Mixed-Fidelity Reinforcement Learning for Aircraft Conflict-Resolution

A. Moec[1] & D.J. Groot[1], and J. Ellerbroek[1]
[1]Control and Simulation, Faculty of Aerospace Engineering, TU Delft

***Abstract*—The growing density of civil air traffic is tightening operational safety margins and motivating the search for data-driven conflict-resolution policies. However, the rising compute demand for the training of AI models collides with the need to minimize its environmental impact. In an effort to reduce this climate impact, this paper investigates mixed-fidelity reinforcement learning (MiFi RL) as an alternative to training in high-fidelity (HiFi) simulators only, by first pre-training in a computationally lightweight low-fidelity (LoFi) environment before fine-tuning in HiFi.**

We analyze this paradigm across five single-agent algorithms – A2C, PPO, DDPG, SAC, and TD3 – using a fixed training budget of 3 million timesteps. Off-policy methods yield a large curriculum benefit: with a 60% LoFi / 40% HiFi split, SAC achieves a 24% increase in evaluated HiFi reward and a 20% reduction in wall-clock training time relative to pure-HiFi training; DDPG attains gains of 37% and 16% at a 40% LoFi share. In contrast, the on-policy algorithms exhibit negligible or negative improvements, possibly underscoring the replay buffer's role in mitigating the domain shift between simulators. Efficient curriculum setup can alleviate computational load and environmental impact while improving final policy performance.

***Keywords*—Mixed-Fidelity Reinforcement Learning (MiFi RL), Air Traffic Management (ATM), Aircraft Conflict-Resolution, High-Fidelity Simulation, Soft-Actor-Critic (SAC), BlueSky Simulator, Artificial Intelligence**

I. Introduction

As commercial aviation sees a return to pre-COVID levels [1], [2] in combination with notable air traffic controller (ATCo) fatigue in the United States [3], [4], there have been a worrying number of near-misses and close-calls both on the ground and mid-flight in recent years [5], [6]. With critical hiring goals not met, the consequent labor force shortage ensures that ATCos remaining in the industry face far more challenging conditions, with longer shifts and higher individual levels of stress [5]. This environment naturally results in a dangerous situation wherein the probability of fatal or significant human error increases.

The potential for artificial intelligence (AI) in air traffic control (ATC) has been noted by EUROCONTROL [7], and could partially reduce the workload demanded of existing ATCos. Indeed by its very nature, ATC involves repetitive procedures that generate vast amounts of data, which lends itself well to data-driven methods such as machine learning (ML). Reinforcement learning (RL) has specifically been an active field of research for this application [8]. There is presently, however, no consensus on the training environment in which RL solutions are deployed, as demonstrated by Wang et al. in their paper reviewing past and present RL approaches to aircraft conflict-resolution [8]. While certain models, such as the one proposed by Brittain and Wei [9], make use of the BlueSky Air Traffic Simulator [10] as a training environment, others opt for lower-fidelity (LoFi) environments that may, for instance, disregard turning radius and assume instantaneous actions [11]. Beyond these ATM-specific approaches, reinforcement learning for safety-critical control has also explored entropy-regularized off-policy methods to improve robustness, safe or constrained RL to enforce operational limits [12], [13], and sim-to-real techniques such as domain randomization to narrow the reality gap [14]. These lines of work are complementary to the mixed-fidelity approach studied here, because they target the policy's ability to generalize rather than the simulator itself.

Mixed-Fidelity (MiFi) training – where agents first learn in faster, simplified simulators before fine-tuning in high-fidelity environments – has proven to be effective in robotics and CFD-based optimization, cutting expensive simulator calls by 30% with minimal loss in policy performance [15], [16]. However, to our knowledge no prior work has applied or quantified such mixed-fidelity RL in aircraft conflict-resolution, nor measured the drop in performance when switching between simulators (transfer penalty) or training-time savings in this domain. As we move toward real-world deployment of RL-based ATC tools, measuring the LoFi to HiFi transfer penalty gives us a tangible proxy for the expected degradation when moving from high-fidelity simulators to live airspace.

Moreover, higher-fidelity (HiFi) simulations and ML training in High-Performance Computing (HPC) centers contribute substantially to carbon emissions – data centers alone are projected to account for up to 8% of global emissions by 2030 [17]. As the use of AI expands, novel methods of reducing reliance on energy-intensive HiFi simulators is thus both computationally and environmentally critical to meet the European Union's 2050 net-zero emissions target [18].

Against this backdrop, the present study pursues these following objectives:

1) **Generalization -** Determine whether both off and on-policy single-agent RL models first trained in a LoFi environment can complete training in a HiFi environment without a loss in conflict-resolution performance as compared to controls trained exclusively in the HiFi environment.
2) **Transfer Penalty -** Quantify any performance penalty incurred in switching training from the LoFi to HiFi

environment during training, and the speed at which recovery occurs, if applicable.

3) **Mixed-Fidelity Optimization -** Determine whether the overall training time can be reduced using this hybrid training approach, and particularly in minimizing the time spent in the HiFi environment without significant performance drop.
4) **Optimal Algorithm Identification -** Determine whether algorithmic performance is consistent across fidelities to extrapolate the best RL algorithm for the task at hand.

II. Environments

The choice of LoFi and HiFi simulators to perform the MiFi training rested on the familiarity of the underlying framework and codebase, as building purpose-fit LoFi and HiFi simulators were not the aim of this paper. Instead, the LoFi environment is derived from the same RL environment used by Badea et al. [11]. Modifications pertaining to this environment consisted of wrapping the existing codebase as a Gym environment built upon The Farama Foundation's Gymnasium API [19]. Similarly, a modified version of the BlueSky-Gym *SectorCREnv-v0* environment as published by Groot et al. was used for the HiFi training in order to leverage both the high-accuracy air traffic simulation provided by the BlueSky simulator, and the same Gymnasium framework [10]. It is worth noting that given the lack of standard for simulator fidelity, the terms LoFi and HiFi correspond to the relative fidelities of the environments used. In this study, one features instantaneous agent actions, disregarding physical constraints imposed by true aircraft motion, while the other takes turning radii into account as the simulation advances for instance.

