Discovering Creative Behaviors through DUPLEX: Diverse Universal Features for Policy Exploration

Borja G. Leon*
Iconic
borja@iconicgames.ai

Francesco Riccio
Sony AI
francesco.riccio@sony.com

Kaushik Subramanian
Sony AI
kaushik.subramanian@sony.com

Peter R. Wurman
Sony AI
peter.wurman@sony.com

Peter Stone
Sony AI
The University of Texas at Austin
peter.stone@sony.com

Abstract

The ability to approach the same problem from different angles is a cornerstone of human intelligence that leads to robust solutions and effective adaptation to problem variations. In contrast, current RL methodologies tend to lead to policies that settle on a single solution to a given problem, making them brittle to problem variations. Replicating human flexibility in reinforcement learning agents is the challenge that we explore in this work. We tackle this challenge by extending state-of-the-art approaches to introduce DUPLEX, a method that explicitly defines a diversity objective with constraints and makes robust estimates of policies' expected behavior through successor features. The trained agents can (i) learn a diverse set of near-optimal policies in complex highly-dynamic environments and (ii) exhibit competitive and diverse skills in out-of-distribution (OOD) contexts. Empirical results indicate that DUPLEX improves over previous methods and successfully learns competitive driving styles in a hyper-realistic simulator (i.e., GranTurismo™ 7) as well as diverse and effective policies in several multi-context robotics MuJoCo simulations with OOD gravity forces and height limits. To the best of our knowledge, our method is the first to achieve diverse solutions in complex driving simulators and OOD robotic contexts. DUPLEX agents demonstrating diverse behaviors can be found at https://ai.sony/publications/Discovering-Creative-Behaviors-through-DUPLEX-Diverse-Universal-Features-for-Policy-Exploration/.

1 Introduction

In non-stationary complex environments, reinforcement learning (RL) (Sutton & Barto, 2018) agents are compelled to exhibit flexible and diverse behaviors to robustly adapt to different scenarios and interact with other actors (Vinyals et al., 2019; Zahavy et al., 2023b). To this end, a growing community is researching methodologies to train agents that, unlike conventional RL (Haarnoja et al., 2018; Espeholt et al., 2018), can solve tasks with a diverse set of near-optimal strategies (Zahavy

*Work done at Sony AI.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

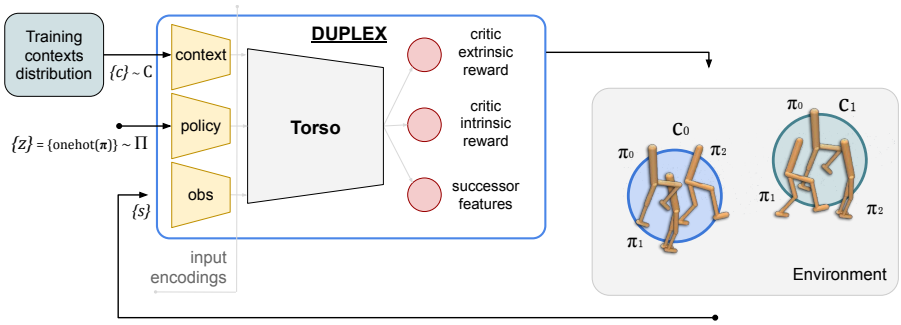


Figure 1: DUPLEX data flow while training three policies over a set of two contexts. At each iteration, we provide three inputs to our multi-policy agent: (i) a context vector describing task requirements and environment dynamics in the current episode c; (ii) an encoding of the policy used in the episode z; (iii) and the current state of the environment s. The critic network returns estimates for the intrinsic and extrinsic rewards and successor features to drive diverse behavior discovery. Finally, the algorithm samples policies in Π uniformly and rolls them out to collect more experience.

et al., 2023a). Such methodologies are explicitly crafted to enhance the exploration of the state-action space, equipping agents with the capability to discover qualitatively diverse solutions for a given task distribution. For example, when planning a route from home to work, a human commuter might identify one route that uses the highway and one that sticks to side streets. While using the highway may be optimal in expectation, using the alternative route may be called for different contexts that influence the traffic report, or weather forecast.

Existing approaches struggle to generalize to highly dynamic environments and are designed to learn in single-task settings (Zahavy et al., 2023b; Fu et al., 2023). To alleviate this, we introduce *Diverse Universal features for PoLicy EXploration* (DUPLEX), an algorithm that trains an agent to optimize a set of diverse policies over different contexts. In this work, contexts comprise two components: a task requirement and a description of the current environment dynamics. To better ground such concepts, let us refer to the previous commuter example. In this case, we would refer to task requirements as time-to-completion. On the other hand, factors such as the weather forecast and traffic report would form part of the environment dynamics DUPLEX builds on top of the strengths of diversity learning (Zahavy et al., 2023a) in RL and universal estimators (UE) (Schaul et al., 2015; Borsa et al., 2019) to introduce a novel methodology that preserves both performance and diversity in highly dynamic environments and multi-context settings – which is key to enabling human-level task execution. For example, if the commuter is a human that injures their leg, they can immediately balance on one leg and even jump forward without using the injured leg. We aim to transfer this adaptability to agents in different contexts. To this end, we adopt the Contextual Markov Decision Process (CMDP) framework (Hallak et al., 2015), where different episodes correspond to different contexts. As shown in Figure 1, DUPLEX is designed to receive three inputs representing the encoding of the context, policy, and state. Contexts are represented as feature vectors and uniformly sampled from a discrete set of predefined task requirements and dynamics settings (e.g., weather condition); the i-th policy in the set of policies Π is selected and passed to the model (e.g., different routes); and the state comes directly from the environment observation. Then, the critic output is split into different heads to support the estimation of the extrinsic and intrinsic rewards, and successor features (SFs) (Barreto et al., 2017). We refer to the reward coming from the environment as the extrinsic reward, while the metric computed to promote diversity is the intrinsic reward. It is worth noting that, at the beginning of each training episode, we uniformly sample a context from C and a policy from Π to collect experience. DUPLEX then iteratively trains the set of policies to maximize both the distance in their successor features (guided by intrinsic rewards) and the task performance (given by the extrinsic rewards). We rely on SFs to measure distances as they, by definition, represent the state features that a given policy is expected to experience along a trajectory, and thus, they are an intuitive way to quantify policy diversity.

Hence, our main contribution is DUPLEX, a novel algorithm that contributes to diversity learning in RL by improving on previous work to better preserve the diversity vs. near-optimality trade-off in highly-dynamic environments and multi-context settings. We evaluate our approach on the real-time, physically realistic, car-racing simulator GranTurismo™ 7 (GT) (Wurman et al., 2022) and on two

multi-task MuJoCo environments with changing dynamics (Walker2D, Ant) (Todorov et al., 2012). Our experimental results indicate that DUPLEX improves over previous state-of-the-art RL diversity baselines (Zahavy et al., 2023a) and UE baselines (Schaul et al., 2015; Borsa et al., 2019). In fact, DUPLEX is the only algorithm to learn diverse competitive policies in GT and outperforms other baselines when evaluated in out-of training distribution (OOD) contexts. Moreover, we conduct a detailed ablation study that isolates the effects of each of DUPLEX's main components.

2 Related Work

In this work we build on top of two main bodies of research: diversity learning and universal function approximators. We organize related work accordingly.

Diversity. Diversity learning is gaining attention within the research community, due to the important benefits it yields to autonomous agents. Such benefits include generalization (Eysenbach et al., 2018), exploration (Gehring et al., 2021; Czarnecki et al., 2020), creativity (Zahavy et al., 2023b), and self-play Vinyals et al. (2019). In this work, we extend the application of diversity learning to highly-dynamic environments and multi-context settings. A central precursor to our work is DOMiNO Zahavy et al. (2023a). DOMiNO employs RL to maximize extrinsic and intrinsic diversity rewards, which are combined with Lagrange multipliers and Van der Waals (VdW) forces to balance diversity and performance. DOMiNO has two variants: one bases its diversity objective on the average of input features which grants stability at the cost of storing a moving average for every task – limiting scalability. The second variant uses the critic to estimate SFs, which solves the scalability issue at the cost of instability due to the added learned target. Intuitively, such limitations prevent DOMiNO from being effective at discovering diverse behaviors in CMDPs. Further, the Lagrange multipliers and VdW forces can limit diversity in complex environments where creative behaviors need significant exploration before providing satisfying returns (Vinyals et al., 2017). Such limitations become more evident in highly dynamic environments such as GT (see Section 5). Our algorithm is instead designed to tackle these limitations.

