
Cognitively Constrained Partners for Zero-Shot Coordination in Real-Time Partitioned Control

Ash Simmons

University of Exeter, UK
as1798@exeter.ac.uk

George De Ath

University of Exeter, UK
g.de.ath@exeter.ac.uk

Richard M. Everson

University of Exeter, UK
r.m.everson@exeter.ac.uk

Abstract

Effective human-machine teaming requires agents that coordinate with unpredictable human partners. Reinforcement learning from live human feedback or large behavioural datasets is expensive, while purely simulated training overfits to artificial conventions and fails at Zero-Shot Coordination with real people. We propose a framework for training teaming agents against Cognitively Constrained Partners: simulated partners whose diversity emerges from procedural constraints grounded in human physiological literature, specifically reaction delays and perceptual limitations, rather than from reward shaping or historical training checkpoints. In a modified Lunar Lander environment, we report preliminary evidence that these partners play more like real people than unconstrained agents do, and that agents trained against them match the strongest cross-play baselines while being the only ones whose performance does not depend on how their training partners were generated. We also describe a human-participant study designed so that its rankings can be compared directly against cross-play, testing how well that standard evaluation predicts real human-agent teaming.

1 Introduction and Motivation

Effective human-machine teaming (HMT) requires autonomous agents which not only effectively maximise a reward function in isolation, but maintain optimality when coordinating with unpredictable human partners. Traditionally, training Reinforcement Learning (RL) agents for cooperation relies on human-in-the-loop feedback or extensive behavioural datasets; these approaches are often prohibitively expensive and create operational bottlenecks.

To bypass this, recent approaches pivot to purely simulated training, such as multi-agent or automated population play. These generate diverse policies efficiently and without human data, but, lacking human grounding, consistently overfit to arbitrary artificial conventions. Consequently, when these agents are deployed alongside real human partners under the conditions of Zero-Shot Coordination (ZSC) – collaboration with a novel partner without prior interaction – their brittle policies tend to collapse. Contemporary methods bridge this gap either by cloning human datasets, shifting the cost of collection to pre-training, or by drawing diversity from self-play checkpoints [36], whose logic still differs fundamentally from a human teammate’s.

In continuous settings, human-machine teaming is usually posed as shared control, with the human and the agent holding overlapping authority over the same task space [32, 41, 30]. Blended assistance and corrective nudges reduce error effectively, but cast the agent as a subordinate safety net rather than a partner. Dividing the task instead runs into boundaries that are traded dynamically: either unwilling to bear the cognitive overhead of predicting the agent or lacking trust in its ability, human operators reclaim large portions of the task and circumvent the teaming process entirely [7, 17], reducing a multi-agent system to a single-actor workflow.

Addressing these challenges, this paper presents an RL architecture deployed within a time-continuous, strictly partitioned control environment, a domain which remains under-explored as an RL benchmark for ZSC. By enforcing a rigid division of control wherein the human and the agent operate over mutually exclusive degrees of freedom, the framework eliminates the possibility of human domination while preventing the agent from being reduced to a purely assistive tool. This forced interdependence requires both partners to continuously coordinate their distinct inputs, facilitating genuine co-execution analogous to that required in real-world systems such as search and rescue, air traffic control, and robotic surgical assistance.

To achieve this without the vast datasets required in behavioural cloning (BC) or the brittle assumptions of Fictitious Co-Play (FCP), we propose a framework centred on partners limited by the same cognitive constraints as people. Teaming performance is evaluated against Multi-Agent Reinforcement Learning (MARL), Population Based Training (PBT) [15], and FCP [36] baselines, both under cross-play with held-out partners and, in a forthcoming study, alongside human players. This pairing tests whether cross-play, the standard measure of ZSC, actually predicts coordination with real people.

2 Related Work

While continuous partitioned control has been modelled extensively in physical systems, traditional solutions rely on computationally rigid frameworks. Javdani et al. [16] managed continuous interdependence in a high-degree-of-freedom robotic arm by formulating shared autonomy as a Partially Observable Markov Decision Process, but this requires exhaustive prior knowledge of the transition dynamics and a predefined set of goals, restricting scalability where ad-hoc coordination is required. Reddy et al. [30] instead demonstrated model-free Deep RL for continuous shared autonomy in a Lunar Lander environment, assisting operators without prior knowledge of the dynamics or the user’s intent. That work focused on blended control, where the agent may override or modify the human’s inputs to ensure safety; our architecture instead enforces strict partitioned control, preventing the agent from overriding suboptimal human play.

Self-play and population play agents coordinate well with one another, but only because each is accustomed to partners performing at a similar level of optimality to itself. Carroll et al. [4] showed that anchoring training in real human data prevents the formation of brittle policies: using an Overcooked environment, they collected paired human trajectories and applied BC to construct a ‘human proxy’ policy. Training an RL agent alongside this suboptimal partner forced it to account for arbitrary human conventions, delays and errors. This prepares agents for more effective ZSC, but does not address the volume of human data required to do so.

Although diverse agent pools have been established to improve performance in population-based training, ensuring that diversity presents further challenges. Lupu et al. [22] addressed this with Trajectory Diversity (TrajeDi), adding an objective rewarding agents for behavioural difference from every other population member alongside the expected environmental reward; in Hanabi, agents best-responding to a TrajeDi population outperformed self-play agents under ZSC with novel partners. Strouse et al. [36] translated such principles to human-AI teaming: their FCP framework achieved robust ZSC with live human partners in Overcooked without any human-in-the-loop training data, constructing a diverse population from historical checkpoints of self-play agents at various stages of learning.

The problem of coordinating with unknown teammates was formalised as ad hoc teamwork by Stone et al. [34]: collaborating without prior opportunity to establish shared conventions. Hu et al. [13] demonstrated why self-play agents fail under these conditions, converging on arbitrary conventions a novel partner has no means of anticipating, and proposed Other-Play to avoid them. Laidlaw and Dragan [18] instead model the partner directly, characterising human decision-making as noisily rational and deriving a distribution over plausible policies to best-respond against. More recent population-based methods broaden the training distribution itself, either rewarding agents for developing policies incompatible with those of their peers [5] or maximising the entropy of the population as a whole [42].

Together these establish that populations diverse enough to prepare agents for ZSC can be generated without exhaustive human datasets or manual tuning. But where TrajeDi forces artificial trajectory divergence and FCP relies on the transient artefacts of under-trained networks, we pivot the source of population variance to procedural constraints: a training pool of partners subject to unique cognitive

processing delays and perceptual limitations, generating trajectory diversity while ensuring the teaming agent adapts to a spectrum of psychologically valid, bounded behaviours.

3 Partitioned Lunar Lander Environment

To address the gap in continuous interdependence identified above, we utilise a modified continuous Lunar Lander environment from Gymnasium [8]. The core objective requires landing safely within a designated target zone using a primary vertical thruster and two lateral thrusters (Figure 1). Episodes initialise with random top-screen spawns and terminate via landing or crashing. This environment offers enough complexity to capture the nuance of continuous control demands found in real-world HMT tasks, yet remains computationally tractable enough to isolate the cooperative dynamics from the agent’s baseline proficiency in the environment’s physics.

Unlike previous adoptions of this environment for teaming tasks, here the action space is decomposed. By assigning authority over the primary engine and lateral thrusters to distinct partners, the environment precludes unilateral task completion. This structural division mandates genuine real-time collaboration, requiring continuous coordination across the joint action space.

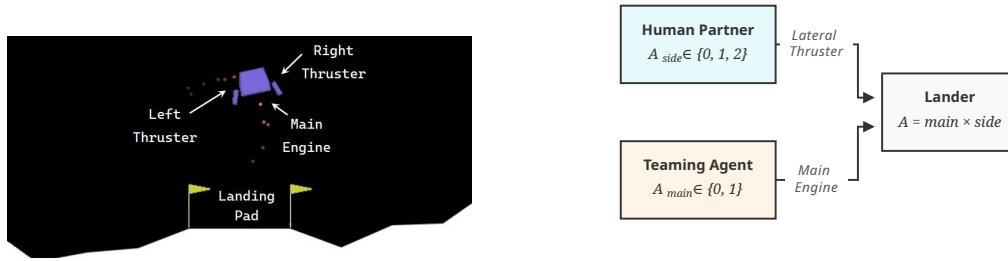


Figure 1: Lunar Lander environment from Gymnasium [8]. Physical geometry (left) and the partitioned control pipeline (right). The main engine and side thrusters must be coordinated to reach the landing pad.

The state space is the standard 8-dimensional vector $\mathcal{S} = [x, y, \dot{x}, \dot{y}, \theta, \dot{\theta}, l, r]$, giving the coordinates, linear velocities, angle relative to the vertical axis, angular velocity, and contact indicators for the left and right legs. Neither of the environment’s built-in control schemes suits split control. The discrete scheme offers four mutually exclusive actions, doing nothing or firing one of the left, main and right thrusters, so the two roles cannot act on the same frame. The continuous scheme takes upward and sideways forces on $[-1, 1]$, requiring precision no player with a keyboard can supply. To allow the two parties to act concurrently we implement a multi-discrete wrapper over the continuous environment, with $\mathcal{A} = \mathcal{A}_{\text{main}} \times \mathcal{A}_{\text{side}}$, where $\mathcal{A}_{\text{main}} = \{0, 1\}$ sets the main engine state (Off/On) and $\mathcal{A}_{\text{side}} = \{0, 1, 2\}$ the lateral thrusters (Off/Left/Right). Each discrete selection maps to a fixed magnitude in the underlying continuous space: the main engine fires at full power, while the lateral thrusters fire at 70% of their maximum, full lateral thrust proving excessive for the scale of correction this task requires.