A. Low-Fidelity Simulator

In keeping with the low-fidelity environment used by Badea et al. [11], the modified environment is an instantaneous-action, two-dimensional, and non-physical representation of a dense airspace. Heading and velocity changes enacted by the RL agent are therefore instant and disregard non-negligible aircraft dynamics such as turning radius, acceleration, and physical realities such as the curvature of Earth. While excellent for rapid training due to the minimal on-step computational overhead, this environment fails to capture aspects of flight that would naturally have a significant impact on real-world performance. Ignoring turning radii would, for instance, lead an RL agent to drastically underestimate the time needed to avoid a conflicting aircraft.

The original *atcenv* environment by Badea et al. [11], [20] was adapted to be compatible with the open-source Stable-Baselines3 Python library. This involved refactoring its codebase to be a Gym environment as specified by the guidelines of Farama [21]. For more implementation details, see the project repository [22].

B. High-Fidelity Simulator

BlueSky-Gym's *SectorCREnv-v0* environment similarly simulates a dense, two-dimensional airspace, but contrasts the LoFi simulator in its underlying use of the BlueSky ATM simulator for aircraft flight dynamics and performance [10]. Heading and velocity changes thus register as time-dependent modifications to the aircraft's flight path and trajectory, and while still two-dimensional, distance calculations between aircraft account for the Earth's curvature by comparing coordinates instead of the simple euclidean distances used in the LoFi environment. Agents training in this environment must therefore adapt both to a modification in the time taken to effectuate changes, and a different definition of distance. Details on modifications made to this base environment for MiFi training compatibility can also be found in the project repository [22].

C. Conflict Scenarios

Delineating the exact nature of training scenarios irrespective of the simulator fidelity is of critical importance in allowing for a direct comparison in performance between LoFi and HiFi-trained RL models. The conflict scenarios have similar characteristics to those used by Badea et al. [11], with the generated airspace modeled as an n-sided circumscribed polygon, whose number of sides varies until its area lies between 2400 and 3750 NM^2.

The scenario variation between episodes largely stems from the random airspace area coupled with the Gaussian distribution of traffic density per nautical mile squared, ρ_{AC}, given by Equation 1.

$$\rho_{AC} \sim N\left(0.005,\, 0.001^2\right) \tag{1}$$

To determine the consequent number of aircraft, N_{AC}, this traffic density is multiplied by the total airspace area. In the case that the number of aircraft is lower than the number required to construct the input vector, $N_{AC,\,state}$, the total number of aircraft is set at $N_{AC,\,state} + 1$ as described by Equation 2.

$$N_{AC} = \max\left[\text{round}\left(A_{poly} \cdot \rho_{AC}\right),\, N_{AC,\,state} + 1\right] \tag{2}$$

During the scenario creation, each aircraft – including the agent – is also assigned a random velocity according to the distribution given by Equation 3. The corresponding number of aircraft is then randomly generated along the boundary of the n-sided airspace together with an associated waypoint. The waypoints are placed randomly along the perimeter of the polygon and assigned to each aircraft, thus resulting in straight-line flight for the uncontrolled aircraft.

$$v \sim U\left(200,\, 250\right) \tag{3}$$

Particularly relevant to the HiFi simulator is the explicit selection of the aircraft model, due to its subsequent impact on aircraft flight dynamics and performance modeling. The decision was thus made to use the Airbus A320 for all the generated aircraft since this research centers on the model transfer from the absence to inclusion of flight dynamics and

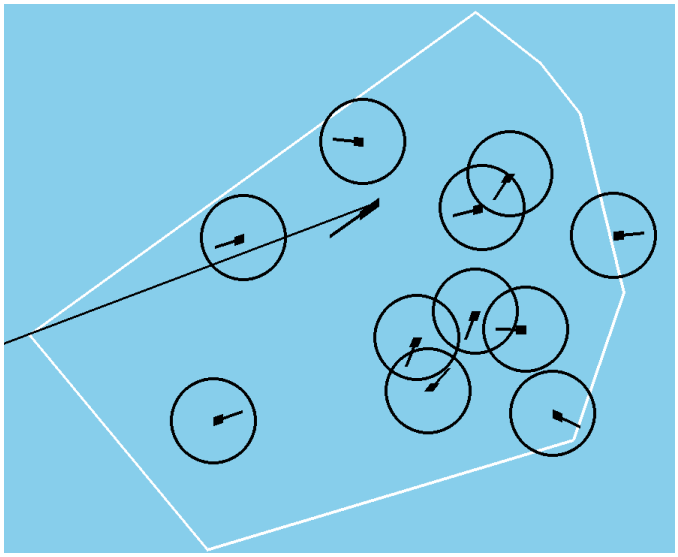

Figure 1. Render of an example conflict scenario in the HiFi simulator – agent is identified by the lack of a surrounding intrusion boundary. Agent heading and target is identified by the short and long black lines respectively.

performance, not the nuances therein which would nevertheless likely be relatively minor.

The use of this initialization logic ensures a suitably complex scenario for both the LoFi and HiFi environment, an example of which can be visualized in Figure 1. Note that while this render is specifically from the HiFi simulator, the LoFi simulator scenarios would be identical. The introduction of fluctuations in these scenario variables aim to approximate the local variations in aircraft concentrations and paths that could be encountered in real conflict-resolution scenarios. Proximity to major hubs or routes would, for instance, have a large impact on the sorts of resolution scenarios an RL algorithm might expect. Indeed, it is important to use random scenarios to prevent overfitting of the model, in which case it would memorize the solution to one specific conflict or airspace geometry. While the stochastic scenario generator will eventually produce some infeasible conflict cases, its avoidance of duplicate scenarios exposes the agents to a wider state space and thus improves generalization.

III. Experiments

Having outlined the nature of the simulators used for the MiFi training alongside the conflict scenario under consideration, the experimental approach adopted in order to meet this research's objectives are described below. Aside from promoting the intelligibility of the results through an accounting of the experimental process, careful consideration is placed on the experiments' reproducibility and the impact of stochastic effects throughout.