Quality-Diversity (QD) is another active line of work within diversity learning that involves evolutionary methods. QD aims to generate large collections of diverse and high-performing solutions, primarily through evolutionary optimization (Pugh et al., 2016; Grillotti & Cully, 2022). In QD, diversity is measured through a descriptor space defined by the user (Cully et al., 2015). Some of the main approaches in QD include MAP-Elites (Mouret & Clune, 2015; Flageat et al., 2023), local competition (Lehman & Stanley, 2011) and more recently those that combine RL with QD (Nilsson & Cully, 2021; Faldor et al., 2023; Tjanaka et al., 2022). Closer to our research, Anne & Mouret (2023) introduce a variant for MAP-Elites to learn diverse behaviors for multiple tasks within the training distribution. Our framework diverges from QD in several ways. First, QD algorithms emphasize diversity through evolutionary strategies, while our method relies on maximizing a reward objective. Second, QD requires the user to define a diversity descriptor, while we only restrict diversity to be in the near-optimal space. Third, one of our main goals is finding competitive diverse behaviors for OOD tasks and dynamics, an objective that has not been studied in Quality-Diversity (QD).

Universal Function Approximators. To make more robust estimations of the policies' expected behaviors, and thus to better quantify diversity, our work pivots around universal function approximator (UE). UE research is grounded on factoring the value estimates separating states from tasks (Schaul et al., 2015) and policies (Ma et al., 2020). Notably, Borsa et al. (2019) improve the formalization of successor features (Barreto et al., 2017), an estimation of state-action visitation, by conditioning their estimation on both task and policies. Still, approaches based on SFs are brittle in complex domains (Carvalho et al., 2021, 2023), requiring sophisticated network architectures (Mittal et al., 2020; León et al., 2022) to work effectively. Akin to Mozifian et al. (2021), we adopt a similar approach to transferring learning of SFs to continuous domains by combining them with SAC (Haarnoja et al., 2018). However, we achieve an improved estimation of the expected SFs by employing the average of the critic outputs and by adding the entropy term to the SFs learning objective. We find that the addition of the entropy term is a novel component that DUPLEX carries with and it is key to improving robustness and performance (see Appendix B).

3 Background and Notation

We briefly introduce the main building blocks of DUPLEX and the notation we adopt. First, we review basic concepts of multi-task RL and explain how it can be represented by the contextual-MDP framework. Next, we describe how universal estimators are used to enhance generalization across contexts (especially when context enumeration is impractical).

Multitask RL. We consider training a multi-policy RL agent that solves a CMDP (Hallak et al., 2015) represented as a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, P, R, \gamma, \mu_C, \mu_S \rangle$, where $\mathcal{S}, \mathcal{A}$ are the state, action spaces respectively, R is the reward function, P is the unknown transition function, μ_S is the initial state distribution conditioned on the context, and μ_C is the context distribution. We use context $c \equiv \{u, w\}$ to summarize both information about the particular dynamics u of that environment (e.g., the effect of gravity), and information about the task w in the current episode (e.g., position constraints). Every episode starts with a context $c \sim \mu_C$ and an initial state $s_0 \sim \mu_S(\cdot \mid c)$. Then, at each time step t, the agent selects an action a_t according to its policy $\pi(\cdot \mid s_t, c)$, receives a reward $r_t \sim R(s_t, a_t, c)$ and transitions to the next state $s_{t+1} \sim P(\cdot \mid s_t, a_t, c)$. Policies are characterized by their state-action occupancy, i.e., how often a policy visits a state-action pair. We consider two state-action occupancy metrics: $d^{\text{avg}}_{\pi_c}(s, a) = \lim_{T \to \infty} \frac{1}{T} \mathbb{E} \sum_{t=1}^{T} \mathbb{P}_{\pi_c}(s_t = s)\, \pi(s, a, c)$ for the average occupancy metric and $d^{\gamma}_{\pi_c}(s, a) = (1-\gamma) \mathbb{E} \sum_{t=1}^{\infty} \gamma^t \mathbb{P}_{\pi_c}(s_t = s)\, \pi(s, a, c)$ for the discounted case – where $\mathbb{P}$ is the probability measure of states at t induced by π in c. The objective of our algorithm is to find a set of diverse policies that maximize the expected return in every context $\max_{d_{\pi_c} \in \mathcal{K}} \sum_{s,a,c} r(s, a, c) d_{\pi_c}(s, a)$, where $\mathcal{K}$ is the set of admissible distributions (Zahavy et al., 2021).

Universal estimators. When aiming to solve multiple tasks, we follow the common approach of decomposing the input of the neural network to facilitate transfer learning between tasks and policies. Methods that are relevant for our work include "Universal Value Function Approximators" (UVFA) (Schaul et al., 2015), which add a task-descriptor vector w as input to a value function approximator parameterized by θ, $V_\theta(s, a, w)$. If V_θ is smooth w.r.t. w, then V_θ is expected to generalize across tasks within the training task space. Akin to previous works (Barreto et al., 2017; Zahavy et al., 2023a), we assume that every state-action pair is correlated with observable *features* known as "cumulants" $\phi(s, a, c) \in \mathbb{R}^d$. These cumulants can either be given through relevant properties within the state[2] or be learned. In a similar fashion to value functions, we can define *expected features* $\psi_{\pi_c}(s, a) = \mathbb{E}_{s', a' \sim d_{\pi_c}(s,a)} \phi(s', a', c) \in \mathbb{R}^d$, which we will refer to as average (ψ^{avg}) or discounted (ψ^{γ}) expected features when using $d^{\text{avg}}_{\pi_c}$ and $d^{\gamma}_{\pi_c}$ respectively. In the discounted case, ψ^{γ} are also known as "successor features" (SFs). This latter formalization is key to our work and also to USFA (Borsa et al., 2019), a general framework that combines UVFA-like task decomposition with the SFs, and exploits generalized policy improvement algorithm (GPI) Barreto et al. (2017). Importantly, USFA disentangles both tasks and policies by giving as input a vector z that is a representation of the current policy $z = e(\pi)$, where e is an encoding function – in our approach e encodes policies as one-hot vectors z.

4 Learning Near-Optimal Diverse Behaviors in Multi-Goal Continuous Settings

DUPLEX is a method designed to provide diverse solutions to complex tasks by simultaneously maximizing both the true reward and dissimilarity within a set of policies Π across the different environments of a CMDP. Thus, according to previous work (Zahavy et al., 2023a) we define diversity as:

Definition 4.1 (Diversity) $\text{Diversity}(\Pi)$ *is a metric of dissimilarity among policies in a set* Π *with a common goal. Formally, if* ψ_{π_i} *and* ψ_{π_j} *are a function of state-occupancy of relevant features of any two policies in* Π*, then their dissimilarity is given by* $||\psi_{\pi_i} - \psi_{\pi_j}||$*. A non-zero value of this norm indicates dissimilarity, with larger values indicating greater divergence between the policies.*

[2]For example, in a car racing scenario features could correspond to relevant information for the car such as the course velocity or if a car has crashed into another

Mathematically, diversity is defined as the sum of the minimum L2 dissimilarity norms in Π*:*

$$\text{Diversity}(\Pi) = \frac{1}{2\ size(\Pi)} \cdot \sum_{\substack{\forall \pi_i, \pi_j \in \Pi, \\ i!=j}} \min ||\psi_{\pi_i} - \psi_{\pi_j}||_2^2$$

Using this definition, we formulate our learning problem and describe DUPLEX's main contributions in the following sections. We first extend the formulation of diversity learning to context-conditioned environments, then we introduce a novel mechanism to more efficiently control the diversity-performance trade-off, and finally, we describe how to improve SFs estimation.