Both partners share a single reward, derived from the environment’s shaping potential and charged per frame for each thruster in use, so that at each step the agent receives

$$R_t = \Phi(s_t) - \Phi(s_{t-1}) - 0.3 u_{\text{main},t} - 0.021 u_{\text{side},t}, \quad (1)$$

where $u_{\text{main},t}, u_{\text{side},t} \in \{0, 1\}$ indicate whether each thruster is firing and Φ is that potential, which penalises distance from the pad, speed and tilt and rewards leg contact [8]. The lateral cost is below the nominal 0.03 because of this reduced power (Appendix A.1). On the terminal frame the reward is replaced by $+100 - P(d)$ for a landing and -100 for a crash or for leaving the screen, with truncated episodes receiving neither. Since this reward permits landing anywhere on the terrain, whereas players naturally aim between the flags, a placement penalty realigns the two: with d the horizontal distance from the centre of the landing pad, $P(d) = 0$ for $d \leq 0.2$ and $P(d) = 25 + 93.75(d - 0.2)$ otherwise, so landing just outside the zone costs 25 points, rising to 100 at the edge of the screen.

Reward shaping was considered for generating a diverse human-like population, forcing biases towards risk-taking, fuel economy or caution. Although this has improved teaming under ZSC with

human partners [22, 25, 28], shaping explicitly to force diversity encodes designer bias, driving agents into artificial niches rather than capturing authentic human variance. We instead retain the standard incentives otherwise unaltered, so the constrained agents optimise the same objective as people. Diversity is located in computational rationality [10], in which behaviour is utility maximisation under cognitive and physical limits. Observable constraints such as sensory latency and motor bottlenecks are also more empirically tractable than unobservable latent reward preferences.

4 Cognitively Constrained Partners

Cognitive biases and heuristic thinking inherent to human reasoning have long been seen as evidence that human thinking is far from optimal [38], yet recent developments support the notion that human cognition is best understood as the optimal use of limited computational resources [20]. Building upon this, the constrained agents are designed to generate human-like behaviour by optimising within physiologically grounded constraints. Their policies should converge towards the hesitant, temporally chunked decision-making of human gameplay. To achieve this, a psychological wrapper is applied to the agent’s environment which limits reaction speed, ability to perceive changes, and action frequency.

The agent’s decision-making pipeline is deconstructed into three distinct phases: processing (thinking) time, mechanical delay, and a refractory (mental recovery) period. Since human reaction times consistently exhibit a positive skew due to attentional fluctuations, intra-episode delays for each phase are drawn from an exponentially modified Gaussian distribution, which serves as the standard mathematical model for reaction time variance within individuals [40]. To generate a diverse population of unique constrained agents, the mean (μ) for each delay category is randomly initialised from a uniform distribution covering the spectrum of reaction speeds expected of real human players. The simulated environment runs at 50 frames per second, meaning that each frame of the game lasts 20 ms. The processing delay, representing the central nervous system’s perceptual and cognitive cycle, is therefore initialised with a mean uniformly between 4.0 and 5.5 frames in duration. This translates to an 80–110 ms window, aligning with the Model Human Processor [3], which empirically establishes the human perceptual-cognitive cycle time at approximately 110 ms.

Similarly, the mechanical delay, representing the neuromuscular lag required to actuate a control input, is initialised with a mean between 3.0 and 4.0 frames which maps to the 70 ms motor processor cycle defined in the same human-computer interaction framework [3]. When combined, the processing and mechanical delays range from 140–190 ms. This range of delays is also validated by the biological constraints observed in continuous human manual tracking and control tasks [23]. Finally, following the execution of an action, the agent is subject to a refractory period sampled with a mean between 6.0 and 7.5 frames (120–150 ms). This models the known psychological refractory period bottleneck observed when human players attempt to process sequential stimuli or rapidly switch tasks [1, 35].

Across all three phases the standard deviation (σ) is held small (0.6/0.5/1.0), isolating low-level physiological jitter so that sensory-motor transmission remains tight. Distributional shape is instead governed by the exponential parameter (τ). The processing and refractory phases carry a heavy tail ($\tau = 1.5$), modelling the vulnerability of complex decision-making and mental recovery to micro-distractions [39], while the mechanical phase is near-symmetric, reflecting the reflex-driven nature of muscle actuation once a decision is finalised ($\tau = 0.5$). The distributions for all three phases, and for the convolved processing-plus-mechanical latency, are plotted in Appendix A.2.

At the start of each episode, and whenever the agent commits to a new input, delays are drawn for each category. Releasing an input does not trigger this, reflecting the lower cognitive cost of ceasing an action relative to initiating one. Rather than receiving live observations, the agent is fed from a buffer in line with its current processing delay. Following the refractory period, the agent can perform an action, which is sent to the environment after its mechanical delay period. Actions must persist for at least four frames, reflecting the minimum human input duration of 80 ms [27].

The delays make the problem partially observable, so a Recurrent PPO network [33] was chosen over a typical feedforward approach. As well as mirroring human working memory, the hidden state within acts as a continuous cognitive buffer, allowing integration of sequential observations over time to bridge the temporal blind spots caused by the imposed reaction delays. Consequently, the agent generates temporally consistent inputs that replicate human goal-oriented behaviour, rather than reacting blindly to isolated frames.

Beyond reaction delays, the wrapper restricts the agent’s ability to detect small changes in velocity and orientation. While unconstrained agents register state changes instantaneously, human perception is bounded by Weber’s law, which holds that the minimum detectable change in a stimulus is a constant proportion of that stimulus. This is true for human observation of both velocity [2] and angular rotation [6]. Three channels of the state vector are subject to this filter: the horizontal and vertical velocities, and the tilt angle. Each channel q maintains its own record q_{last} of the value at which it was last registered. Where the difference falls short of the agent’s threshold, q_{last} is held in place of the true value and the record left unchanged; where it meets the threshold, the change is registered and the record updated. Since comparison is against q_{last} rather than the preceding frame, a change too gradual to notice on any single frame accumulates until it crosses the threshold, so no quantity can drift arbitrarily far from what the agent believes it to be. The remaining channels pass through the processing delay unfiltered.

A strict proportional rule cannot be applied directly: Weber’s law breaks down at low stimulus intensities, where a constant fraction implies unbounded sensitivity and would grant the agent superhuman acuity during the precise landing phase. Each threshold is therefore bounded below, and the tilt threshold is further reduced near the cardinal angles, reflecting the greater sensitivity humans show to rotations close to vertical and horizontal, known as the ‘oblique effect’ [9]. These departures leave the constrained agents less sensitive than a literal reading of Weber’s law would make them, not more. The thresholds are given in full in Appendix A.3.

To diversify the population, each agent is assigned a unique perceptual sensitivity before training. A single scalar $p \sim \mathcal{U}(0, 1)$ is drawn and mapped linearly onto both Weber fractions, spanning the 4.5–8% range reported for angle judgement [6] and the 7–11% range reported for velocity discrimination [2]. Linking both thresholds to a shared variable mechanically couples the perceptual traits, ensuring that an agent with acute spatial awareness also possesses above-average velocity tracking, effectively simulating a generalised visual aptitude.

5 Teaming Agent Implementation and Baselines

5.1 Teaming Agent

Solo agents are first trained without the psychological wrapper to learn the environment’s physics, while the cognitively constrained partners are trained alone from scratch on the full action space, each under an independently sampled wrapper with its own reaction-time and perceptual constraints. The task is then split: the constrained partner, its policy now fixed, contributes only its lateral output, while the teaming agent controls the main engine, its action space reduced to $\mathcal{A}_{\text{main}} = \{0, 1\}$ (Off/On). The teaming agent is initialised by partial weight transfer from a solo agent, so it begins already able to fly and spends its training budget learning to integrate its partner’s behaviour. Since neither party observes the other’s forthcoming decision, the partition adds a second source of partial observability alongside the partner’s delays, and the recurrent architecture is retained for that reason.

At each time step the partner is queried for its lateral input, subject to the same delays, refractory period and minimum press duration under which it was trained, and this input, together with the teaming agent’s main-engine action, steps the environment forward. The partner’s *executed* action a_{side} (which, owing to the motor pipeline, is frequently not the one its policy requested) is appended as a ninth observation for the teaming agent’s subsequent timestep. Each episode the teaming agent faces a partner drawn uniformly from a 45-agent pool, each with distinct delay and perceptual constraints. Appendices B.1 and B.2 give the full hyperparameters and the transfer procedure.

5.2 Baselines

Despite significant advancements in ZSC for HMT tasks, the RL literature predominantly focuses on tasks that are either symmetric in their action spaces or rely on discrete, turn-based mechanics where time is afforded before deciding the next action. In contrast, our strictly partitioned continuous control task demands real-time, reflex-driven coordination, representing a class of environments that remains significantly under-explored. Consequently, there is to our knowledge no single benchmark tailored for this specific problem domain, and we compare instead against three approaches adapted to this setting: FCP as the primary baseline, alongside PBT and standard MARL.

Adapting these methodologies to the partitioned control scheme of this environment necessitates a structural departure from their traditional implementations. First, concurrent learning is avoided wherever the comparison permits. Simultaneous gradient updates across the disjoint main and lateral action spaces induce severe environmental non-stationarity [21], and, more consequentially for ZSC, mutually adapting agents converge on interdependent joint policies. This ‘secret handshake’ [31, 14] is a convention a novel human partner does not share [36, 4]. PBT and FCP therefore compute a Best Response against frozen partners, stabilising the transition dynamics into a stationary MDP and forcing the active agent to carry the burden of adaptation, as it must at deployment. MARL retains co-adaptation deliberately, since unconstrained joint optimisation is what that baseline exists to represent; it uses a staggered rather than simultaneous update schedule to control gradient instability.