A. RL Algorithms

As suggested by the choice of Stable-Baselines3 as the RL backbone across simulators, this research investigates the effects of MiFi training across both on- and off-policy RL algorithms. A2C and PPO are used for the former, while SAC, TD3, and DDPG serve to investigate the off-policy performance. On-policy algorithms learn solely from trajectories generated by the current policy, updating parameters based on freshly collected data, whereas off-policy algorithms reuse past experience via a replay buffer to evaluate and improve a different target policy. In all cases, the default hyperparameter values provided by the Stable-Baselines3 `v2.6.1a1` algorithm classes are used, so as to forego the hyperparameter tuning process. Since this research focuses on the relative performance between algorithms and the relative impact of simulator fidelity – the absolute performance of these algorithms does not need to be optimized via hyperparameter tuning. It should be noted, however, that RL algorithms, particularly on-policy ones, are known to be sensitive to hyperparameter choices, random seeds, and even minor implementation details, which can affect apparent performance and reproducibility [23]. To make the cross-algorithm comparison tractable, we fixed the main hyperparameters across fidelities and did not perform per-algorithm tuning, so the results should be read as relative trends under a shared budget rather than best-possible numbers for each method. A fuller sensitivity study in LoFi followed by transfer of the tuned configuration to HiFi would be a natural extension.

Irrespective of RL algorithm, all models share the same action and observation spaces. Since the current implementation of BlueSky-Gym constrained the research to the single-agent domain [24], this yields an observation space described by Equation 4 and expanded upon in Table I.

$$\mathcal{O} = [-1,1]^2 \times \mathbb{R} \times \left(\mathbb{R}^5 \times [-1,1]^2\right)^{N_{AC,\text{state}}} \tag{4}$$

TABLE I. Observation-space components for the MiFi training environments.

Component	Domain	Description
Ownship state		
$\cos(\text{drift})$, $\sin(\text{drift})$	$[-1,1]$	Cosine and sine of ownship's drift angle
Airspeed	$\mathbb{R}$	Ownship airspeed
Traffic states (per each of $N_{AC,\text{state}}$ nearest aircraft)		
x_r, y_r	$\mathbb{R}$	Relative position offsets in x and y
v_x, v_y	$\mathbb{R}$	Relative velocity components
$\cos(\text{track})$, $\sin(\text{track})$	$[-1,1]$	Cosine and sine of each aircraft's track angle
Distance	$\mathbb{R}$	Euclidean distance to each aircraft

The controlled aircraft has – as proposed by Badea et al. [11] – a two degree-of-freedom (DoF) action space, with the ability to make bounded but continuous changes to their heading and velocity of $\pm 22.5^{\circ}$ and ± 6.67 kts respectively. Both control inputs are issued jointly at each timestep, that is, the agent selects a heading change and a speed change simultaneously. While the shape of these action and observation spaces remains constant across algorithm and simulator

fidelity, the calculations used therein may not, as set out in Section II. The inclusion of the Earth's curvature, for instance, changes the numerical values of distance between the agent and uncontrolled aircraft. Moreover, the effects of actions taken by the agent varies greatly across simulator fidelities. As the LoFi environment implements agent actions instantaneously, the transient behavior of an aircraft changing heading or velocity is ignored. In the HiFi environment, however, these transient behaviors stemming from agent actions are factored into the aircraft trajectory, with a 22.5° heading adjustment taking 8.44 seconds of simulation time to be effectuated at 250 m/s for instance.

The reward function used in both LoFi and HiFi training is the same and is an unmodified implementation of that used by Badea et al. [11]. The per-step reward is defined as follows:

$$r_t = -\alpha\delta_t - \beta \sum_{i=1}^{N_{AC}-1} 1(d_{t,i} < D_{int}),$$

where δ_t is the instantaneous magnitude of the agent's drift in radians, $d_{t,i}$ is the distance between the agent and the i^{th} aircraft in the airspace, and D_{int} is the intrusion distance, defined as being five nautical miles. Coefficients α and β are 0.1 and 1 respectively, resulting in a severe intrusion penalty, and a moderately severe deviation penalty. These coefficients were taken directly from Badea et al.'s reward function given their proven functioning in the LoFi environment [11]. Note that given the negative nature of the reward function, the largest possible reward that can be attributed at a single timestep is $r_t = 0$.

Note also that although STABLE-BASELINES3 prescribes algorithm-specific default networks (e.g. 64×64 for PPO/A2C, 256×256 for SAC, and 300×400 for TD3/DDPG [25]), these defaults were overridden to standardize every agent on the same two-layer, 256-unit-per-layer multilayer perceptron (MLP). By holding network capacity constant, this ensures that any observed differences in learning performance arise from the algorithms themselves or the simulator fidelity – not from disparate state-representation architectures.

B. Transfer Policy

Training runs were performed sequentially; first passing through the LoFi environment for a prescribed number of timesteps before completing training in the HiFi environment. In order to quantify the performance delta between the pure HiFi training and the MiFi approach, full training runs were also performed in the HiFi environment. Pure LoFi training runs were also performed in order to quantify the base disparity in performance between a LoFi- and HiFi-trained model.

A set number of timesteps (3×10^6) was chosen cumulatively across both LoFi and HiFi simulators. This corresponds to 20,000 episodes within which convergence for the non-transferred models was ensured, forming a consistent baseline. For off-policy algorithms, the buffer was also transferred between simulators leading to a period of blended training where the algorithm would be exposed to time-histories from both simulators. For this research, training was performed across 0-100% (inclusive) timesteps in the LoFi simulator in increments of 20%. This ensured the collection of a wide range of results, while keeping total computational resources and time required bounded.

C. Experimental Setup

To account for the inherent stochasticity of RL, each agent, for each training split (i.e. percentage of timesteps spent in the HiFi environment) was run on seven independent pseudo-random seeds (42-48 inclusive) using NUMPY's random generator (PCG64) [26], [27]. Evaluating over multiple seeds mitigates the risk of drawing conclusions from random trends or lucky initializations [23]. Indeed, as Henderson et al. [23] note in their research, reporting top results across random trials, or alternatively, reporting averages across a small ($N < 5$) sample size can be misleading. In an effort to avoid this pitfall in RL research, a large number of seeds ($N = 7$) were used. Evaluation episodes were also run on seed zero such that all models experienced a novel series of evaluation scenarios that would not bias the performance of any model.