4.1 Context-conditioned Diversity Learning

Similar to Zahavy et al. (2023a), our objective is to maximize diversity within a set of policies Π. We express the distances between policies in Π in terms of expected features ψ, which are a function of their state-occupancy (Borsa et al. (2019); Carvalho et al. (2023)). We measure ψ distances following Def. 4.1 and use a context-based Hausdorff distance (Rockafellar, 1970) to enforce context-conditioned diversity within Π. We aim at training an RL agent that, given a context c, discovers a set of n near-optimal policies $\Pi = \{\pi_c^i\}_{i=1}^n$ by successfully optimizing:

$$\max_{\Pi} \text{ Diversity}(\Pi) \text{ s.t } d_{\pi_c} \cdot r_e \geq \rho \hat{v}_e, \quad \forall \pi_c \in \Pi \tag{1}$$

where d_{π_c} is the occupancy metric of a context-conditioned policy, r_e is the environment reward (or *extrinsic* reward), $\rho \in [0, 1]$ is a hyper-parameter defining the near-optimality region that we refer to as *optimality-ratio*, and $\hat{v}_e$ is the value of a target policy. In practice, the target policy refers to a policy in Π that ignores the diversity objective and is trained exclusively to maximize the extrinsic rewards. As in Zahavy et al. (2023a), the policy value v expresses the expected reward accumulated by a policy π. According to Definition 4.1, diversity is a distance over a set of occupancy metrics $\text{Diversity} : \{\mathbb{R}^{|S||A||C|}\}^n \to \mathbb{R}$ and Eq. 1 is defined to maximize the diversity of the occupancy metrics and preserve the near-optimality of the policies in Π.

To promote diversity between different policies, we adopt the repulsive reward from Zahavy et al. (2023a) and extend it to be conditioned on the context. More formally, given the tuple $\langle s, a, \pi^i, c \rangle$ and a cumulant function ϕ, the diversity reward that we maximize for is:

$$r_d^i(s, a, c) = \phi(s, a, c) \cdot (\psi_{\pi_c^i} - \psi_{\bar{\pi}_c^i}) \tag{2}$$

where $\psi_{\pi_c^i}$ are the expected features from policy $\pi^i \in \Pi$ at (s, a, c) and $\bar{\pi}_c^i \in \Pi$ refers to the policy with the closest expected features to π^i in (s, a, c) according to Def.4.1. Intuitively, since contexts $c \sim \mu_C$ are fixed for each training episode, this reward encourages the algorithm to train policies that visit different state-action pairs within each context c, thus, promoting context-conditioned diversity.

4.1.1 Stabilising Diversity across Different Domains

Eq. 2 expands DOMiNO's repulsive force to operate across different contexts. However, the scale of intrinsic rewards varies greatly through different environments (i.e., different contexts will inherently yield different successor features). We need to stabilize fluctuations of such rewards when working with diverse contexts. We introduce two constraints to the learning objective that modulate the intrinsic reward r_I: a *dynamic intrinsic reward factor* χ that scales r_d to target a factor of the moving average of the general extrinsic value and a *soft-lower bound* λ that limits the search of diverse policies to a near-optimal subspace. We then define the DUPLEX intrinsic reward as:

$$r_I = \lambda \cdot \chi \cdot r_d \tag{3}$$

where χ and λ are computed independently by the algorithm.

Dynamic intrinsic reward factor. χ scales r_d proportionally to the sum of extrinsic values of policies in Π. Formally, $r_d \propto v_{e_{\text{avg}}} = \frac{1}{n} \sum_{i=1}^n v_{e_{\text{avg}}}^i$ with $v_{e_{\text{avg}}}^i = \alpha_{v_{\text{avg}}} v_{e_{\text{avg}}}^i + (1 - \alpha_{v_{\text{avg}}}) r_{e,t}^i$ where, $\alpha_{v_{\text{avg}}} \in [0, 1]$ weights the contribution of the extrinsic value of the i-th policy and its immediate extrinsic reward. Finally, $r_{e,t}^i$ is the extrinsic reward at t when the agent acts under policy i. Intuitively,

we are scaling the intrinsic rewards according to the average extrinsic value that the set of policies is achieving. Hence, at each algorithmic iteration, χ is updated as follows:

$$\chi_t = \alpha_\chi \chi'_t + (1 - \alpha_\chi)\chi_{(t-1)} \tag{4}$$

where α_χ is the update rate of χ, and $\chi' = |v_{e_{\text{avg}}}/v_{d_{\text{avg}}}|(1-\rho)$ is the target value of the update, where $v_{d_{\text{avg}}}$ is the return value based on diversity intrinsic reward r_d while $v_{e_{\text{avg}}}$ is based on the extrinsic reward r_e instead. Through the optimality ratio ρ, χ minimizes the domain dependency by scaling the intrinsic rewards as a factor of the extrinsic objective and the optimality ratio ρ. Regarding the update rate α_χ, once we converged to a stable value, we kept it fixed across all experiments and domains.

Soft-lower bound. While χ preserves a relationship between extrinsic and intrinsic rewards, it does not prevent policies in Π from exploring regions of the search space that are too far from the near-optimality regions of the target policies. To limit this possibility, we explored Lagrangian-constrained optimization but found it unsatisfactory in complex domains like GT. Hence, we introduce λ to bound the near-optimal subspace for each policy using:

$$\lambda = \left\{ \sigma_k \left(\frac{v^i_{e_{\text{avg}}} - \beta \hat{v}_{e_{\text{avg}}}}{|\hat{v}_{e_{\text{avg}}} + l|} \right) \right\}_{i=1}^{n} \tag{5}$$

where $\sigma_k(x) = 1/(1 + e^{-kx})$ and $\beta \in [0, 1]$ is a hyper-parameter indicating the reward region we are interested in exploring, k regulates how "soft" is the bound, n is the number of policies, l is a small constant to prevent division by zero, and $v_{e_{\text{avg}}}$ is the average extrinsic value. Intuitively, λ limits the exploration of diverse behaviors to a near-optimal area defined by the threshold β. We find that introducing a sigmoid-based limit provides a more stable solution than Lagrangian-constrained optimization (See Figure 2). Similarly to α_χ, once we found a stable value for k, it remained fixed for all experiments.

4.2 Estimating Successor Features

Since we rely on ψ to determine policy diversity (Def. 4.1), it is fundamental to estimate ψ reliably. Additionally, as mentioned in Section 3, it is not tractable to keep ψ^{avg} for each context and task since they may be infinite. Thus building on top of UE, we exploit discounted expected features (SFs) $\psi^\gamma(s, a, z, c)$ and incorporate an extra head in the critic to estimate SFs $\tilde{\psi}^\gamma(s, a, z, c)$.

However, as in related work (Carvalho et al., 2023), we found that the critic struggles to correctly estimate ψ^γ in complex settings. Specifically, since intrinsic rewards are bootstrapped over the expected difference of successor features, the accuracy in the SFs estimation plays a major role. Taking inspiration from the SAC algorithm (Haarnoja et al., 2018), we incorporate an entropy term in the expected features objective to account for the stochastic component within the learned policies.

For policy i in context c, our SFs at s_t are computed as:

$$\psi^{\gamma,i}(s_t, a_t, c) = \phi_t + \mathbb{E}_{\pi_c} \sum_{k=t+1}^{\infty} \gamma^{k-t} \left[\phi_k + \alpha_H H\big(\pi^i_c(\cdot|s, c)\big) \right] \tag{6}$$

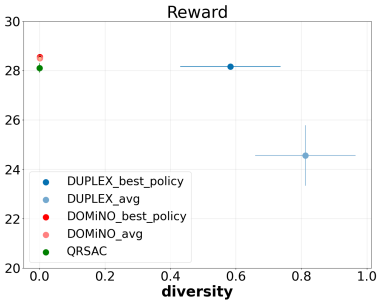
where α_H is the entropy weight. Intuitively, unless the critic is confident that the actor will visit a state-action pair more frequently than others, the network will be encouraged to estimate that the policy will maximize the entropy. We formulate our temporal difference loss for the critic output $\tilde{\psi}^\gamma$ to account for the entropy term as:

$$\text{TD}_\text{H}(\tilde{\psi}^\gamma_{\theta_j}(\pi^z_c)) = \mathop{\mathbb{E}}_{(s,a)\sim\pi^z_c} \left[\frac{1}{2} \left(\tilde{\psi}^\gamma_{\theta_j}(s, a, c, z) - y(\phi, s', c, z) \right)^2 \right] \tag{7}$$