To ensure equal learning opportunity across frameworks, all training pools are matched at 45 partners: 45 cognitively constrained partners, 15 FCP lineages sampled at three skill tiers, and a 46-agent PBT population in which each learner trains against the 45 others. Each condition is subsequently evaluated against a further 50 partners disjoint from their training pools. Matching these pool sizes ensures the comparison isolates the source of partner diversity rather than its quantity.

The information available to each partner necessarily differs between conditions. The MARL partner observes its teammate’s action, as mutual co-adaptation requires it, whereas the partners used by FCP, PBT and our own method receive only the physical state (plus, for the constrained partners, an efference copy of their own input, Appendix A.4). This distinction is deliberate: a partner given a noiseless per-frame feed of its teammate’s action would be considerably more reactive than any person, and an agent best-responding to such a partner would come to depend on assistance a human will not provide. Implementation details for each baseline are given in Appendix A.5.

6 Preliminary Results

6.1 Behavioural Validation

This section provides a preliminary validation of the constrained partners against human play, based on a small informal pilot whose results are indicative rather than conclusive. Twenty participants from the authors’ research department and their colleagues, all new to the game, each played 10 solo episodes on 10 fixed seeds, which were subsequently played by 45 cognitively constrained and 45 unconstrained agents. The unconstrained agents were measured once; since they are deterministic, repeated rollouts would produce identical behaviour given the use of fixed seeds. Conversely, the delays applied to the cognitively constrained agents are stochastic, so each agent repeated every scenario three times so as not to under-sample the ex-Gaussian distributions. Solo play was measured rather than cooperative play, as the absence of a teammate ensures that any observed difference is attributable to the controller alone.

Twenty-two features were extracted from each episode, characterising control behaviour as well as performance. They cover how long and how often each control is pressed and how variable those presses are, how input is divided between each of the thrusters, how quickly the controller responds to tilt, how the craft moves during descent, and the state in which the episode ends. These are all expressed as rates or durations rather than counts, to avoid confounding by episode length, and each is explained in Appendix D. Episodes were averaged per controller, as those produced by the same controller constitute repeated sampling of a single policy rather than independent observations.

Divergence between controller populations was measured by energy distance [37]:

$$\mathcal{E}(X, Y) = 2 \mathbb{E}|X - Y| - \mathbb{E}|X - X'| - \mathbb{E}|Y - Y'|, \quad (2)$$

where X and X' are two controllers drawn at random from one population, and Y and Y' from the other. The first term measures how far apart the two populations are, and the others subtract each population’s internal spread, so a highly variable population is not mistaken for a distant one. $\mathcal{E}$ is 0 only for identical distributions and grows as they separate, so a lower distance from the human sample means an agent population whose play is harder to distinguish from that of real players. As $\mathcal{E}$ has no absolute scale, each value is read against the distance between two random halves of the human sample. Over 2,000 random splits a distance of 1.19 (95% CI [0.57, 3.03]) was found between the two human populations, so a genuinely indistinguishable population would be expected to fall within this interval.

Following feature standardisation, human players lay 4.12 from the constrained agents and 7.41 from the unconstrained, a difference of +3.29 (95% CI [+1.64, +4.86]) over 5,000 bootstrap resamples of controllers, with the constrained agents closer in 99.9% of samples. The constrained agents therefore lie roughly 3.5 times further from human players than human players lie from one another, against 6.2 times for the unconstrained agents. Since the human sample is small, with only 10 players on each side of a split, the human-to-human distance varied widely across the 2,000 resamples, which limits how precisely these distances can be estimated; the result therefore supports the ordering of the two agent populations relative to human players more firmly than any claim about how close either lies to them.

Energy distance shows whether populations differ but not where, so three measures are also reported per feature: the difference in group means, the Wasserstein distance normalised by group spread (lower is closer), and the overlap in probability mass (higher is closer), with definitions and limitations given in Appendix D. The constrained agents lie closer to human play on 15 of the 22 features under each measure independently, and the three measures agree on 17 (Tables 3 and 4).

Mean press duration for the side thrusters is the closest match of any feature, sharing 73% of its probability mass with participants where the unconstrained agents share none, so the constraints reproduce human input lengths almost exactly (Figure 2). The main-engine presses are closer than the unconstrained agents manage but still short, at 353 ms against 98 ms, with participants averaging 562 ms. Participants also vary widely in how long they hold the main engine (SD 171 ms) where the constrained agents cluster tightly (SD 49 ms), so the constraint lengthens presses towards the human range without reproducing the variation between players.

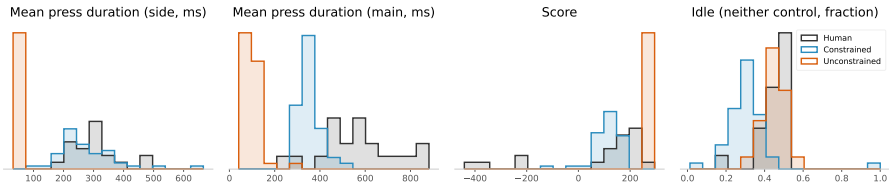


Figure 2: Individual controller distributions for mean press duration (side and main), score, and idle time (neither control) across human, constrained, and unconstrained groups.

Participants pressed the lateral thrusters 0.94 times per second and the constrained agents 1.62, against 6.12 for the unconstrained agents; for bursts of three or more direction changes within half a second the rates are 0.43, 1.39 and 9.73. No player could sustain the rate the unconstrained agents produce, and the constraints bring the agents to within a small multiple of human behaviour where they would otherwise be an order of magnitude adrift. All four terminal measures (displacement from the centre, speed, tilt and score) also favour the constrained agents, with terminal tilt the highest overlap of any feature at 78%, although no constraint pertains to how an episode should end.

Five features favour the unconstrained agents under all three measures, all concerning idle time, directional preference, or tilt-correction latency. The unconstrained agents are almost indistinguishable from participants in the proportion of frames with the side thrusters inactive (0.714 against 0.696, versus 0.565 for constrained) and with neither control active (0.446 against 0.452, versus 0.311). Participants also show a directional bias in their lateral input (0.34) which the unconstrained agents partially reproduce (0.17) and the constrained do not (0.10), and correct tilt more slowly (286 ms) than either the constrained (131 ms) or unconstrained (146 ms) agents, the latter marginally closer.

6.2 Cross-Play Generalisation

Partners are grouped into four families by the procedure that generated them (constrained, PBT, FCP and MARL), and each teaming agent’s matched family is the one it trained against, the other three being held out. Figure 3 shows the mean return for every agent against every family. Each condition is represented by ten independently trained replicates rather than a single agent, since one training run conflates a method’s behaviour with the outcome of that particular run; every replicate plays fifteen partners from each family on twenty fixed seeds, twice, giving 6,000 episodes per cell and 96,000 in total. Every partner in every pool is drawn from an evaluation set disjoint from the training pools of all four methods, so no agent has met any of these partners before, bar the exception noted in Figure 3. The diagonal therefore does not pair an agent with partners it trained against, but with unseen partners

produced by the same generative procedure; the off-diagonal cells pair it with partners produced by a procedure it has never encountered, which is the situation a human teammate presents.

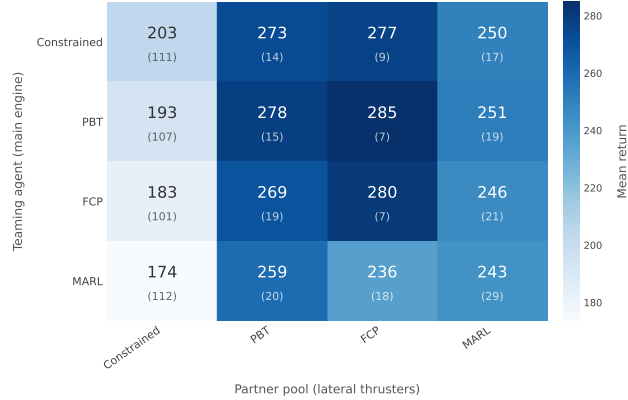


Figure 3: Mean return of each agent-partner pairing, averaged over 10 independently trained replicates per agent type. Parenthesised figures are the standard deviation across the 15 partners comprising each pool. One of the fifteen MARL side agents is the co-adapted partner of each MARL replicate.

The clearest result is what happens when an agent moves away from the procedure it was trained against. Table 1 reports each agent’s return against its matched pool and against the three it did not train alongside. The constrained agent is the only one to improve, rising from 203.4 to 266.6; PBT falls by 34.9 points and FCP by 47.6, both intervals excluding zero, while MARL’s smaller decline of 19.9 is not distinguishable from zero. Because the comparison is paired by replicate, the strength of any individual training run cancels from the difference rather than adding noise to it. PBT and FCP agents thus depend on the procedure that produced their partners, in a way that agents trained against cognitively constrained partners do not.

Table 1: Mean return when each agent is paired with partners from the pool generated by the procedure it trained against and from the three pools it did not train against. Intervals are percentiles of a two-way cluster bootstrap over 5,000 resamples of trained replicates and partners, paired by replicate so that the strength of a given training run cancels from the difference.

Teaming agent	Matched pool	Held out	Change	95% CI
Constrained	203.4	266.6	+63.1	[+23.1, +125.3]
PBT	277.9	242.9	−34.9	[−58.3, −15.8]
FCP	280.1	232.6	−47.6	[−70.7, −28.9]
MARL	242.7	222.7	−19.9	[−51.1, +5.5]

That property is not accompanied by an advantage in raw performance. Restricting the between-agent comparison to families neither agent trained alongside, so that both are averaged over exactly the same thirty partners, the constrained agent is indistinguishable from PBT (−4.5, 95% CI [−11.4, +1.9]) and from FCP (+4.0, [−1.8, +10.0]), and ahead of MARL (+27.5, [+8.7, +52.7]; Table 5). Averaged across all four pools the three population-based methods are close, at 250.8 for the constrained agent, 251.7 for PBT and 244.4 for FCP, with MARL at 227.7. Training against delayed and inconsistent partners therefore costs nothing against competent ones.