All training jobs were executed on NVIDIA V100 GPUs of the DelftBlue supercomputer at the Delft High Performance Computing Centre (DHPC) [28]. Each job – constituting of a full 0-100% LoFi- and HiFi-trained series of models – was allocated one GPU and one CPU to ensure consistent compute resources across seeds.

D. Recorded Data and Evaluation Metrics

The total per-episode reward stemming from the summation of the individual r_t's as presented in Subsection III-A was recorded for each episode across all training runs. Alongside this, the total time spent in the LoFi and HiFi simulators across all LoFi training percentages was recorded as a proxy for computational load. Given the lack of accurate power-monitoring tools for the server-side jobs, the relative times spent in the LoFi and HiFi environments serve to approximate their relative environmental impact, as well as the overall MiFi training efficiency as a function of training split.

Additionally, the use of several seeds for each training run made it possible to average performance metrics across seeds and quantify their variance. These performance metrics aim to capture both the final and transient behavior of the algorithms subjected to an environmental transfer, as well as quantify the efficiency of the MiFi approach as opposed to pure HiFi training. In this vein, the following performance metrics are proposed and collected:

1) $\bar{\mathbf{R}}_{\mathbf{eval}}\,[-]$: A measure of the final performance of MiFi-trained algorithms when compared to their purely LoFi- or HiFi-trained counterparts. It consists of the average reward across $N = 1000$ episodes evaluated in the HiFi environment and setting the seed to zero to introduce novel scenarios to all trained models.
2) $\mathbf{\Delta\bar{R}_S}\,[-]$: A measure of the absolute drop in performance occurring immediately after an environment

switch (transfer penalty). It consists of the absolute difference between the pre- and post-switch 100 episode average reward. Note that naturally this metric is only valid for MiFi-trained models where an environment switch occurs.

3) $\mathbf{T_{train}}$ [hr]: The total training time across both the LoFi and HiFi-trained episodes.

In addition, the $\bar{R}_{eval}$ metric collected for the purely LoFi- and HiFi trained algorithms allow for an average cross-seed ranking of the algorithms' evaluated performance. The stability of this ranking from LoFi to HiFi simulators allows us, in turn, to ascertain the degree to which a performance in LoFi translates to performance in HiFi. Consequently, a high Spearman rho, coupled with a low p-value, offers evidence as to the utility of initially benchmarking RL algorithm performance in a lower-fidelity simulator, and subsequently selecting the best performing algorithm to continue on to the higher-fidelity training.

E. Hypotheses

The MiFi concept rests on a simple tension: LoFi pre-training is cheap but risks over-specializing to LoFi dynamics, while HiFi training is costly yet more representative of the real world. As such, we test four directional hypotheses, each paired with the metric that adjudicates it:

H1: **Performance degradation.** As the proportion of LoFi timesteps (%LoFi) increases, the final HiFi reward $\bar{R}_{eval}$ decreases. [Metric: $\bar{R}_{eval}$]

H2: **Transfer penalty.** The immediate post-switch drop in reward $\Delta\bar{R}_S$ grows with %LoFi as the policy must unlearn LoFi-specific trajectories. [Metric: $\Delta\bar{R}_S$]

H3: **Algorithm robustness.** Off-policy methods (DDPG, SAC, TD3) incur a smaller transfer penalty than on-policy methods (A2C, PPO), owing to experience-replay exposure to a blended (LoFi/HiFi) state distribution during environment switch. [Metric: $\Delta\bar{R}_S$]

H4: **Predictive validity.** Ranked algorithm LoFi performance generalizes: the Spearman correlation between LoFi and HiFi ranks exceeds 0.6 with $p < 0.05$.

IV. Results

The results, that is, the aggregated performance metrics across algorithms, splits, and seeds, are presented in this section. Within each heatmap, the on-policy algorithms (A2C & PPO) are separated from the off-policy algorithms in order to distinguish their markedly different approaches to RL. The heatmap cells are populated by the cross-seed averaged performance metrics detailed in Section III, alongside their standard deviation.

A. Performance

Figure 2 demonstrates the average evaluation performance of the algorithms across training splits when evaluated in the HiFi environment. The 100% and 0% LoFi-trained models representing the purely LoFi- and purely HiFi-trained algorithms respectively are included. The pure LoFi-trained algorithm displays a markedly lower average reward than those that are MiFi- or HiFi-trained. This is expected given their lack of exposure to the higher-fidelity aircraft dynamics and non-instantaneous actions that would be present in the HiFi simulator. Their inclusion thus serves as evidence for the markedly different simulator fidelities and shows that a purely LoFi-trained algorithm would be insufficient in a higher-fidelity environment.

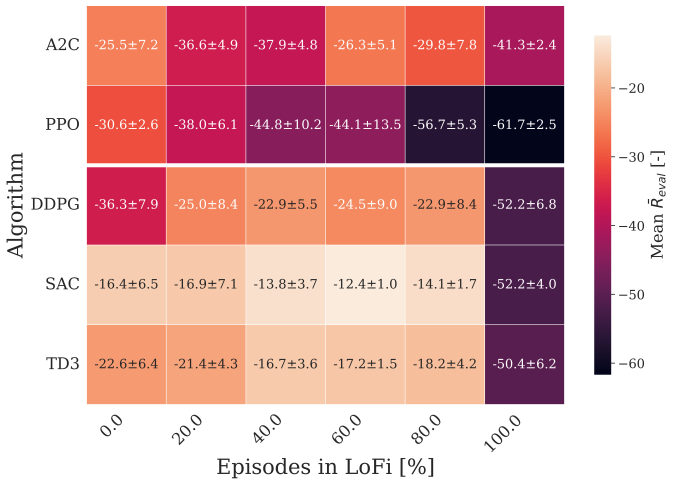

Figure 2. Average absolute evaluation reward ($\bar{R}_{eval}$).

Aside from the much lower $\bar{R}_{eval}$ metrics observed in the 100% LoFi-trained algorithms, a clear dichotomy in the effect of MiFi training can be seen between the on- and off-policy algorithms. Indeed, not only does A2C and PPO obtain lower $\bar{R}_{eval}$ metrics than their off-policy counterparts, but the $\bar{R}_{eval}$ as a function of training split does not follow the same trend. PPO progressively returns worse $\bar{R}_{eval}$ metrics as the MiFi split is increased, indicating that any level of blended training is detrimental to the overall performance of algorithm as hypothesized in H1. A2C, the other on-policy algorithm tested, experiences a decrease in the $\bar{R}_{eval}$ metric up until a MiFi split of 60%, at which point a sudden increase in the metric quickly fades back to the poor performance of A2C for this conflict-resolution application.