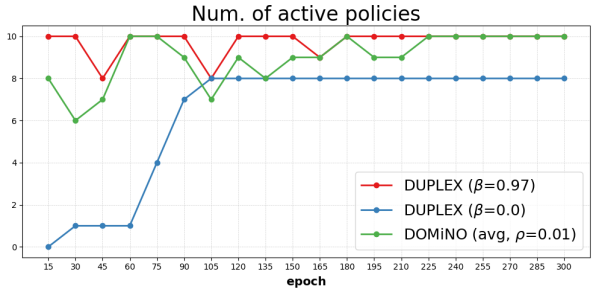
where, similar to SAC, we have two critic estimates $j = \{1, 2\}$, but instead of taking the minimum of the two, our target is obtained from the average of the estimates. Then, we define $y(\phi, s', c, z)$ as:

$$y(\phi, s', c, z) = \phi(t) + \gamma \left(\mathop{\text{avg}}_{j=1,2} \tilde{\psi}_{\theta_{\text{targ},j}}(s', \tilde{a}'_z, c) - \alpha \log \pi^z_\omega(\tilde{a}'_z|s', c) \right) \tag{8}$$

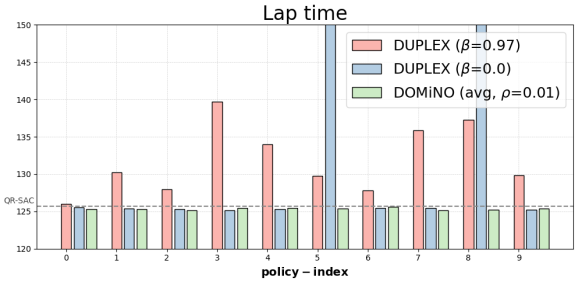
with $\tilde{a}'_z \sim \pi^z_\omega(\cdot|s', c)$. The motivation lies in the different impacts of overestimation of values and successor features. When predicting values, overestimation directly influences the policy since it



(a) Rewards vs Diversity



(b) Active Policies



(c) Lap Times

Figure 2: Results in GT. a) is the reward-diversity trade off, DUPLEX/DOMiNO_best_policy refers to $\max_{\mathrm{r_e}}(\pi \in \Pi)$ of every run, while DUPLEX/DOMiNO_avg refers to $\mathrm{avg}_{\mathrm{r_e}}(\pi \in \Pi)$ of that run. Differently from DOMiNO, DUPLEX is able to generate *both* competitive and diverse behaviors. Figure b) illustrates the number of policies that finish laps (active). Figure c) min lap times of the diverse policies. Note that the number of active policies is important to showcase that all policies are actively searching for near-optimal diverse behaviors through different epochs. Applying our soft lower bound ($\beta \neq 0$) is key to focusing diversity in the region of interest.

is trained to maximize advantage. However, in DUPLEX, SFs assume a distinct role: the policy is not incentivized to select actions with the highest successor features but rather those that differ the most from the SFs of the remaining policies (subject to not sacrificing too much performance, equation 1). Consequently, precision becomes of greater significance for ψ^{γ}, and taking the average emerges as a more reliable solution.

The introduction of entropy improves stability in the estimation of SFs through most of the training process and, as we report in the experimental section, it supports OOD generalization. However, as learning stabilizes, we typically encounter phases where the estimated difference between policies rapidly increases its order of magnitude. For this reason, we also introduce a fixed upper bound to α_{χ} when estimating SFs. Details on the upper bound, DUPLEX pseudocode, and hyper-parameters are included in Appendix C and D.

Summarizing, DUPLEX enhances diversity learning in context-conditioned environments through four key components: (i) *dynamic intrinsic reward factor* (χ) that balances diversity with extrinsic rewards and avoids environment-specific tuning; (ii) *soft lower bound* (λ) that constrains policies to optimize for diversity only within a target near-optimal region; (iii) *entropy regularization* (α_H) in successor feature estimation to improve robustness; and (iv) averaging over critic estimates for computing diversity rewards. In Section 5 we demonstrate how these features improve both performance and diversity of the learned policies.

5 Experiments

The objectives of our experiments are to demonstrate that DUPLEX: 1) is the first successful method in learning diverse driving styles in highly-dynamic physics simulators such as GT (Figure 2); 2) when

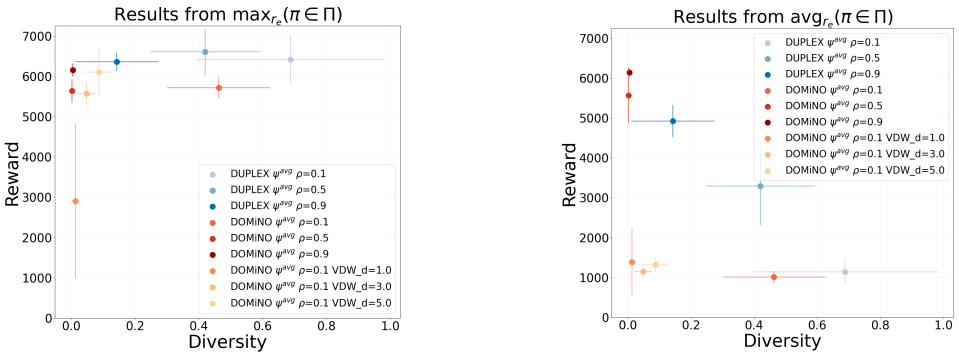


Figure 3: Walker single-task results. Both DUPLEX and DOMiNO do not estimate SFs and use $d^{\text{avg}}_{\pi_c}$. On the left, we show results from the best policy in each variant while on the right, the average reward across all policies. Dots and lines represent mean and standard deviation, respectively. It is worth reporting that the vanilla SAC implementation scores 5576.24±256.783 when evaluated at the same training epoch.

compared against previous state-of-the-art diversity approaches in RL, yields a better performance vs. diversity trade-off in canonical physics simulators (Figure 3); 3) exhibits diverse and effective behaviors in OOD dynamics and tasks (Figure 4); and finally, 4) improves on previous baselines in estimating SFs (Figure 5 and 6). All results in MuJoCO are obtained from five training runs. GT experiments are more computationally demanding requiring a minimum of seven days to converge, so the results are obtained from three training runs.

Baselines. Our main baseline is DOMiNO (Zahavy et al., 2023a), which is the state-of-the-art of diversity learning in RL. Then, since DOMiNO is not designed for OOD generalization, our experiments in OOD settings include as baselines the two most popular frameworks within UE literature: UVFA (Schaul et al., 2015) and USFA (Borsa et al., 2019). Note that, throughout the section and given the two versions of DOMiNO, we will refer to DOMiNO$_{avg}$ as the version of the algorithm using averaged cumulants as successor features, and as DOMiNO$_{est}$ as the version of the algorithm that estimates SFs.

Benchmarks. We use three benchmarks. The first is a racing track in GT, where the goal is to have different driving styles while completing fast laps. Achieving diverse and competitive driving styles in such realistic simulator Wurman et al. (2022) can unlock multiple applications both in gaming and self-driving. Second, a set of experiments is conducted in the MuJoCo's Walker2D and Ant environments where we compare DUPLEX against DOMiNO$_{avg}$ in the same environments used in Zahavy et al. (2023a). Finally, in the last benchmark, we evaluate our approach in a multi-context version of MuJoCo's Walker2D and Ant environments. We extend the default versions of these scenarios to include a three-dimensional context c where the inputs represent: the gravity coefficient and, the upper and lower height thresholds we would like the agent to respect while moving. The goal of this benchmark is to find diverse policies in OOD contexts while guaranteeing competitive performance in task execution. Note that, to have a fair comparison against the baselines, only the last benchmark configures DUPLEX to estimate SFs through ψ^{γ}. Additional details about the metrics and the environments are reported in Appendix A.