Every agent performs worst against the constrained pool, which is also far more heterogeneous than the others: the standard deviation across its fifteen members exceeds 100 return points for every agent, where the FCP, PBT and MARL pools fall between 7 and 29. Delayed and perceptually limited partners therefore differ from one another to a degree that conventionally generated populations do not reproduce. Against it the constrained agent returns 203.4, ahead of FCP by 20.8 ([+0.1, +43.7]) and MARL by 29.9 ([+13.0, +50.7]), though not distinguishable from PBT (+10.5, [−7.1, +30.5]). These partners are unseen by every agent, but only the constrained agent trained against the procedure that generated them, so the comparison should be read with that asymmetry in mind.

7 Human Study Design

Cross-play against synthetic partners is the field’s standard proxy for teaming performance, but how well its rankings predict coordination with real people is rarely tested directly. To measure the efficacy of the proposed architecture, and to test whether the cross-play rankings of Section 6.2 persist with human partners, a formal within-subject study that has received ethical approval from the authors’ institution is scheduled to take place. This study will see 100 participants play the game alone, and with each of the four teaming agent types (ten instances of each). Scores will be analysed using a linear mixed-effects model, with the agent as a fixed effect and the participant as a random intercept, so that agents are compared relative to each individual’s own baseline rather than to the group mean. Planned contrasts then compare the proposed agent against each baseline in turn. Full details for the planned study, model and the secondary analyses are set out in Appendix C.

8 Limitations

Humans are not static policies themselves, and adapt to their partners over time; our work focuses specifically on the first instances of cooperation, in which a player approaches the task with a relatively fixed set of cognitive heuristics and physiological constraints. This architecture is intended as a strong starting point for more specialised human-in-the-loop training, reducing the effort needed to reach an agent policy which acts as a best response to novel human partners.

The cross-play evaluation reported in Section 6.2 pairs teaming agents with held-out artificial partners, and consequently demonstrates generalisation across partner types rather than to human players specifically. Although the behavioural validation partially bridges this gap by establishing that our partners occupy a similar behavioural region to real players, the inference from cross-play performance to human teaming performance remains untested until the study is complete.

The constrained pool is the hardest of the four for every agent, so the constrained agent begins from the lowest matched-pool baseline and an improvement is easier to obtain from 203.4 than from PBT’s 277.9 or FCP’s 280.1. The change reported in Table 1 is therefore not directly comparable across agents. The comparison in Table 5, where every agent meets the same partners, is free of this effect, and shows parity with PBT and FCP rather than an advantage over them.

The constrained partners perceive only the physical state, whereas a player would also see the main engine’s exhaust plume. As set out in Section 5.2, this under-models partner awareness deliberately, yielding a cautious policy more analogous to a player concentrating on their own role. A more faithful treatment would grant the partner intermittent awareness of its teammate’s actions, filtered through the same perceptual pipeline as the rest of its observations.

Although restrictions on the constrained agents’ perception of changes in both speed and angle are imposed, these do not cover all channels pertaining to those variables, and it is possible that the agent infers what it has not been allowed to see. Angular velocity in particular passes through unfiltered, so an agent whose tilt perception is hindered still receives a clean signal of the rate at which that tilt is changing, from which the true angle can in principle be integrated. Extending these restrictions to every channel, granting partners filtered awareness of their teammate, and building more specialised human-in-the-loop training on this starting point are all left to future work.

9 Conclusion

This paper has presented a framework in which partner diversity for ZSC is generated procedurally from perceptual and motor constraints grounded in human physiological literature, applied to a partitioned environment in which neither party can complete the objective alone. The framework’s central claim, that agents trained against these partners will coordinate more effectively with human players, awaits the study described in Section 7, but the preliminary results establish the two conditions it depends upon: the partner population occupies a similar region of behavioural space to real players, though without reproducing the full variation between them, and agents trained against it match the strongest baselines under cross-play while losing nothing when the partner-generation procedure changes, where PBT and FCP degrade. Beyond ranking the agents, the study will show whether the orderings produced by cross-play evaluation survive contact with human partners, a direct measure of how well this standard proxy predicts performance in real human-agent teams.

References

- [1] Philip A. Allen, Albert F. Smith, Heli Vires-Collins, and Susan Sperry. The psychological refractory period: Evidence for age differences in attentional time-sharing. *Psychology and Aging*, 13(2):218–229, 1998. doi: 10.1037/0882-7974.13.2.218.
- [2] Robert H. Brown. Weber ratio for visual discrimination of velocity. *Science*, 131(3416): 1809–1810, 1960. doi: 10.1126/science.131.3416.1809.
- [3] Stuart K Card, Thomas P Moran, and Allen Newell. *The psychology of human-computer interaction*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1983.
- [4] Micah Carroll, Rohin Shah, Mark K. Ho, Tom Griffiths, Sanjit A. Seshia, Pieter Abbeel, and Anca D. Dragan. On the utility of learning about humans for human-ai coordination. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 5175–5186, 2019. URL <https://proceedings.neurips.cc/paper/2019/hash/f5b1b89d98b7286673128a5fb112cb9a-Abstract.html>.
- [5] Rujikorn Charakorn, Poramate Manoonpong, and Nat Dilokthanakul. Generating diverse cooperative agents by learning incompatible policies. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=UkU05G0H7_6.
- [6] Shuai Chen and Dennis M. Levi. Angle judgment: Is the whole the sum of its parts? *Vision Research*, 36(12):1721–1735, 1996. doi: 10.1016/0042-6989(95)00245-6.
- [7] Gijs de Rooij, Adam Balint Tisza, and Clark Borst. Flight-based control allocation: Towards human–autonomy teaming in air traffic control. *Aerospace*, 11(11):919, 2024. doi: 10.3390/aerospace11110919.
- [8] Farama Foundation. Lunar lander - gymnasium documentation. https://gymnasium.farama.org/environments/box2d/lunar_lander/, 2026. Accessed 2026-08-28.
- [9] C. S. Furmanski and S. A. Engel. An oblique effect in human primary visual cortex. *Nature Neuroscience*, 3(6):535–536, 2000. doi: 10.1038/75702.
- [10] Samuel J Gershman, Eric J Horvitz, and Joshua B Tenenbaum. Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, 349(6245): 273–278, 2015. doi: 10.1126/science.aac6076.
- [11] Sven Gronauer and Klaus Diepold. Multi-agent deep reinforcement learning: a survey. *Artificial Intelligence Review*, 55(2):895–943, 2022. doi: 10.1007/s10462-021-09996-w.
- [12] Pablo Hernandez-Leal, Bilal Kartal, and Matthew E Taylor. A survey and critique of multiagent deep reinforcement learning. *Autonomous Agents and Multi-Agent Systems*, 33(6):750–797, 2019. doi: 10.1007/s10458-019-09421-1.
- [13] Hengyuan Hu, Adam Lerer, Alex Peysakhovich, and Jakob Foerster. ‘other-play’ for zero-shot coordination. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119, pages 4399–4410. PMLR, 2020. URL <https://proceedings.mlr.press/v119/hu20a.html>.
- [14] Hengyuan Hu, Adam Lerer, Brandon Cui, Luis Pineda, Noam Brown, and Jakob Foerster. Off-belief learning. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139, pages 4369–4379. PMLR, 2021. URL <https://proceedings.mlr.press/v139/hu21c.html>.
- [15] Max Jaderberg, Valentin Dalibard, Simon Osindero, Wojciech M. Czarnecki, Jeff Donahue, Ali Razavi, Oriol Vinyals, Tim Green, Iain Dunning, Karen Simonyan, Chrisantha Fernando, and Koray Kavukcuoglu. Population based training of neural networks, 2017. arXiv:1711.09846.
- [16] Shervin Javdani, Henny Admoni, Stefania Pellegrinelli, Siddhartha S Srinivasa, and J Andrew Bagnell. Shared autonomy via hindsight optimization for teleoperation and teaming. *The International Journal of Robotics Research*, 37(7):717–742, 2018. doi: 10.1177/0278364918776060.

