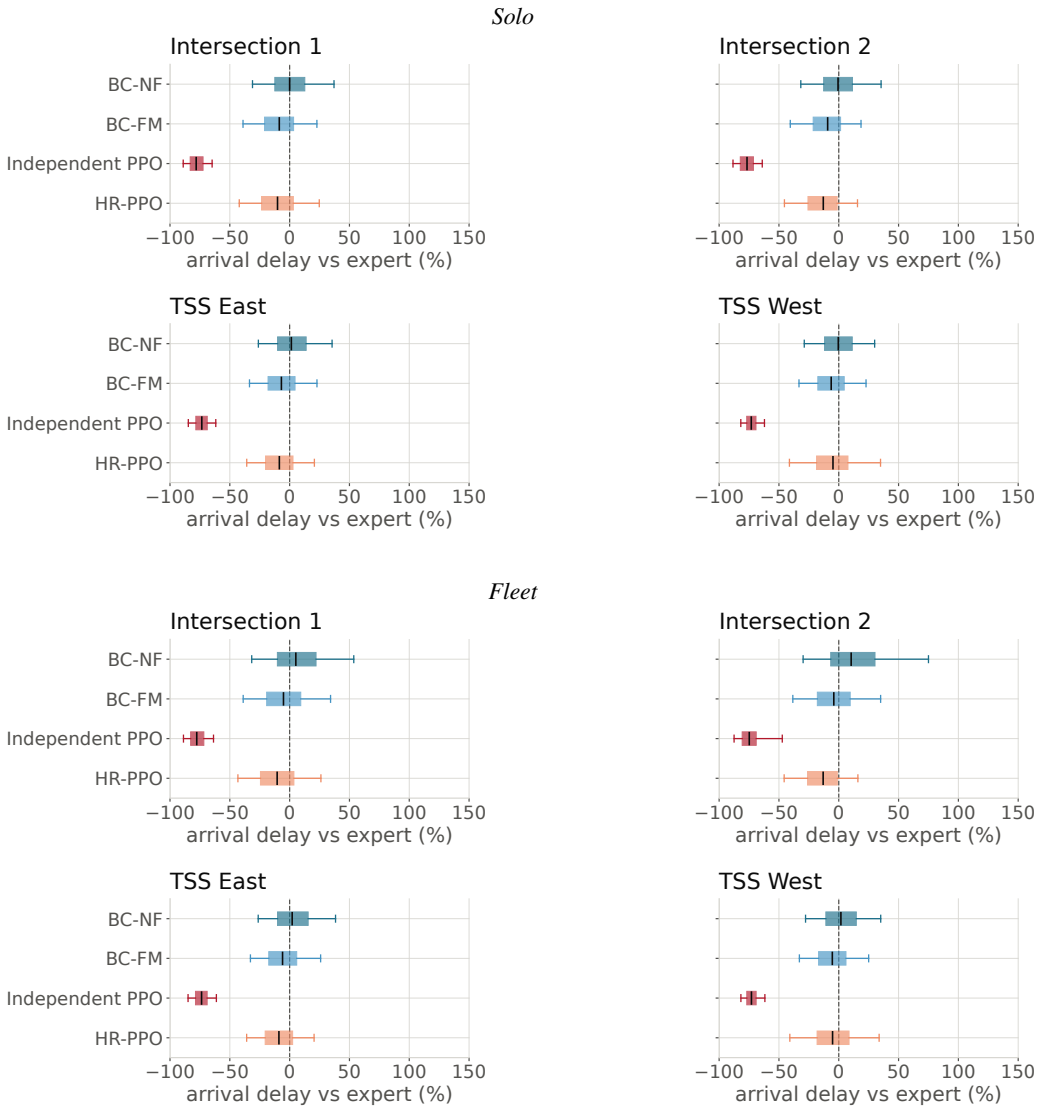


Figure 11: Arrival delay against the expert, over episodes, for the four learned methods in each navigation region under both control regimes. Negative values are earlier than the recorded transit and positive values later; the dashed line marks the expert, and each box shows the interquartile range with the median.