Results in GranTurismo™ 7. As in Wurman et al. (2022), we use QRSAC as the base RL algorithm for both DOMiNO$_{avg}$ and DUPLEX. Figure 2 (top-left) shows that only DUPLEX learns diverse policies in the near-optimal region. Figure 2(c) illustrates how this diversity translates into diverse lap times. Moreover, Figure 2(b) provides evidence of the benefits that our lower bound β formulation carries with it. When using $\beta = 0$, the most diverse policies are "busy" finding different ways to *not* finish the track. Furthermore, we see DUPLEX results oppose to DOMiNO that instead collapses all policies to a similar behavior even when using an extremely forgiving value of the optimality ratio ρ. It is worth mentioning, that DOMiNO fails to provide diversity even if it implements a Lagrangian constraint optimization objective that should support more adaptability. For example,

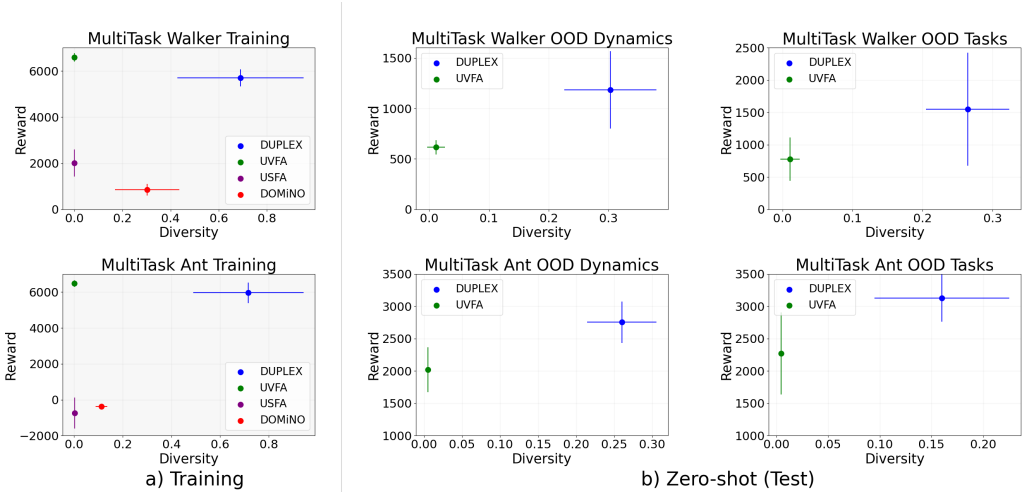


Figure 4: Results in MuJoCo Walker (Top) and Ant (bottom) multitask environments. On the y-axis, we report the reward accumulated by the agent while on the x-axis normalized diversity according to Def. 4.1. Ideally, we want to be as close as possible to the top right region. OOD dynamics refers to gravities that are at least 40% stronger or weaker than the strongest and weakest gravities seen in training, respectively. OOD tasks represent tasks where the agent needs to walk forward below a height 20% lower than the lowest height seen in training. Only DUPLEX and UVFA are evaluated in the Test since the other algorithms already failed in training.

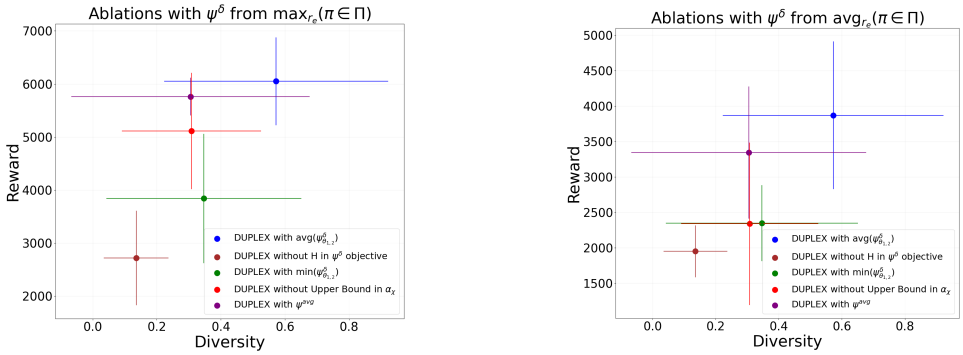


Figure 5: Ablation of DUPLEX in Multitask Walker. Reward and diversity scores of the best trained policy are reported on the left, while averaged values of all policies are on the right.

when configuring $\rho = 0.01$, DOMiNO should consider as "acceptable" any diverse policy achieving 1% of the value of the target policy $\hat{v}_{e_{\text{avg}}}$. But still, the algorithm struggles to find diverse policies.

Results in MuJoCo. To further validate the contribution of DUPLEX to diversity learning, we include a direct comparison of DUPLEX and DOMiNO$_{avg}$ while configuring the Walker2D environment that the authors present in Zahavy et al. (2023a). Within this set of experiments, both algorithms have been configured to work with ψ^{avg} which fosters the opportunity to ablate the benefits of our constrained r_I (Eq. 3) isolated from the impact of estimating SFs. Figure 3 includes the results from both frameworks with multiple optimality ratios ρ. Moreover, to provide an exhaustive evaluation, we also configure DOMiNO$_{avg}$ with various Van Der Waal (VdW) distance hyper-parameters. Such a parameter modulates the desired distance between all the policies in Π (see Zahavy et al. (2023a) for details). Finally, DUPLEX is able to find a better Pareto-frontier for the quality-diversity objective (Figure 3 right), while achieving better near-optimal policies than the baseline (Figure 3 left).

Results in multi-context MuJoCo. Figure 4 reports the results of our baselines in multi-context scenarios when evaluated within- and out-of training distribution. In the within-distribution setting

(Figure 4 left-hand side) only DUPLEX shows good performance and diversity while USFA and DOMiNO fail at both learning competitive policies and providing diversity. As we report in the USFA ablation (B), DUPLEX succeeds in such benchmark due to the entropy term introduced to make the estimation of SFs more robust (Eq. 7). Additionally, note that USFA and UVFA do not optimize for diversity, and their diversity score is expected to be close to zero. Then, Figure 4 (right-hand side) reports the generalization effectiveness of DUPLEX and UVFA when operating out-of training distribution. Note that USFA and DOMiNO do not qualify in this setting as they failed to learn already in the training set. We observe that in all the OOD scenarios DUPLEX outperforms UVFA and, most importantly, it covers a bigger range of diversity values while completing the tasks.

Ablation of successor-feature estimation. Figure 5 illustrates the contribution of ψ^{δ} and the different features incorporated by DUPLEX. We have evidence that all features that compose DUPLEX are key to guaranteeing robust learning of diverse behaviors in highly-dynamic environments and multi-context settings. Notably, USFA can also effectively solve multi-context MuJoCo environments when akin to DUPLEX, adds the entropy term (see Appendix B).

6 Conclusions

DUPLEX provides a powerful technique that enables learning of diverse near-optimal behaviors. Our experimental session shows that (differently from other baselines) DUPLEX (i) learns effective diverse behaviors in hyper-realistic complex domains (such as GranTurismo) and (ii) generalizes to OOD in challenging multi-context MuJoCo environments by demonstrating competitive diverse behaviors.

Limitations and Future Directions. There is potential to refine and enhance the methodology presented in this work. Nevertheless, we perceive the current limitations of DUPLEX not as setbacks, but as exciting opportunities that pave the way for new research avenues. For example, (1) can we generate more than twenty diverse policies and still guarantee meaningful diversity? (2) DUPLEX carries an additional cost due to the need to compute SF distances of each policy pair in our set, thus how can we improve sample efficiency and keep a low computational footprint? (3) DUPLEX does not impose any exploration strategy on each policy, and thus, can we control in what measure a particular policy will be different? And finally, (4) can we combine the strengths of different on a single solution? This latter question is very interesting to us. We suppose that once diverse policies have been learned, each of them retains different useful skills (e.g. different commuting routes) that can be combined dynamically to reconstruct a single agent capable of acting optimally and tackling unseen portions of the state-space.

Finally, the presented approach holds significant promise for advancing research in diverse policy learning, scalability, and efficient transfer and exploration (Parker-Holder et al., 2020; Peng et al., 2020) across varied contexts. We note also that RL algorithms may also risk reinforcing biases and require careful implementation to ensure transparency and fairness. Diversity-learning algorithms might result in training policies with potentially unethical behaviors that are harder to predict. These are reasons that motivate us to pursue controllability while learning diverse policies in this and future works.

References

Altman, E. *Constrained Markov decision processes*, volume 7. CRC press, 1999.