The off-policy algorithms (DDPG, SAC, and TD3), by contrast, all show a similar trend in $\bar{R}_{eval}$ as the MiFi split is increased. As the proportion of blended training increases, the $\bar{R}_{eval}$ increases, indicating superior performance when compared to the purely HiFi-trained algorithm. The observed behavior surprisingly goes contrary to H1 detailed in Section III. While SAC is generally a highly-performing model for this conflict-resolution application given its generally high average evaluation reward, DDPG and TD3 benefit from higher relative gains thanks to blended training. DDPG, for instance, benefits from a 36.9% increase in performance if 80% of the training is done in the LoFi simulator. Moreover, this same 80% MiFi training results in a comparable $\bar{R}_{eval}$ to purely HiFi-trained SAC or TD3.

The trends in variance are more complex to discern and may be tenuous at best, although for SAC, an improvement

in $\bar{R}_{eval}$ is generally accompanied by a reduction in cross-seed variance. The overall low standard deviation, especially compared to the obtained $\bar{R}_{eval}$ metrics does point, however, to a consistency in the impact of blended training irrespective per-episode stochastic variation.

B. Transfer Penalty

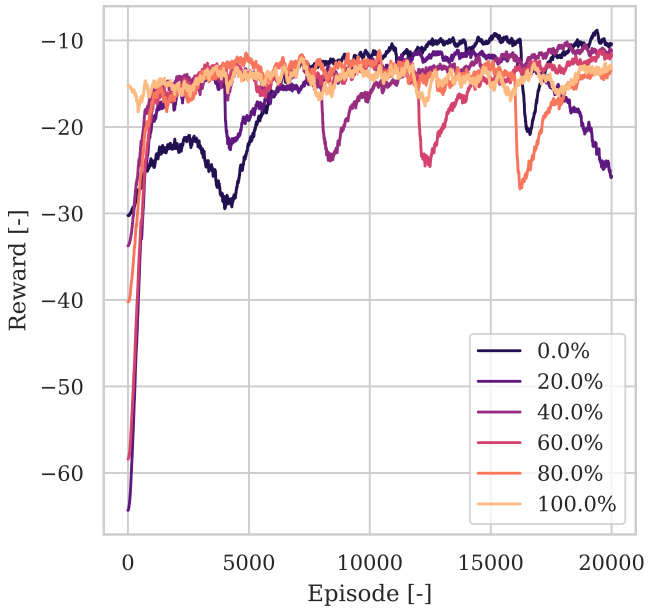

Figure 3. Example of SAC MiFi training reward over episodes. Observe the dips in reward, these correspond with a change from LoFi to HiFi and represent the transfer penalty. Note also that the dips seen for the 0.0% HiFi-trained agent are outliers likely stemming from instability during training since updates to the network are based on a sample of experiences, instead of the entire memory.

The transient effect of transferring training of a model from a LoFi to HiFi environment is graphically represented by Figure 3. Corresponding to one of the SAC MiFi training runs, the successive transfer penalties are evident from the dips in reward observed upon environment switch. The calculated transfer penalties are reported in Figure 4. PPO experiences a severe transfer penalty ($\Delta\bar{R}_S$), with high variance between seeds and no clear trend with respect to the training split. Interestingly, this phenomenon does not appear to be policy-specific given the far lower transfer penalties experienced by A2C models. There, transferring from one level of simulator fidelity to another results in only a small transient loss in reward, with generally smaller cross-seed variance. Indeed, while PPO demonstrates little consistency in transfer penalties across seeded scenarios, the smaller transfer penalties of A2C are themselves accompanied by smaller and roughly constant (≈ 4) standard deviation. Note that the highest $\Delta\bar{R}_S$ value occurring at a 40% split corresponds to an outlier transfer penalty of approximately -35.

Within the off-policy algorithms, two clear groups emerge with regards to transfer penalties. While all such algorithms (DDPG, SAC, and TD3) experience performance drops of a similar magnitude, the cross seed variance of DDPG and TD3

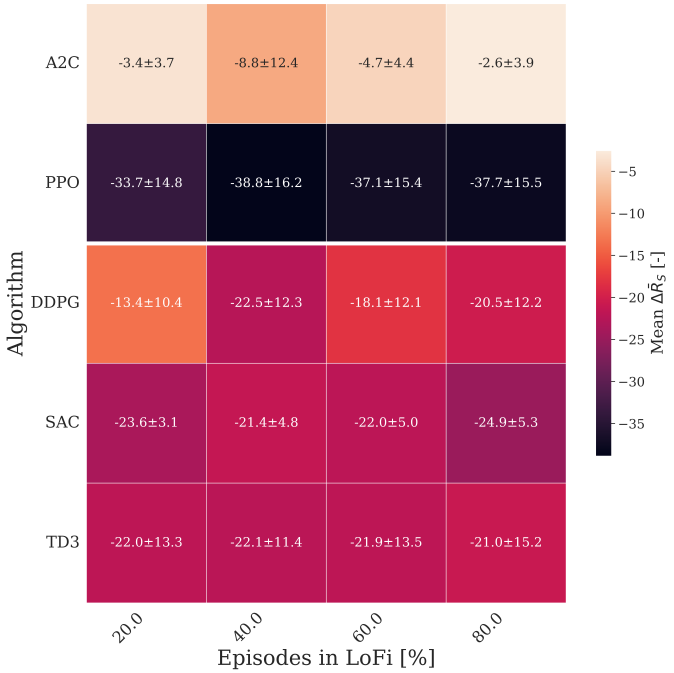

Figure 4. Average absolute transfer penalty ($\Delta\bar{R}_S$).