- [17] S. Mo Jones-Jang and Yong Jin Park. How do people react to ai failure? automation bias, algorithmic aversion, and perceived controllability. *Journal of Computer-Mediated Communication*, 28(1):zmac029, 2022. doi: 10.1093/jcmc/zmac029.
- [18] Cassidy Laidlaw and Anca Dragan. The boltzmann policy distribution: Accounting for systematic suboptimality in human models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=_l_QjPGN5ye.
- [19] Marc Lanctot, Vinicius Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Perolat, David Silver, and Thore Graepel. A unified game-theoretic approach to multiagent reinforcement learning. *Advances in neural information processing systems*, 30, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/3323fe11e9595c09af38fe67567a9394-Abstract.html.
- [20] Falk Lieder and Thomas L. Griffiths. Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and Brain Sciences*, 43:e1, 2020. doi: 10.1017/S0140525X1900061X.
- [21] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 30:6379–6390, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/68a9750337a418a86fe06c1991a1d64c-Abstract.html.
- [22] Andrei Lupu, Brandon Cui, Hengyuan Hu, and Jakob Foerster. Trajectory diversity for zero-shot coordination. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139, pages 7204–7213. PMLR, 2021. URL <https://proceedings.mlr.press/v139/lupu21a.html>.
- [23] Duane T. McRuer and Henry R. Jex. A review of quasi-linear pilot models. *IEEE Transactions on Human Factors in Electronics*, HFE-8(3):231–249, 1967. doi: 10.1109/THFE.1967.234304.
- [24] Paul Muller, Shayegan Omidshafiei, Mark Rowland, Karl Tuyls, Julien Perolat, Siqi Liu, Daniel Hennes, Luke Marris, Marc Lanctot, Edward Hughes, Zhe Wang, Guy Lever, Nicolas Heess, Thore Graepel, and Remi Munos. A generalized training approach for multiagent learning. In *International Conference on Learning Representations (ICLR)*, 2020. URL <https://openreview.net/forum?id=Bk15kxrKDr>.
- [25] Mohammad Saif Nazir and Chayan Banerjee. Zero-shot llms in human-in-the-loop rl: Replacing human feedback for reward shaping, 2025. arXiv:2503.22723.
- [26] Frans A Oliehoek and Christopher Amato. *A concise introduction to decentralized POMDPs*. Springer, 2016. doi: 10.1007/978-3-319-28929-8.
- [27] Antti Oulasvirta, Sunjun Kim, and Byungjoo Lee. Neuromechanics of a button press. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI ’18, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356206. doi: 10.1145/3173574.3174082.
- [28] Keenan Powell, Peihong Yu, and Pratap Tokekar. Zero shot coordination for sparse reward tasks with diverse reward shapings, 2026. arXiv:2604.25076.
- [29] Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-Baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL <http://jmlr.org/papers/v22/20-1364.html>.
- [30] Siddharth Reddy, Anca D Dragan, and Sergey Levine. Shared autonomy via deep reinforcement learning. In *Robotics: Science and Systems (RSS)*, 2018. doi: 10.15607/RSS.2018.XIV.005.
- [31] Arthur J Robson. Efficiency in evolutionary games: Darwin, nash and the secret handshake. *Journal of theoretical biology*, 144(3):379–396, 1990. doi: 10.1016/S0022-5193(05)80082-7.
- [32] Charles Schaff and Matthew R. Walter. Residual policy learning for shared autonomy. In *Robotics: Science and Systems*, volume 16, 2020. doi: 10.15607/RSS.2020.XVI.072.

- [33] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. arXiv:1707.06347.
- [34] Peter Stone, Gal A. Kaminka, Sarit Kraus, and Jeffrey S. Rosenschein. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, pages 1504–1509, 2010. doi: 10.1609/aaai.v24i1.7529.
- [35] Tilo Strobach, Peter A Frensch, and Torsten Schubert. Video game practice optimizes executive control skills in dual-task and task switching situations. *Acta Psychologica*, 140(1):13–24, 2012. doi: 10.1016/j.actpsy.2012.02.001.
- [36] DJ Strouse, Kevin R. McKee, Matt Botvinick, Edward Hughes, and Richard Everett. Collaborating with humans without human data. In *Advances in Neural Information Processing Systems*, volume 34, pages 14502–14515, 2021. URL <https://proceedings.neurips.cc/paper/2021/hash/797134c3e42371bb4979a462eb2f042a-Abstract.html>.
- [37] Gábor J. Székely and Maria L. Rizzo. Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8):1249–1272, 2013. doi: 10.1016/j.jspi.2013.03.018.
- [38] Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974. doi: 10.1126/science.185.4157.1124.
- [39] Nash Unsworth, Thomas S Redick, Chad E Lakey, and Diana L Young. Lapses in sustained attention and their relation to executive control and fluid abilities: An individual differences investigation. *Intelligence*, 38(1):111–122, 2010. doi: 10.1016/j.intell.2009.08.002.
- [40] Robert Whelan. Effective analysis of reaction time data. *The Psychological Record*, 58(3): 475–482, 2008. doi: 10.1007/BF03395630.
- [41] Liang Yan, Xiaodong Wu, and Hangyu Lu. Human-centered collaborative decision-making and steering control with reinforcement learning. In *16th International Symposium on Advanced Vehicle Control (AVEC 2024)*, pages 742–748. Springer, 2024. doi: 10.1007/978-3-031-70392-8_105.
- [42] Rui Zhao, Jinming Song, Yufeng Yuan, Haifeng Hu, Yang Gao, Yi Wu, Zhongqian Sun, and Wei Yang. Maximum entropy population-based training for zero-shot human-ai coordination. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, volume 37, pages 6145–6153, 2023. doi: 10.1609/aaai.v37i5.25758.

A Implementation Details

A.1 Reward Function

As in the standard Lunar Lander environment [8], the continuous step-wise reward in Equation (1) is derived from the difference in potential between consecutive states, incentivising the agent to continuously minimise its speed, tilt, and distance from the goal. Per-frame penalties are applied for fuel consumption to discourage inefficient hovering or erratic manoeuvres. The environment charges these penalties in proportion to the power at which each thruster fires, so the fixed magnitudes imposed by the multi-discrete wrapper of Section 3 determine the effective coefficients: the main engine, firing at full power, incurs the nominal 0.3 per frame, while the lateral thrusters, firing at 0.7, incur 0.021 rather than the typical 0.03. Lateral input is therefore roughly fourteen times cheaper per frame than main-engine input, as opposed to the environment’s ten-to-one ratio at full side thruster power, though the weaker thrust means a given correction must be held for longer. The final step’s reward is replaced by +100 for a landing or -100 for a crash, including leaving the screen, and $P(d)$ is deducted from landings only.

A.2 Ex-Gaussian Delay Distribution

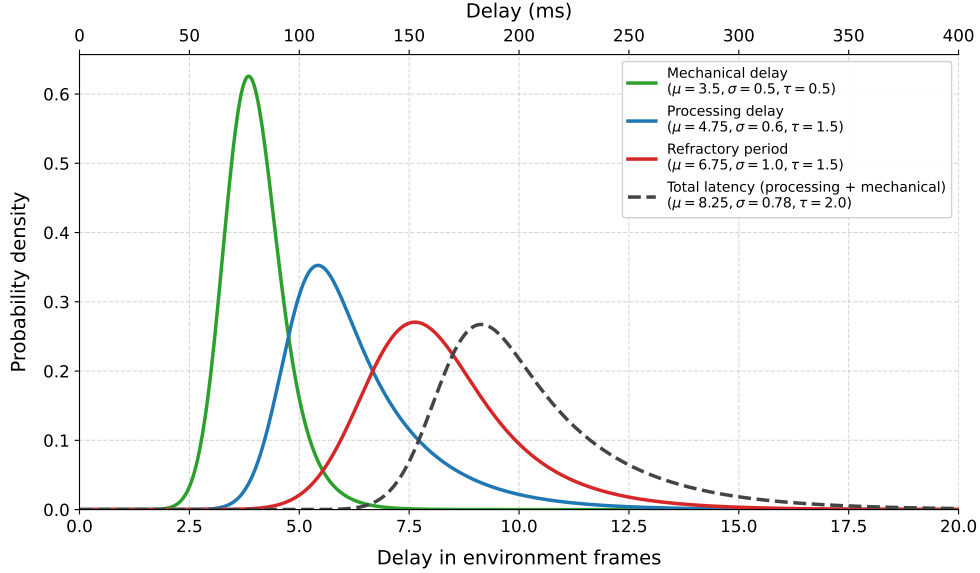


Figure 4: Ex-Gaussian delay distributions for a single constrained agent, shown at the midpoint of each parameter range. The mean of every phase is sampled independently for each constrained agent, so each member of the population carries a different set of curves. The dashed curve is the convolution of the processing and mechanical delays, giving the total latency between a change in the environment and the corresponding change in the partner’s output.

A.3 Perceptual Thresholds

Section 4 describes a perceptual filter applied to the horizontal and vertical velocities and to the tilt angle, with each channel compared against the value at which it was last registered. The agent’s two Weber fractions are drawn before training from a single scalar $p \sim \mathcal{U}(0, 1)$, mapped linearly onto the ranges reported in the psychophysical literature:

$$w_\theta = 0.045 + 0.035 p, \quad w_v = 0.07 + 0.04 p. \quad (3)$$

For the velocities the threshold is $\max(0.02, w_v |\dot{q}_{\text{last}}|)$, applied separately to $\dot{x}$ and $\dot{y}$ with $\dot{q}$ denoting whichever is being tested. The Weber fraction governs where it exceeds the floor, and the constant

otherwise; the crossover falls at roughly $|\dot{q}_{\text{last}}| = 0.2\text{--}0.3$, so the floor is the operative term through a substantial portion of a controlled descent.

For tilt the correction takes the form of a constant offset rather than a floor, giving a threshold of $w_\theta (|\theta_{\text{last}}| + 1)$. The offset sets a baseline below which no change is perceptible regardless of orientation, so the agent requires a change of w_θ radians when level and proportionally more as the craft tilts further. Since $|\theta|$ seldom exceeds 0.5 rad, this yields a near-constant threshold of approximately $2.6\text{--}4.6^\circ$, several times stricter than the ratio alone would give at these magnitudes.

An exception is the noted increased sensitivity in humans to rotational changes near the horizontal and vertical axes (cardinal angles), known as the ‘oblique effect’ [9]. To replicate this, the tilt threshold is halved when $|\theta_{\text{last}}|$ lies within 10° of a cardinal angle and quartered when within 2° , so an agent flying close to level detects tilt changes two to four times smaller than one flying obliquely. The adjustment is keyed to the last perceived angle rather than the true one, since a player’s acuity depends on where they believe the craft to be pointing. Since the lander is near-vertical for much of a controlled descent, the reduced threshold is the operative one throughout the approach, with the unmodified value applying mainly during recovery from a heavy tilt. Even under the strongest reduction the tilt threshold remains several times larger than a pure ratio would give at typical tilts.