Anne, T. and Mouret, J.-B. Multi-task multi-behavior map-elites. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation*, pp. 111–114, 2023.

Barreto, A., Dabney, W., Munos, R., Hunt, J. J., Schaul, T., van Hasselt, H. P., and Silver, D. Successor features for transfer in reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Borkar, V. S. An actor-critic algorithm for constrained markov decision processes. *Systems & control letters*, 54(3):207–213, 2005.

Borsa, D., Barreto, A., Quan, J., Mankowitz, D. J., van Hasselt, H., Munos, R., Silver, D., and Schaul, T. Universal successor features approximators. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL `https://openreview.net/forum?id=S1VWjiRcKX`.

Carvalho, W., Lampinen, A. K., Nikiforou, K., Hill, F., and Shanahan, M. Feature-attending recurrent modules for generalization in reinforcement learning. *CoRR*, abs/2112.08369, 2021. URL https://arxiv.org/abs/2112.08369.

Carvalho, W., Filos, A., Lewis, R. L., Lee, H., and Singh, S. Composing task knowledge with modular successor feature approximators. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/pdf?id=DrtSx1z40Ib.

Cully, A., Clune, J., Tarapore, D., and Mouret, J.-B. Robots that can adapt like animals. *Nature*, 521(7553):503–507, May 2015. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature14422. URL http://arxiv.org/abs/1407.3501. arXiv:1407.3501 [cs, q-bio].

Czarnecki, W. M., Gidel, G., Tracey, B., Tuyls, K., Omidshafiei, S., Balduzzi, D., and Jaderberg, M. Real world games look like spinning tops. *Advances in Neural Information Processing Systems*, 33:17443–17454, 2020.

Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I., Legg, S., and Kavukcuoglu, K. IMPALA: scalable distributed deep-rl with importance weighted actor-learner architectures. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1406–1415. PMLR, 2018. URL http://proceedings.mlr.press/v80/espeholt18a.html.

Eysenbach, B., Gupta, A., Ibarz, J., and Levine, S. Diversity is All You Need: Learning Skills without a Reward Function. September 2018. URL https://openreview.net/forum?id=SJx63jRqFm.

Faldor, M., Chalumeau, F., Flageat, M., and Cully, A. Map-elites with descriptor-conditioned gradients and archive distillation into a single policy. In Silva, S. and Paquete, L. (eds.), *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 2023, Lisbon, Portugal, July 15-19, 2023*, pp. 138–146. ACM, 2023. doi: 10.1145/3583131.3590503. URL https://doi.org/10.1145/3583131.3590503.

Flageat, M., Lim, B., and Cully, A. Multiple hands make light work: Enhancing quality and diversity using map-elites with multiple parallel evolution strategies. *CoRR*, abs/2303.06137, 2023. doi: 10.48550/arXiv.2303.06137. URL https://doi.org/10.48550/arXiv.2303.06137.

Fu, W., Du, W., Li, J., Chen, S., Zhang, J., and Wu, Y. Iteratively learn diverse strategies with state distance information. *Advances in Neural Information Processing Systems*, 36, 2023.

Gehring, J., Synnaeve, G., Krause, A., and Usunier, N. Hierarchical Skills for Efficient Exploration. November 2021. URL https://openreview.net/forum?id=NbaEmFm2mUW.

Grillotti, L. and Cully, A. Relevance-guided unsupervised discovery of abilities with quality-diversity algorithms. In Fieldsend, J. E. and Wagner, M. (eds.), *GECCO '22: Genetic and Evolutionary Computation Conference, Boston, Massachusetts, USA, July 9 - 13, 2022*, pp. 77–85. ACM, 2022. doi: 10.1145/3512290.3528837. URL https://doi.org/10.1145/3512290.3528837.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1856–1865. PMLR, 2018. URL http://proceedings.mlr.press/v80/haarnoja18b.html.

Hallak, A., Di Castro, D., and Mannor, S. Contextual markov decision processes. *arXiv preprint arXiv:1502.02259*, 2015.

Lehman, J. and Stanley, K. O. Evolving a diversity of virtual creatures through novelty search and local competition. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, GECCO '11, pp. 211–218, New York, NY, USA, July 2011. Association for Computing Machinery. ISBN 978-1-4503-0557-0. doi: 10.1145/2001576.2001606. URL https://dl.acm.org/doi/10.1145/2001576.2001606.

León, B. G., Shanahan, M., and Belardinelli, F. In a nutshell, the human asked for this: Latent goals for following temporal specifications. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=rUwm9wCjURV.

Ma, C., Ashley, D. R., Wen, J., and Bengio, Y. Universal successor features for transfer reinforcement learning. *CoRR*, abs/2001.04025, 2020. URL https://arxiv.org/abs/2001.04025.

Mittal, S., Lamb, A., Goyal, A., Voleti, V., Shanahan, M., Lajoie, G., Mozer, M., and Bengio, Y. Learning to combine top-down and bottom-up signals in recurrent neural networks with attention over modules. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6972–6986. PMLR, 2020.

Mouret, J. and Clune, J. Illuminating search spaces by mapping elites. *CoRR*, abs/1504.04909, 2015. URL http://arxiv.org/abs/1504.04909.

Mozifian, M., Fox, D., Meger, D., Ramos, F., and Garg, A. Generalizing successor features to continuous domains for multi-task learning. 2021.

Nilsson, O. and Cully, A. Policy gradient assisted map-elites. In Chicano, F. and Krawiec, K. (eds.), *GECCO '21: Genetic and Evolutionary Computation Conference, Lille, France, July 10-14, 2021*, pp. 866–875. ACM, 2021. doi: 10.1145/3449639.3459304. URL https://doi.org/10.1145/3449639.3459304.

Parker-Holder, J., Pacchiano, A., Choromanski, K. M., and Roberts, S. J. Effective diversity in population based reinforcement learning. *Advances in Neural Information Processing Systems*, 33: 18050–18062, 2020.

Peng, Z., Sun, H., and Zhou, B. Non-local policy optimization via diversity-regularized collaborative exploration. *arXiv preprint arXiv:2006.07781*, 2020.

Pugh, J. K., Soros, L. B., and Stanley, K. O. Quality diversity: A new frontier for evolutionary computation. *Frontiers Robotics AI*, 3:40, 2016. doi: 10.3389/frobt.2016.00040. URL https://doi.org/10.3389/frobt.2016.00040.

Rockafellar, R. T. *Convex Analysis*. Princeton Landmarks in Mathematics and Physics. Princeton University Press, 1970. ISBN 978-1-4008-7317-3.

Schaul, T., Horgan, D., Gregor, K., and Silver, D. Universal value function approximators. In *International conference on machine learning*, pp. 1312–1320. PMLR, 2015.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Szepesvári, C. Constrained MDPs and the reward hypothesis. *Musings about machine learning and other things (blog)*, 2020.

Tjanaka, B., Fontaine, M. C., Togelius, J., and Nikolaidis, S. Approximating gradients for differentiable quality diversity in reinforcement learning. In Fieldsend, J. E. and Wagner, M. (eds.), *GECCO '22: Genetic and Evolutionary Computation Conference, Boston, Massachusetts, USA, July 9 - 13, 2022*, pp. 1102–1111. ACM, 2022. doi: 10.1145/3512290.3528705. URL https://doi.org/10.1145/3512290.3528705.

Todorov, E., Erez, T., and Tassa, Y. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

Vinyals, O., Ewalds, T., Bartunov, S., Georgiev, P., Vezhnevets, A. S., Yeo, M., Makhzani, A., Küttler, H., Agapiou, J. P., Schrittwieser, J., Quan, J., Gaffney, S., Petersen, S., Simonyan, K., Schaul, T., van Hasselt, H., Silver, D., Lillicrap, T. P., Calderone, K., Keet, P., Brunasso, A., Lawrence, D., Ekermo, A., Repp, J., and Tsing, R. Starcraft II: A new challenge for reinforcement learning. *CoRR*, abs/1708.04782, 2017. URL http://arxiv.org/abs/1708.04782.

Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell, R., Ewalds, T., Georgiev, P., et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.