$\Delta\bar{R}_S$ values is significantly higher than those for SAC. The average standard deviation of 11.8 and 13.4 for DDPG and TD3 respectively is much larger than that of SAC being 4.6. Interestingly, however, the number of episodes spent in the LoFi simulator prior to the switch seems to have a negligible impact on $\Delta\bar{R}_S$, be it an on- or off-policy algorithm. The independence of transfer penalties with regards to both %LoFi and policy type is quite striking and counters hypotheses H2 and H3 presented in Section III

C. Mixed-Fidelity Optimization

Given one of the core research questions of this paper being MiFi training's potential for reducing overall training times and consequently, reducing financially and ecologically costly simulator calls, the total training times each algorithm across each training split is reported in Figure 5. Naturally, the aggregate training times across both LoFi and HiFi environments decreases as the MiFi split increases. Since the LoFi environment does not rely on expensive calls to the Bluesky ATM simulator for aircraft dynamics and motion, this trend is to be expected. Additionally, the dichotomy in training times between on- and off-policy algorithms is to be expected given the larger per-step gradient updates associated with the latter [29].

Perhaps more noteworthy is the percentage reduction in total training time garnered by the blended training approach. A2C and PPO both benefit from an over 50% reduction in total training time by spending 80% of their episodes in the LoFi environment whereas DDPG and TD3 both hover around a 35% reduction only. SAC, however, benefits only from a 28.4% reduction in total training time. While still impressive in their ability to reduce their overall dependence on dedicated

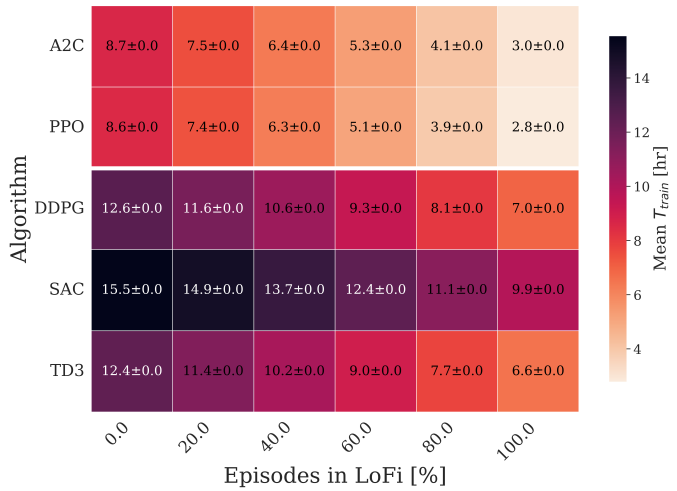

Figure 5. Average total training times (T_{train}).

training environments and hardware for long periods of time, this discrepancy hints at the fact that for off-policy algorithms the model updates form a larger proportion of the overall training overhead, even when using high-fidelity environments such as the BlueSky ATM simulator. In fact, as the time taken for the simulator to run through one episode increases in contrast to the algorithmic time, off-policy becomes a stronger contender for MiFi training for wall-clock time optimization.

D. Optimal Algorithm Identification

In an effort to evaluate whether low-cost LoFi training could be an accurate predictor as to the best performing RL algorithm for a given application, the purely LoFi- and HiFi-trained models were ranked according to their $\bar{R}_{eval}$ metrics. These ranks, averaged across seeds, are provided in Table II.

TABLE II. AVERAGE EVALUATION PERFORMANCE RANKING OF PURE LOFI AND HIFI-TRAINED MODELS.

Algorithm	LoFi	HiFi
SAC	1.71	1.29
TD3	2.00	2.71
PPO	3.00	3.86
DDPG	3.71	4.43
A2C	4.67	2.71

While SAC was the highest performing algorithm across both LoFi and HiFi simulators – and would therefore have substantiated the above hypothesis – the average rank of all algorithms does not show the same stability. The Spearman rho for these two ranked datasets is evaluated to 0.56, which, while suggesting moderate positive monotonic link between performance in LoFi and HiFi, must be balanced by the subsequent p-value. The associated significance was obtained with an exact permutation test over all 5! rank permutations, yielding $p = 0.3719$. This p-value therefore does not support H4 detailed in Section III, likely requiring many more seeds to come to a statistically significant result.

V. DISCUSSION

Aggregating the individual metrics and results demonstrated in Section IV, it becomes clear that certain algorithms, and indeed, entire classes of algorithms, are better suited to MiFi training. On the whole, not only do on-policy algorithms perform poorly at conflict-resolution given the spread of $\bar{R}_{eval}$ metrics in Figure 2, but, in the case of PPO, performance *degrades* as the proportion of LoFi episodes increases.

Comparing these $\bar{R}_{eval}$ results with the off-policy results immediately points to the replay buffer as a decisive component. Figure 4 shows that off-policy agents suffer pronounced instantaneous transfer penalties ($\Delta\bar{R}_S$), yet they rebound and ultimately *outperform their pure-HiFi counterparts*. The key distinguishing feature is that MiFi training moves *both* the network parameters *and* the replay buffer from LoFi to HiFi. Hence DDPG, SAC, and TD3 begin HiFi training with a mixed memory of LoFi *and* early-HiFi transitions. This blended distribution smooths the domain shift – effectively a form of on-the-fly data augmentation – and allows the critic to refine rather than relearn its value landscape. An ablation study, that is, retraining SAC while *flushing* its buffer at the fidelity switch, would directly test this hypothesis and is left for future work.

A second algorithmic factor is SAC's entropy-maximizing objective. By explicitly rewarding higher policy entropy during training, SAC maintains broader exploration and can adjust more readily when the transition dynamics change at the LoFi to HiFi switch. On-policy methods such as PPO, even with an entropy bonus, typically converge to lower-entropy policies and thus face a sharper exploration–exploitation dilemma once the HiFi dynamics are introduced. This provides a plausible explanation for why the MiFi curriculum benefits the off-policy, maximum-entropy method most strongly [12].

SAC is the clearest beneficiary: its cross-seed variance remains low in both $\bar{R}_{eval}$ and $\Delta\bar{R}_S$, indicating a predictable response to fidelity increases. A $60\,\%$ MiFi split yields a $24.4\,\%$ boost in average evaluation reward and trims total training time by $20\,\%$. More broadly, Figure 2 suggests that off-policy, single-agent RL can generalize conflict-resolution policies in HiFi simulators *more efficiently than pure HiFi training*. Although the $\bar{R}_{eval}$ values for DDPG and TD3 are lower than those achieved by SAC, they nonetheless benefit from even more impressive gains in both $\bar{R}_{eval}$ and T_{train}. DDPG, for instance, obtains a 36.9% increase in $\bar{R}_{eval}$ for a higher training split (40%), corresponding to a 15.9% decrease in T_{train}.