The filter is applied to the buffered observation after the processing delay has been imposed. Both the difference and the threshold are evaluated on magnitudes, so a quantity passing through zero always registers. Each channel maintains its own record, so a registered change on one does not refresh the others.

A.4 Efferece Copy

A player operating a keyboard is continuously aware of which key they are holding – this sensation is tactile and immediate, rather than inferred through observing what happens in the game. The constrained agent, by contrast, emits an intent each frame and receives no indication of whether that intent was committed, queued behind the mechanical delay, or discarded during a refractory period. To get this information, it would have to reconstruct its own motor state from delayed and filtered physics in a decidedly inhuman manner. To remove this deficit the agent observes the lateral input it is currently executing, expanding its observation space from eight dimensions to nine. This value is not subject to any delays or to the perceptual filter of Appendix A.3, as efferece copy (the internal copy of motor commands created by the brain) and tactile feedback are substantially faster than the visual pathway. The internal timers governing the refractory and commit windows are deliberately withheld, as a person has no introspective access to their own psychological refractory period.

A.5 Baselines

A.5.1 MARL

Our standard MARL baseline represents traditional, unconstrained joint-optimisation, where agents co-adapt from scratch without pre-existing understanding of the underlying physics environment. However, applying simultaneous policy updates across disjoint continuous action spaces induces severe non-stationarity, rendering the environment’s transition dynamics highly unstable from either agent’s perspective. To resolve this “moving-target” problem while preserving the capacity for mutual adaptation, we implement a staggered training paradigm. By temporarily freezing one partner’s policy, the transition dynamics are stabilised from the perspective of the active agent, effectively reducing the multi-agent challenge to a stationary, single-agent Markov Decision Process (MDP). This alternating approach circumvents the gradient instability inherent to simultaneous independent learning [12]. Accordingly, our active learning role alternates between the main and lateral agents every 20,000 timesteps.

This interval equates approximately to 85 full episodes, and is intended to balance statistical significance with policy entropy. Extremely short intervals allow gradient updates to be dominated by the variance of random environmental initialisations, effectively recreating the moving-target instability. Conversely, prolonged active phases cause the agent to fully converge against its static partner, leading to catastrophic forgetting and highly brittle, exploitative policies [19]. This overfitting can be prevented by truncating the learning process to compute an approximate best response [24].

This ensures the policy shifts enough to meaningfully improve coordination, without fully overfitting to the frozen partner, enabling stable co-adaptation.

To prevent swap intervals from terminating active simulations mid-episode, the agents train in completely decoupled environment instances. The active agent interacts with its dedicated environment while querying its frozen partner, ensuring experience collection spans uninterrupted episodes to avoid injecting bias into the policy’s advantage estimation [11]. Furthermore, this strictly partitioned control induces a Decentralised Partially Observable Markov Decision Process (Dec-POMDP). To mitigate this partial observability problem, each agent’s observation space is appended with its partner’s previously executed action in order to construct a basic action-observation history, a mathematically established mechanism for facilitating coordination in decentralised tasks [26].

A.5.2 PBT

Before any multi-agent coordination begins, models are initially trained to develop a foundational understanding of the environment’s physics. A random subset of these agents are then initialised using these pre-adapted weights, isolating the subsequent learning phase strictly to the teaming dynamics. During this collaborative phase, control is again partitioned such that the primary agent has control over the main engine, while the partner agent uses the lateral thrusters.

At the start of each episode, the environment randomly selects a frozen partner from the population of partner agents. The primary agent has no explicit knowledge of which static policy is controlling the lateral thrusters in a given episode, meaning this training effectively results in a generalised best response to the entire distribution of partners. As with the MARL framework, the partner agent’s current input is appended to the primary agent’s state space, although in this case the knowledge is shared unilaterally.

A.5.3 FCP

FCP computes a best response across a distribution of partners drawn from different stages of their learning trajectories, conventionally by checkpointing symmetric self-play agents [36]. Partitioned control requires two adaptations.

The first concerns pool generation. Training both roles from scratch simultaneously yields unrecoverable catastrophic failures, while initialising them from a pre-trained joint policy artificially flatlines the learning curve, precluding the extraction of intermediate skill checkpoints. We instead generate partners by computing an asymmetric best response against fixed unconstrained policies: a set of pre-trained, unconstrained main agents is frozen to provide the vertical stability necessary to prevent premature episode termination, and independent lateral agents are initialised from scratch and trained against these static main controllers. This stabilisation permits the lateral agents to generate the clean, upward-trending learning trajectories necessary for historical checkpointing.

The second concerns checkpoint selection. The standard framework samples from precisely the start, middle and end of training to maximise behavioural diversity. This is ill-suited to strictly partitioned continuous control: where a proficient agent in a symmetric task can compensate for a severely inept partner, extreme incompetence on the lateral thrusters produces repeated unrecoverable failures and no actionable learning signal. To ensure both population diversity and usable feedback, we sample at the 33%, 66% and 100% marks of peak learning. Fifteen independently trained lineages sampled at three tiers give the 45-partner pool used in training.

B Training Details

Experiments were run under Python 3.9.6 with Gymnasium 1.1.1, Stable-Baselines3 2.7.1 [29] and sb3-contrib 2.7.1, PyTorch 2.8.0 and NumPy 2.0.2. All parameters not listed in Table 2 take the Stable-Baselines3 defaults for RecurrentPPO with an MlpLstmPolicy. Training uses a single environment instance, and no observation or reward normalisation is applied. Only the evaluation seeds are fixed; training runs are unseeded, so exact numerical reproduction of a given agent is not possible, although each constrained partner’s sampled constraint profile is written to disk alongside its weights so that the population itself is recoverable.

All training was performed on CPU with a single PyTorch thread per run, with runs executed 32-wide across pinned physical cores. Training runs required up to 4 GB of memory each, and had an approximate duration of 8–12 hours per training instance. No GPU was used. Each agent was trained for a maximum of 2,500,000 timesteps, with early stopping applied within that budget as described in Appendix B.4. The full project required considerably more compute than the runs reported here, as the training pipeline and constraint implementation were revised over several iterations whose agents are not reported.

B.1 Hyperparameters

All agents, across both the proposed framework and the baselines, use Recurrent Proximal Policy Optimisation with an `MlpLstmPolicy`, for the reasons given in Section 4; in the teaming phase the recurrence additionally compensates for the partitioned action space, in which neither party observes the other’s forthcoming decision.

Hyperparameters were established through a combination of standard convention and empirical testing during the training of the solo agents, and are identical across all conditions except where noted in Table 2.

Table 2: Training hyperparameters. All agents use RecurrentPPO with an `MlpLstmPolicy`.

Parameter	Value	Notes
Discount factor (γ)	0.99	Standard for continuous control
Learning rate	0.0005	Constant throughout training
Entropy coefficient	0.005	Balances exploration with destabilisation risk
Batch size	128	Empirically tuned on solo agents
Rollout length	1024	Empirically tuned on solo agents
Episode limit	750 steps	Same for all conditions
Training budget	2,500,000 steps	5,000,000 for MARL, split across two learners
Early-stopping patience	5,000 episodes	50 evaluations at 100-episode intervals
Evaluation frequency	100 episodes	On 50 fixed deterministic seeds
Partner pool size	45	Matched across non-MARL conditions
Evaluation partners	50	Disjoint from the training pool

B.2 Weight Transfer

The PBT, FCP and primary teaming agents are initialised from solo agents pre-trained on the environment’s physics without a partner. Because the partner’s executed lateral action is appended to the observation, the input layer expands from eight to nine dimensions. The LSTM and shared feature layers are transferred directly. Due to this expansion, the recurrent input matrix expands from eight to nine columns; the eight physical-state columns are copied and the ninth is randomly initialised. The action head is not transferred, since the solo agent’s `MultiDiscrete` head does not match the teaming agent’s binary main-engine head, so the agent inherits a learned representation of the environment’s physics and relearns the output mapping over it. The MARL agents receive no transfer, since unconstrained joint optimisation from scratch is what that baseline represents. The cognitively constrained partners are similarly trained from scratch.

B.3 Evaluation Protocol

Evaluation is conducted every 100 training episodes against a fixed array of 50 deterministic seeds, with matched fixed partners in the teaming conditions. Holding both the environment seeds and the partner constant isolates policy improvement from variation in initial conditions or partner sampling.

The size of the evaluation set was calibrated to control false positives during checkpoint selection. Since the environment’s returns are high-variance and bimodal, a checkpoint evaluated across too few seeds may post a higher mean reward through favourable initial conditions alone, causing it to be erroneously saved as the new best model and to reset the early-stopping counter. Evaluating against a very large seed pool would suppress this effect, but is computationally prohibitive given the number of checkpoints and agents involved.

A selection of agents was therefore evaluated at every training checkpoint across a pool of 100 fixed seeds, retaining the per-seed returns. For each pair of successive checkpoints exhibiting an increase in mean return, a paired one-tailed t -test was applied to progressively larger subsets of the seed pool in increments of ten. This established the subset size at which improvements of the magnitude typically observed during training were distinguishable from seed variability at the 5% level. Fifty seeds proved sufficient across the agents and improvements examined, so this was adopted as the evaluation set size.

B.4 Early Stopping

Models were trained for a maximum of 2,500,000 timesteps, with the exception of the MARL condition, whose budget of 5,000,000 is divided between two concurrently trained agents so that each receives an equivalent 2,500,000.