Wurman, P. R., Barrett, S., Kawamoto, K., MacGlashan, J., Subramanian, K., Walsh, T. J., Capobianco, R., Devlic, A., Eckert, F., Fuchs, F., et al. Outracing champion gran turismo drivers with deep reinforcement learning. *Nature*, 602(7896):223–228, 2022.

Zahavy, T., O'Donoghue, B., Desjardins, G., and Singh, S. Reward is enough for convex MDPs. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*,

pp. 25746–25759, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/d7e4cdde82a894b8f633e6d61a01ef15-Abstract.html.

Zahavy, T., Schroecker, Y., Behbahani, F. M. P., Baumli, K., Flennerhag, S., Hou, S., and Singh, S. Discovering policies with domino: Diversity optimization maintaining near optimality. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023a. URL https://openreview.net/pdf?id=kjkdzBW3b8p.

Zahavy, T., Veeriah, V., Hou, S., Waugh, K., Lai, M., Leurent, E., Tomasev, N., Schut, L., Hassabis, D., and Singh, S. Diversifying AI: Towards Creative Chess with AlphaZero. August 2023b. doi: 10.48550/arXiv.2308.09175. URL http://arxiv.org/abs/2308.09175. arXiv:2308.09175 [cs].

A Environment details

In MuJoCo environments, we measure diversity directly on the agent's observations, which is represented as a feature vector containing the angles and velocities of the agent's joints. In GT, diversity is measured through the most characterizing descriptors of a driving style: action intensities (brake, throttle, and steering values), wheel angles, and distances to the track edges.

Note that these diversity features were chosen to align with each environment's objectives. Nevertheless, we can maximize the diversity of any feature in the observation. Such an arbitrary selection of features intuitively leads to different optimization objectives and thus learned strategies. For instance, in GT, we could maximize diversity on the frequency of hitting other cars, and this would lead to a population of policies that are all driving as fast as possible while exploring the spectrum of aggressiveness: ranging from a very timid behavior (letting opponents pass to avoid contact) to a very aggressive one (hitting cars intentionally if the collision does not result in a significant speed loss).

We employ three types of environments to conduct our experiments. We use the canonical MuJoCo benchmark for single-task experiments; then the MountainTimeTrial track in GranTurismo™7 game for the GT experiments; and finally a MuJoCo wrapper environment for the multi-context experiments. Such a wrapper augments the observation of the agent by adding a context vector of three elements $c = \langle g, l, u \rangle$ representing the gravity coefficient g, the lower height the agent should walk below of l and the upper height the agent should walk above of u. It is worth noticing that we excluded context configurations that would not be possible to solve, i.e. episodes were designed to let the agent learn to either walk low or jump high. At training time, in the Walker2D environment, the context features $\langle g, l, u \rangle$ take values in $g \in [6, 15]$, $l \in [0.8, 1.2]$ and $u \in [1.25, 1.8]$. While in the Ant environment features take value in $g \in [6, 15]$, $l \in [0.35, 0.8]$ and $u \in [0.9, 1.2]$.

In the OOD scenarios, we used two out-of-distribution contexts for the Walker2D and the Ant environments respectively. In the former, we used two values of the gravity coefficient and a lower bound constraint, namely $c_0 = \langle 3, 0.6, None \rangle$ and $c_1 = \langle 24, 0.6, None \rangle$, while in the latter we used $c_0 = \langle 3, 0.2, None \rangle$ and $c_1 = \langle 24, 0.2, None \rangle$

B Ablation in USFA

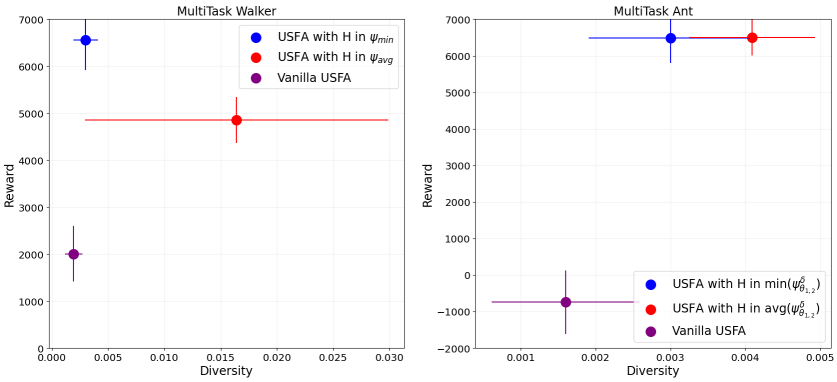


Figure 6: Ablation on USFA, we see that incorporating entropy to the learning objective of the successor features estimator as in DUPLEX yields a significant boost to accumulate rewards.

In this section, we ablate the addition of the entropy term to the SFs objective function. In particular, we extrapolate such component from the others introduced by DUPLEX, and we evaluate the entropy term in isolation within the original USFA framework. The assumption that we validate with this ablation study is that Equation 7 has a broader impact and that it makes the estimation of SFs more robust. Figure 6 illustrates our findings, we report that the entropy term enables effective learning in multi-context MuJoCo environments and significantly improves performance w.r.t. vanilla USFA.

Importantly, Figure 6 illustrates that, using $avg(\phi_{\theta_{1,2}})$ instead of the $min(\phi_{\theta_{1,2}})$, is not beneficial for USFA. Note that differently from DUPLEX, USFA infers extrinsic rewards from SFs. Thus, in USFA an overestimation of SFs (i.e., overestimating how often the policy visits some state-action

Algorithm 1 DUPLEX

1: Initialize dataset with random exploration, set up critics θ and policy parameters ω. and initialize moving averages $\{v^{\pi^i}_{\text{avg}}\}_{i=1}^{n} = 0$ reward weight $\kappa = 0$, $\chi = 1$ and $\text{step} = 0$
2: **while** not converged **do**
3: Generate more data samples
4: Sample batch of m transitions $\{\text{t}_j\}_{j=1}^{m}$, where a transition is $\text{t}_j = (s^j, a^j, c^j, \phi(s^j, a^j, c^j), z^j, r^j_e, s'^j)$
5: Use critic to get $\tilde{v}^{\pi^i}_{e_j}$, $\tilde{v}^{\pi^i}_{i_j}$ and $\tilde{\psi}^{\gamma}(s, a, c, z)$ for every $\pi^z \in \Pi$. Where $\tilde{\psi} = \underset{\text{h}=1,2}{\text{avg}}\, \tilde{\psi}_{\theta_{\text{h}}}$
6: Compute $r_I(s)$ from $\tilde{\psi}, z, \phi(s, a, c)$ with Equation 3
7: Compute extrinsic temporal difference (TD) errors and q-values: TD_e, Q_e
8: For sampled z get TD error with entropy on successor features with Equation 7: TD_H
9: Compute intrinsic TD errors and q-values: TD_I, Q_I
10: Combine q-values $Q = \sigma(\kappa)Q_e + (1 - \sigma(\kappa))Q_I$
11: **if** $\text{step} >$ critic warming-up period **then**
▷ *Policy loss:*
12: $\nabla_\omega \sum_j \Big(\min_{h=1,2} Q_{\theta_i}(s, \pi_\omega(\cdot|s^j, c^j, z^j)) - \alpha \log \pi_\omega \left(\pi_\omega(\cdot|s^j, c^j, z^j) \middle| s^j, c^j, z^j \right) \Big)$
13: Calculate weighted critic loss: $b_v(\text{TD}_e + \text{TD}_i) + b_\psi(\text{TD}_H)$
14: **if** Constrained optimization **then**
▷ *Lagrange loss*
15: $\sum_{i=1}^{n} \sigma(\kappa^i)(v^i_{\text{avg}} - \rho \hat{v}_{\text{avg}})$
16: Update κ
17: Update θ and ω
18: Update $v^i_{e_{\text{avg}}} = \alpha_{v_{\text{avg}}} v^i_{e_{\text{avg}}} + (1 - \alpha_{v_{\text{avg}}}) r^i_{e,t}$, $v^i_{d_{\text{avg}}} = \alpha_{v_{\text{avg}}} v^i_{d_{\text{avg}}} + (1 - \alpha_{v_{\text{avg}}}) r^i_{d,t}$, χ with Eq. 4 and $\text{step} += 1$

pairs) yields an overestimation of rewards. Such an overestimation motivates our design choice of using the avg of two critics in DUPLEX. In fact, since DUPLEX relies on the distance between SFs to estimate intrinsic rewards, we achieve a more accurate representation of intrinsic rewards by taking the average of the two estimates.