The LoFi phase functions as a curriculum: the agent first masters the broad geometry of aircraft encounters under simplified dynamics, then fine-tunes to the richer HiFi physics. Carrying over the replay buffer lets off-policy methods keep sampling LoFi transitions during the early HiFi updates, providing a blended data distribution that smooths the domain shift. On-policy algorithms, which discard past trajectories at each optimization step, face a cold start in the new reward landscape and their policy gradients become correspond-

ingly noisier. This "easy-to-hard" continuation is precisely the mechanism highlighted by curriculum-learning theory [30]. In order to avoid this cold start for on-policy algorithms, interleaved-fidelity roll-outs could be employed, whereby on-policy algorithms train in the LoFi environment with decaying probability such that the policy benefits from learning the coarse geometry of the conflict scenario while still receiving some exposition to the richer dynamics. Alternatively, running N parallel Gym environments per A2C/PPO update would allow a smooth shift in batch composition from LoFi-heavy to HiFi-heavy to achieve a similar effect. The formulation studied here, however, still treats the fidelity change as a train-time adaptation problem. In realistic ATM deployment the harder problem is test-time tuning under explicit safety constraints, where traffic characteristics or protected separation minima may change without the possibility of full retraining. Combining mixed-fidelity pre-training with online safety layers, constrained policy optimization, or shielded updates would allow the HiFi-trained policy to be adjusted in situ rather than only offline [13].

Notably, while A2C's evaluated performance ($\bar{R}_{eval}$) was poor relative to SAC and TD3, the transfer penalties observed were the smallest of all trialed RL algorithms. This is a substantial discovery that warrants further research, especially as RL for ATM scenarios look to real world deployment. A2C's robustness with regards to environmental fidelity could make it a prime candidate for RL applications looking to minimize performance dips upon deployment.

A related limitation is the absence of per-algorithm hyperparameter tuning at each fidelity level. Prior work has shown that reported deep-RL performance can vary noticeably with minor hyperparameter changes [23], so some of the absolute gaps reported here may partly reflect these choices. At the same time, this highlights a practical advantage of off-policy MiFi training: most of the expensive hyperparameter search could be run in LoFi, and only the selected configuration continued in HiFi, thereby lowering wall-clock time and energy use while retaining the benefits of the replay buffer.

VI. Conclusion

This paper examined whether mixed-fidelity (MiFi) training can deliver high-quality aircraft conflict-resolution policies at lower computational cost than pure high-fidelity (HiFi) training. Agents first learned in a rapid, instant-action simulator, then completed training in the BlueSky HiFi environment across seven random seeds and six LoFi/HiFi splits.

It was found that MiFi training can potentially enhance the final performance for off-policy RL algorithms, whilst reducing the computational load. This was most obvious for SAC (24.4% better performance, 20.0% less training time) and also observed for TD3 (26.1% better performance, 17.7% less training time), and DDPG (36.9% better performance, 15.9% less training time). On-policy algorithms either stayed relatively constant in their split-dependent performance, or degraded significantly as was observed with PPO.

The strong transfer penalties observed upon environment transfer in off-policy algorithms supports the hypothesis, when paired with the superior performance, that the replay buffer mediates the transfer allowing for better recovery from a large transient drop in reported reward. Together, these findings support the notion that generalization of LoFi-trained conflict-resolution RL policies in HiFi environments is not only possible but can result in improved performance *if the replay buffer is also transferred to the new simulator*.

The cross-seed stability in SAC behavior and response to MiFi training marks this algorithm as predictable. Using the fidelity delta between simulators as a proxy for the difference between HiFi simulation and real-world applications, we can posit that although SAC will reliably incur a strong transient performance loss, it has a high propensity for retraining and recovery.

Although a significant number of random seeds (7) were employed to conduct this RL research, the principles of this paper should be extended to even more seeds (e.g $N \geq 25$) in order to not only reinforce confidence in the above remarks, but also to verify whether LoFi training can be a reliable predictor of HiFi performance. Additionally, the proposed mechanism behind off-policy outperformance should be verified in future work through ablation of the models' replay buffer upon environment switch. Furthermore, simulator/environment fidelity is a nebulous and poorly-defined term. The ability to quantify the degree of simulator fidelity for a given RL application should be investigated. Finally, it is suggested that the performance improvements observed in off-policy algorithms could be replicated for on-policy using interleaved training curricula between LoFi and HiFi simulators, or even the use of blended-fidelity batches.

Future work should also attempt to disentangle how much of the observed MiFi gain is due to entropy-regularized off-policy learning, and should include a targeted hyperparameter sensitivity study in LoFi with transfer to HiFi, as well as test-time, safety-aware adaptation to better match operational ATM requirements. For aircraft conflict-resolution applications, MiFi has shown itself, however, to be a promising avenue of RL research that can not only decrease the number of expensive simulator calls during the training process, but also results in higher-performing policies in high-fidelity environments for off-policy single-agent RL models.

VII. Code Availability

The environments and high-level wrappers are available online via: https://github.com/amoec/ATC_FlexRL.git

References

[1] J. Hong and A. Palazzo, "Air Travel Is Back to Pre-Pandemic Levels With New Turbulence Ahead," *Bloomberg.com*, Oct. 2023. [Online]. Available: https://www.bloomberg.com/news/articles/2023-10-08/air-travel-finally-reaches-pre-covid-l-evels-but-profits-suffer

[2] IATA, "Global Air Passenger Demand Reaches Record High in 2024," Jan. 2025. [Online]. Available: https://www.iata.org/en/pressroom/2025-releases/2025-01-30-01/

[3] D. Shepardson, "Panel to review US air traffic controller fatigue after near-miss incidents," *Reuters*, Dec. 2023. [Online]. Available: https://www.reuters.com/business/aerospace-defense/panel-review-us-air-traffic-controller-fatigue-after-near-miss-incidents-2023-12-20/

[4] E. Steel and S. Ember, "Drunk and Asleep on the Job: Air Traffic Controllers Pushed to the Brink," *The New York Times*, Dec. 2023. [Online]. Available: https://www.nytimes.com/2023/12/02/business/air-traffic-controllers-safety.html