To preserve computational resources while preventing premature termination, an early-stopping patience mechanism was implemented. Preliminary training visualisations of agents revealed that policies frequently underwent periods of severe temporary destabilisation, sometimes lasting several thousand episodes, before recovering to achieve a higher global optimum. A tolerant patience threshold was therefore adopted. Training terminates only when the mean evaluation reward fails to improve across 5,000 consecutive episodes, equivalent to 50 successive evaluations. This affords agents sufficient exploratory bandwidth to recover from performance dips during late-stage fine tuning.

C Future Analysis Plan

C.1 Human Study Details

Initially, participants will play through 20 episodes of the game on their own to familiarise themselves with the mechanics and goals, ensuring a basic competence before the teaming phase begins. Participants will then play 20 episodes with each of the four teaming agent types (MARL, PBT, FCP, and the Cognitively Constrained Partner agent), 80 in total, with agent order randomised to mitigate ordering effects and learning bias. Finally, the participants will play 20 episodes alone again before completing a short debrief questionnaire. With each episode lasting roughly ten seconds, the full 120-episode protocol amounts to approximately 20 minutes of play, and participants may take breaks between episodes whenever they wish. Each teaming agent is colour-coded so that participants can retrospectively rank their preferences, and the questionnaire asks whether they felt each agent adapted to their intent and whether it made the game easier than playing alone. One hundred participants will take part in the study, which has been given ethical approval. They will be recruited through the university, with no restriction on gaming aptitude so as to ensure a broad spectrum of behaviour types. Since the design is within-subject, each agent contrast is estimated within participants rather than between them, and the 20 episodes played alongside each agent average out the episode-level variance the environment’s bimodal returns produce.

Ten teaming agents will be trained under each condition for use in the study, and each of the four resulting lists of 10 agents will then be randomly shuffled separately. In order to ensure we can separate differences that could arise from different agent initialisations, the indexes of these agents in their respective lists will act to pair them. The 100 participants in the study will randomly be assigned an index at the start of the experiment such that each batch of agents is played against 10 times.

C.2 Mixed-effects Model

Let y_{ij} denote the mean score achieved by participant i when partnered with agent j , where j indexes the four teaming conditions (Constrained, PBT, FCP, MARL). Each participant is assigned, at the start of the study, a single index $k_i \in \{1, \dots, 10\}$ drawn from a balanced random allocation, so that every value of k_i is held by exactly ten participants. This index selects which of the ten agent instances retained for each condition (Section 7) the participant plays: participant i is partnered under condition j with the instance identified by (j, k_i) . Scores are modelled as

$$y_{ij} = \beta_0 + \beta_j + u_i + v_{j,k_i} + \varepsilon_{ij}, \quad u_i \sim \mathcal{N}(0, \sigma_u^2), \quad v_{j,k} \sim \mathcal{N}(0, \sigma_v^2), \quad \varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2),$$

where β_0 is the expected score of a typical participant partnered with a typical instance of the reference agent, β_j the change in expected score attributable to condition j , u_i the baseline proficiency of participant i , $v_{j,k}$ the deviation of agent instance k under condition j from that condition’s mean, and ε_{ij} the residual variation unexplained by other terms. The constrained-partner agent is taken as the reference level, so each β_j expresses a baseline’s performance relative to it and the three planned contrasts are read directly from the fitted coefficients. Trial index is included as a covariate, since the order in which agents are encountered is randomised rather than balanced and participants may improve over the course of a session.

Each participant is paired with every agent, so their scores are not independent: a proficient player scores highly under every condition and a struggling one poorly. Rather than fitting a separate baseline for each of the hundred participants, the model estimates the variance σ_u^2 of baseline proficiency across the population. This is recoverable from the data because participants who are more able score highly under every condition, so their four scores rise and fall together; the stronger that tendency, the more participants must differ from one another. Estimating it allows the variation between participants to be set aside, so that agents are compared within individuals rather than across them. Pilot data indicated substantial between-participant variation, with mean scores spanning several hundred points, which would otherwise be attributed to residual error and obscure any difference between agents.

The random effect $v_{j,k}$ serves a similar purpose for agent instances rather than participants. Ten agents are trained under each condition, so any single instance’s result is one draw from the outcomes that condition’s training procedure can produce, not a deterministic property of the condition itself. Without $v_{j,k}$, systematic differences between a condition’s own instances (one agent initialisation that happened to train unusually well or poorly) would be absorbed into ε_{ij} and reported as if they reflected the participant’s response to the method in general. Because the balanced index assignment guarantees exactly ten participants per instance, σ_v^2 is identifiable within each condition from the spread of those ten instance means around the condition mean, and β_j is correspondingly interpreted as the effect of a condition averaged over the instances it can produce, with the standard errors of the planned contrasts widened to reflect that averaging rather than treating all ten instances of a condition as interchangeable repetitions of one true effect.

Both u_i and $v_{j,k}$ will be checked for normality after fitting. Pilot data suggests a bimodality in baseline proficiency, with participants tending either to land consistently or rarely succeed, so it is possible that the participant random intercepts depart from normality. Should this occur, u_i will be modelled as a mixture of two normal components rather than a single one, corresponding to the two levels of competence apparent in the pilot. No comparable prior evidence exists for $v_{j,k}$, so it will be assumed normal.

A separate concern applies to $v_{j,k}$ regardless of its shape: with only ten instances per condition, its variance σ_v^2 will necessarily be estimated imprecisely, and diagnostics may be unable to distinguish it from zero. This does not threaten the model’s main results; the fixed-effect estimates β_j and their standard errors come from the same fit that includes $v_{j,k}$, and already reflect whatever instance-level variation is present regardless of how precisely σ_v^2 itself can be determined, so they remain the paper’s primary result under either outcome. What changes is only whether σ_v^2 is reported as a finding in its own right, as if well-estimated, it will be presented alongside σ_u^2 as evidence of how much a participant’s score depended on the specific trained instance they played. If it cannot be reliably distinguished from zero, no claim will be made about its magnitude beyond noting that it was included in, and accounted for by, the model.

This retains the structure of the model and the interpretation of the agent effects, while representing both the population of participants and the population of trained instances as they are rather than as single homogeneous groups.

C.3 Secondary Analyses

Since each participant plays multiple episodes alongside every agent, their scores cannot be treated as independent observations. The random intercept accounts for this by allowing each participant their own baseline level of performance, such that the comparison between agents is made relative to that individual rather than to the group mean. This isolates the effect of the agent’s policy from the substantial variation in baseline competence between participants.

Given that the hypothesis requires the proposed agent to outperform each of the baselines, all three contrasts must reach significance for it to be supported. Requiring every test to succeed is a stricter criterion than requiring any one of them to succeed, so the risk of a false positive is already controlled and no correction for multiple comparisons is applied.

A secondary analysis will examine episodes separately by outcome, indicating whether an agent contributes primarily by maximising the score of successful attempts or by limiting the losses incurred during a crash. Finally, further analysis will investigate whether the agents' support varies across different user demographics. By comparing performance across age groups and levels of video game experience, this will reveal whether the cognitively constrained agent provides a universal benefit, or if it disproportionately aids less experienced players by compensating for a lack of pre-existing aptitude.

D Extended Results

Twenty-two metrics in total were analysed when comparing how similar the constrained and unconstrained agents were to human players, shown in Table 3.

Three measures are reported for each, since no single one is adequate alone. The raw difference in group means is directly interpretable but scale-dependent, so cannot be compared across features. The Wasserstein distance normalised by group spread represents how far apart distributions are, but can be distorted by distributions with one long tail. The overlap measures the shared probability mass, but cannot distinguish magnitudes of separation if distributions share no overlap.

The mean press duration refers to the average amount of time an input is held down for on each instance of its use, with the main and side thrusters measured separately. The final displacement is the absolute horizontal distance from the landing pad at termination, with the tilt and speed measuring the angle and velocity at the last frame of the game. Idle refers to the proportion of frames throughout the episode where each or neither of the engines are inactive, with mean idle length referring to the average duration of intervals in which neither control is active. Tilt measures the root-mean-square tilt across the episode. Score refers to the total reward accumulated over the episode. Press duration CV is the coefficient of variation, computed by dividing the standard deviation of press lengths by their mean. Vertical speed is the mean descent rate across the episode. Inputs per second is the number of separate presses made each second, counted once per press regardless of how long it is held, and measured separately for the main and side thrusters. Episode length is the number of frames elapsed before the episode terminates. Action entropy refers to the Shannon entropy over the three lateral thruster states (off/left/right), measuring the shape of the whole action distribution. The reversal rate is the number of times left-to-right or right-to-left switches with a gap of 10 frames or less occur per second, and the burst rate measures occurrences of three or more direction changes within 25 frames, per second. The corrective latency counts the frames from the craft tilting past an angle of 0.1 radians off centre to when a thrust occurs to correct this, and the left-right balance measures the directional asymmetry between the left and right actions.

Table 3: Individual feature comparison of agents with the human distribution. Overlap is the shared probability mass (higher is closer); distance is the normalised Wasserstein distance (lower is closer). Bold indicates the closer of the two agent types under each of the two distributional measures.