C Training DUPLEX

DUPLEX follows an actor-critic approach, where a policy network determines actions to take while a critic network estimates values and, in the case of multiple contexts, also successor features. Figure 1 depicts the workflow of our algorithm. The networks have three independent input encoders for the state, policy, and context respectively. Since we use SAC as the base RL algorithm, we employ two critic networks and take the min of the two for value estimates $\tilde{v}^{\gamma}$ and the average for $\tilde{\psi}^{\gamma}$ (Eq. 8).

The training procedure of DUPLEX is described in Alg. 1. After warming up to collect data, we start sampling transitions t and the learning routine (Line 4). Since we need to estimate the distance between all policies to compute the intrinsic rewards, we broadcast the input of the critic to have the SF estimates for all policies in every transition and do a forward pass through the critic (Line 5). Then, we obtain r_I with Eq. 3 and compute the temporal-difference errors (Line 7 – Line 9). Similarly, after a short warm-up period, we compute the policy loss (Line 12). Then, depending on if we enable constrained optimization, we update the Lagrange multipliers (Line 15). Finally, we update the network weights and the average policy values to update the dynamic factor χ (Line 17 and Line 18).

Upper bound on α_χ. As described in Section.4, when the critic learns to estimate SFs, we typically observe events where the estimated difference between policies rapidly increases its order of magnitude. Such a shift in the magnitude of expected distances results in a rapid increase of intrinsic rewards, larger value estimates, and ultimately, the divergence of the critics output. To address such magnitude shifts, we modify the update rate of χ:

$$\alpha_\chi = \begin{cases} 1 & \text{if } |v_{d_{\text{avg}}}| > \epsilon_\chi |v_{e_{\text{avg}}}| \\ \alpha_{\text{default}} & \text{otherwise} \end{cases} \tag{9}$$

Intuitively, we want r_I to motivate the agent to explore increasingly diverse behaviors without stagnating the extrinsic rewards. For this reason, we need α_χ to be a very low value – we use $1\text{e}{-5}$

across all our experiments. An exception is made when a sudden drastic change in $\tilde{\psi}$ could cause the critic to diverge. We automatically detect those changes through a threshold $\epsilon_\chi |r_e^{\text{avg}}|$, where ϵ_χ is a scalar, and perform a one-step update to prevent r_I from destabilizing the learning process. Similarly to other hyper-parameters, once we found a stable value for ϵ_χ it remained fixed for all experiments and benchmarks.

Constrained optimization. Within the algorithm framework, we allow Langrangian constraint optimization Borkar (2005) to be contingent to specific domains. When constrained optimization is configured, we establish a multi-goal objective Altman (1999); Szepesvári (2020) using Lagrange multipliers κ, whose objective is to maximize $\kappa(\rho\hat{v}_e - d_\pi \cdot r_e)$. Intuitively, this objective yields a multiplier κ that decreases in value when the policy is satisfying the optimality constraint, and increases otherwise. However, we empirically find that while Lagrangian constrained optimization might be beneficial in canonical domains, it is not effective in complex environments such as GT and fails to find competitive diverse behaviors. Nevertheless, Equation (4) and 5 make our r_i less reliant on constraint optimization than previous approaches. For these reasons, we implement constrained optimization in DUPLEX by keeping κ fixed by default and making it learnable as an optional feature.

Remark C.1 *DUPLEX is designed so that only two hyper-parameters significantly impact its performance: ρ and β. The former regulates how much the user wants to weight the intrinsic rewards w.r.t. the extrinsic reward, while the latter defines the near-optimality region that the agent should explore when learning diverse policies.*

D Hyper-parameters and Computational Analysis

D.1 Hyper-parameters

Table 2 lists DUPLEX hyper-parameters and neural network configuration.

Table 1: Hyper-parameters of DUPLEX.

HYPER-PARAMETER	MUJOCO	GT
N	10	10
ρ	$[0.9, 0.7]$	$[0.88, 0.8]$
β	0.5	0.97
α_χ	$1e^{-5}$	$1e^{-5}$
ϵ_χ	20	20
k FROM SIGMOID	5	5
$\alpha_{\psi^{\text{avg}}}$	0.9999	0.9999
$\alpha_{v_{\text{avg}}}$	0.999	0.999
$\hat{v}$ SAMPLE RATE	0.3	0.3
κ LEARNING RATE	$1e^{-2}$	-
SAC/QRSAC LEARNING RATE	$3e^{-4}$	$2.5e^{-4}$
TARGET CRITIC TARGET SOFT UPDATE RATE	0.005	0.005
CLIP GRADIENT GLOBAL NORM	3.0	
OBS ENCODER SIZE	256	2048
CONTEXT ENCODER SIZE	64	-
POLICY ENCODER SIZE	256	256
TORSO SIZE	2x256	4x2048
CRITIC HEAD HIDDEN LAYER SIZE	256	2048
SAC ENTROPY LEARNING RATE	$1e^{-4}$	FIXED ENTROPY

D.2 Computational Analysis

Given the additional components that DUPLEX requires to stabilize training, we report in the following table the computational cost of the update function of the core learning algorithm while varying the number of used policies. We compare our approach to DOMiNO and vanilla SAC and demonstrate that the additional cost is negligible when considering related work, and does not increase with the number of policies. Execution times of the selected algorithms are reported in milliseconds

and have been evaluated on the Walker2D MuJoCo environment rolled out on a 13th Gen Intel(R) Core(TM) i9-13900HX and NVIDIA GeForce RTX 4080 Laptop GPU.

Table 2: Execution time of the algorithms update function. Time is reported in milliseconds when computing diversity for 2, 4, 10, and 20 policies simultaneously.

ALGORITHM	1	2	5	10	20
DUPLEX		55.78 ± 1.51	55.72 ± 1.82	55.21 ± 1.37	56.73 ± 2.06
DOMINO		52.64 ± 1.88	53.68 ± 1.22	54.64 ± 1.36	56.08 ± 1.63
SAC	41.01 ± 2.05				

E Hardware

We ran our experiments on an internal cluster designed for distributed training. Each job has been configured with a trainer (to update the models) and N rollout workers (to collect data).

MuJoCo. In this environment we use one trainer and one rollout worker – none of them equipped with a GPU. The former uses 7.7 vCPUs and 8Gi of RAM, while the latter uses 1 vCPUs and 2Gi of RAM. The duration of each experiment is two days.

GT. In this environment, each job is configured with one trainer and ten rollout workers. The trainer is equipped with 7.7 vCPUs, 1 Nvidia V100, and 55Gi of RAM, while each rollout worker features 2000m vCPUs, 3328Mi of RAM, and 1 PlayStation4. The duration of each experiment is seven days.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We listed and described claims made in this work both in the abstract and the introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discussed the limitations of our framework in Section 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: The research described in this paper does not make theoretical claims.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Our main contribution is the introduction of a novel iterative algorithm: DUPLEX. We provide (i) flow diagrams to illustrate how the building blocks are connected; (ii) describe each component and equation in detail in the methodological section; (iii) outline the pseudocode of DUPLEX to improve readability; and (iv) details of our benchmarks both in Section 5 and Appendix A – along with the hyper-parameters used in Appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The data and code used for empirical results and analysis are proprietary.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We include all the details to reproduce our results in Section 5 and Appendix A. Additional details, including the hyper-parameters used, are available in (Appendix D).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report average results and error bars of our experiments. Figure 2(b) and Figure 2(c) represent an exception as we report the performance of the most competitive agent for each baseline.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We include details about the compute resources and hardware setup in Appendix E.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes, we believe the research on DUPLEX and empirical analysis conform to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no direct societal impact of the work described in the paper. We focus on learning diverse policies in dynamic domains to advance the field of RL.

    Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does include the release of data or models and as such does not pose a risk of misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have provided citations and references to existing MuJoCo and GT assets, including GT's trademark in Section 1.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not include the release of any new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The research described in this paper does not include any work with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: IRB approval was not required as the research does not include any work with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.