[5] S. Ember, E. Steel, L. Abraham, E. Lutz, and E. Koeze, "Airline Close Calls Happen Far More Often Than Previously Known," *The New York Times*, Aug. 2023. [Online]. Available: https://www.nytimes.com/interactive/2023/08/21/business/airline-safety-close-calls.html

[6] D. Enrich, "Before Crash, an Alarming Pattern of Near-Misses Between Planes," *The New York Times*, Jan. 2025. [Online]. Available: https://www.nytimes.com/2025/01/31/us/plane-crash-near-misses-airlines.html

[7] EUROCONTROL, "Why artificial intelligence is highly relevant to air traffic control," Nov. 2019. [Online]. Available: https://www.eurocontrol.int/article/why-artificial-intelligence-highly-relevant-air-traffic-control

[8] Z. Wang, W. Pan, H. Li, X. Wang, and Q. Zuo, "Review of Deep Reinforcement Learning Approaches for Conflict Resolution in Air Traffic Control," *Aerospace*, vol. 9, no. 6, p. 294, Jun. 2022, number: 6 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2226-4310/9/6/294

[9] M. Brittain and P. Wei, "Autonomous Separation Assurance in An High-Density En Route Sector: A Deep Multi-Agent Reinforcement Learning Approach," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. Auckland, New Zealand: IEEE, Oct. 2019, pp. 3256–3262. [Online]. Available: https://ieeexplore.ieee.org/document/8917217/

[10] TU Delft Control & Simulation, "Bluesky." [Online]. Available: https://github.com/TUDelft-CNS-ATM/bluesky.git

[11] C. A. Badea, D. J. Groot, A. M. Veytia, M. Ribeiro, J. Ellerbroek, J. Hoekstra, and R. Dalmau, "Lateral and Vertical Air Traffic Control Under Uncertainty Using Reinforcement Learning," 2022.

[12] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor," in *Proceedings of the 35th International Conference on Machine Learning*. PMLR, Jul. 2018, pp. 1861–1870, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v80/haarnoja18b.html

[13] J. García and F. Fernández, "A Comprehensive Survey on Safe Reinforcement Learning," *Journal of Machine Learning Research*, vol. 16, no. 42, pp. 1437–1480, 2015. [Online]. Available: http://jmlr.org/papers/v16/garcia15a.html

[14] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Vancouver, BC: IEEE, Sep. 2017, pp. 23–30. [Online]. Available: http://ieeexplore.ieee.org/document/8202133/

[15] M. Cutler, T. J. Walsh, and J. P. How, "Real-World Reinforcement Learning via Multifidelity Simulators," *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 655–671, Jun. 2015. [Online]. Available: https://ieeexplore.ieee.org/document/7106543

[16] S. Bhola, S. Pawar, P. Balaprakash, and R. Maulik, "Multi-fidelity reinforcement learning framework for shape optimization," *Journal of Computational Physics*, vol. 482, p. 112018, Jun. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999123001134

[17] B. Li, R. Basu Roy, D. Wang, S. Samsi, V. Gadepally, and D. Tiwari, "Toward Sustainable HPC: Carbon Footprint Estimation and Environmental Implications of HPC Systems," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, ser. SC '23. New York, NY, USA: Association for Computing Machinery, Nov. 2023, pp. 1–15. [Online]. Available: https://dl.acm.org/doi/10.1145/3581784.3607035

[18] European Commission, "2050 long-term strategy." [Online]. Available: https://climate.ec.europa.eu/eu-action/climate-strategies-targets/2050-long-term-strategy_en

[19] M. Towers, A. Kwiatkowski, J. K. Terry, J. U. Balis, G. de Cola, T. Deleu, M. Goulão, A. Kallinteris, M. Krimmel, A. KG, R. Perez-Vicente, A. Pierré, S. Schulhoff, J. J. Tai, H. J. S. Tan, and O. G. Younis, "Gymnasium: A Standard Interface for Reinforcement Learning Environments." [Online]. Available: https://github.com/Farama-Foundation/Gymnasium

[20] C. A. Badea, D. J. Groot, A. M. Veytia, M. Ribeiro, and R. Dalmau, "jangroter/atcenv at Master," 2022. [Online]. Available: https://github.com/jangroter/atcenv

[21] "Gymnasium Documentation." [Online]. Available: https://gymnasium.farama.org/index.html

[22] A. Moëc, "amoec/ATC_flexrl," Jul. 2025, original-date: 2025-02-19T15:33:46Z. [Online]. Available: https://github.com/amoec/ATC_FlexRL

[23] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep Reinforcement Learning That Matters," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, Apr. 2018. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/11694

[24] D. J. Groot, G. Leto, S. Vlaskin, A. Moec, and J. Ellerbroek, "BlueSky-Gym: Reinforcement Learning Environments for Air Traffic Applications," *SESAR Innovation Days 2024*, p. 1, 2024. [Online]. Available: https://doi.org/10.61009/SID.2024.1.10

[25] DLR, "Policy Networks — Stable Baselines3 2.6.1a1 documentation." [Online]. Available: https://stable-baselines3.readthedocs.io/en/master/guide/custom_policy.html

[26] C. R. Harris, K. J. Millman, S. J. Van Der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. Van Kerkwijk, M. Brett, A. Haldane, J. F. Del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. [Online]. Available: https://www.nature.com/articles/s41586-020-2649-2

[27] M. O'Neill, "PCG: A Family of Simple Fast Space-Efficient Statistically Good Algorithms for Random Number Generation," Harvey Mudd College, Technical HMC-CS-2014-0905, Sep. 2014.

[28] Delft High Performance Computing Centre (DHPC), "DelftBlue Supercomputer (Phase 2)," https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase2, 2024.

[29] V. Joshi, Z. Xu, B. Liu, P. Stone, and A. Zhang, "Benchmarking Massively Parallelized Multi-Task Reinforcement Learning for Robotics Tasks," 2025.

[30] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th Annual International Conference on Machine Learning*. Montreal Quebec Canada: ACM, Jun. 2009, pp. 41–48. [Online]. Available: https://dl.acm.org/doi/10.1145/1553374.1553380