Feature	Overlap		Wasserstein	
	Constrained	Unconstrained	Constrained	Unconstrained
Mean press duration (side)	0.73	0.00	0.32	2.01
Final speed	0.77	0.50	0.72	1.07
Final displacement	0.59	0.25	0.73	1.32
Final tilt	0.78	0.75	0.74	0.94
Idle (side)	0.36	0.71	0.86	0.42
Tilt (RMS)	0.67	0.22	0.89	1.14
Mean idle length	0.17	0.00	0.89	1.95
Score	0.43	0.05	0.90	1.39
Idle (main)	0.29	0.56	0.91	1.02
Press duration CV (side)	0.49	0.09	0.92	1.73
Vertical speed	0.32	0.31	1.06	1.11
Action entropy	0.26	0.42	1.21	0.50
Idle (neither control)	0.24	0.61	1.22	0.35
Press duration CV (main)	0.40	0.31	1.22	1.05
Reversal rate	0.26	0.49	1.26	0.85
Episode length	0.34	0.05	1.27	1.58
Burst rate	0.24	0.00	1.27	1.65
Corrective latency	0.24	0.52	1.42	1.21
Inputs per second (main)	0.19	0.00	1.44	1.69
Inputs per second (side)	0.24	0.00	1.47	1.50
Left/right balance	0.24	0.47	1.49	1.13
Mean press duration (main)	0.19	0.02	1.50	1.95

Table 4: Mean value and standard deviation of each feature by group, averaged within and then across controllers. Bold indicates the agent type nearer the human mean. Reported alongside Table 3, since normalisation via pooled standard deviation can distort rankings for features with long tails.

Feature	Human		Constrained		Unconstrained	
	Mean	(SD)	Mean	(SD)	Mean	(SD)
Mean press duration, side (ms)	298.5	(80.1)	279.8	(98.2)	47.1	(9.3)
Mean press duration, main (ms)	561.8	(171.0)	353.2	(49.3)	98.5	(42.4)
Press duration CV, side	0.438	(0.173)	0.533	(0.088)	0.982	(0.196)
Press duration CV, main	0.537	(0.084)	0.435	(0.063)	0.858	(0.333)
Mean idle length (ms)	321.1	(104.7)	196.1	(192.5)	49.2	(13.0)
Inputs per second, side	0.940	(0.245)	1.623	(0.416)	6.122	(2.970)
Inputs per second, main	0.924	(0.270)	1.422	(0.284)	4.766	(1.687)
Burst rate (per s)	0.432	(0.406)	1.391	(0.690)	9.733	(4.344)
Reversal rate (per s)	0.144	(0.125)	0.607	(0.349)	2.362	(2.908)
Corrective latency (ms)	286.3	(132.0)	130.5	(46.5)	145.8	(76.7)
Idle, side (fraction)	0.696	(0.130)	0.565	(0.167)	0.714	(0.108)
Idle, main (fraction)	0.569	(0.049)	0.531	(0.074)	0.608	(0.025)
Idle, neither control (fraction)	0.452	(0.077)	0.311	(0.129)	0.446	(0.058)
Action entropy (bits)	0.997	(0.128)	1.283	(0.266)	1.066	(0.222)
Left/right balance	0.340	(0.205)	0.101	(0.036)	0.170	(0.075)
Tilt, RMS (rad)	0.317	(0.213)	0.199	(0.044)	0.159	(0.018)
Vertical speed	0.343	(0.125)	0.217	(0.112)	0.342	(0.026)
Final displacement	0.263	(0.227)	0.202	(0.073)	0.055	(0.022)
Final speed	0.477	(0.704)	0.189	(0.223)	0.001	(0.007)
Final tilt (rad)	0.367	(0.630)	0.142	(0.066)	0.008	(0.012)
Episode length (frames)	261.0	(87.4)	397.8	(89.0)	198.6	(19.9)
Score	77.3	(211.0)	108.6	(50.0)	286.3	(8.2)



Figure 5: Distribution of behavioural features comparing human, constrained agent, and unconstrained agent groups.

Confidence intervals throughout Section 6.2 are bootstrap percentiles over 5,000 resamples. The design has two sampling units rather than one: the ten independently trained replicates of each condition, and the fifteen held-out partners comprising each pool. Episodes are first collapsed to a mean for each (replicate, partner) pairing, since the episodes one policy plays against one partner are repeated measurements of that pairing rather than independent observations; resampling episodes directly would treat them as independent and yield intervals several times too narrow.

Because every replicate meets every partner, the two units are crossed rather than nested. Each bootstrap iteration therefore draws one set of replicates and one set of partners, with replacement, and applies the same partner draw across all replicates. Resampling partners independently within each replicate would average the partner variance component over the ten replicates and understate it by a factor of approximately $\sqrt{10}$.

Table 1 pairs by replicate such that the same replicates are drawn for the own-pool and held-out sides of each difference, so a draw containing unusually strong runs raises both sides together and run strength cancels. Table 5 pairs by partner instead, drawing one partner set applied to both agents so that per-partner difficulty cancels, but resamples replicates independently for each agent, since the two methods' training runs have no correspondence on which to pair them.

All intervals are conditional on the fixed set of twenty evaluation seeds, which is held constant across every agent, partner and replicate.

Table 5: Held-out advantage of the constrained agent over each baseline, restricted to the partner families neither agent trained on so that both are averaged over the same partners. Intervals are percentiles of a two-way cluster bootstrap over 5,000 resamples, paired by partner so that per-partner difficulty cancels from the difference, with replicates resampled independently for each agent.

Comparison	Shared families	Advantage	95% CI	Significance
vs PBT	FCP, MARL	-4.5	$[-11.4, +1.9]$	Not significant
vs FCP	PBT, MARL	+4.0	$[-1.8, +10.0]$	Not significant
vs MARL	PBT, FCP	+27.5	$[+8.7, +52.7]$	Significant

E Questionnaire Items

Immediately after their final block of solo play, participants will complete a short questionnaire covering demographic and gaming-experience covariates followed by their subjective experience of each teaming agent, as described in Section 7. Agents are identified to participants only by a fixed colour label (White, Blue, Red, Yellow), held consistent across all participants. Items 6–9 refer back to the ranking given in Item 5, so that each participant's most- and least-successful agent is unambiguous both to them and in the resulting data.

1. Participant number (free entry), used to link responses to the corresponding session log.
2. Age, in one of seven bands: 18–25, 26–35, 36–45, 46–55, 56–65, 66–75, 76+.
3. Frequency of video game play, on a six-point scale from Daily to Never.
4. Perceived difference between the four agents' cooperative ability, from 1 (no difference) to 10 (large difference).
5. Ranking of the four agents by perceived success of the partnership (1 = most successful, 4 = least successful).
6. Rated impact of the agent ranked first in Item 5 on the participant's score, from 1 (very detrimental) to 10 (very helpful).
7. As Item 6, for the agent ranked last.
8. Rated extent to which the agent ranked first in Item 5 supported the participant's intended play style, from 1 (became a barrier to the participant's intentions) to 10 (understood and supported the participant's intentions).
9. As Item 8, for the agent ranked last.
10. Overall preference between solo play and play supported by a teaming agent (preferred alone / no preference / preferred with agent).
11. Free-text comments on specific agents or the teaming experience in general.

F Gameplay Information

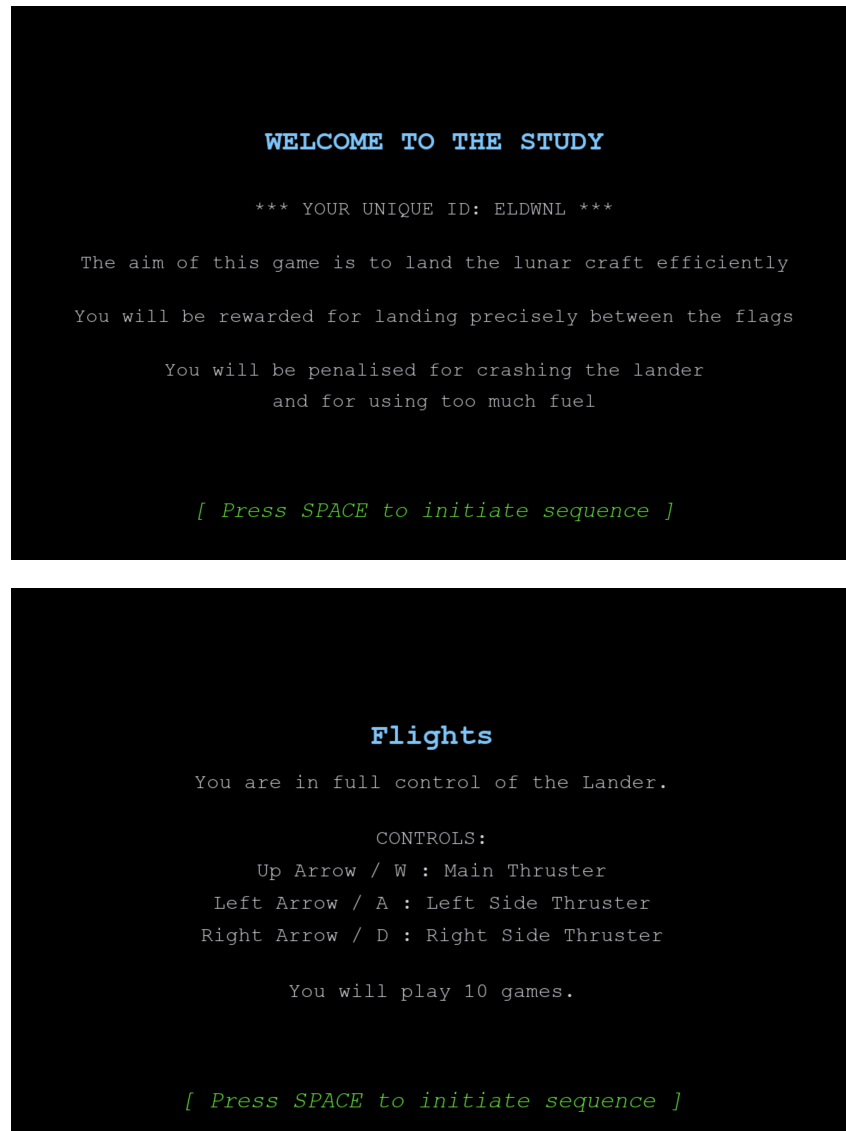


Figure 6: Instructions seen by participants at the start of the trial.