MAP-Elites with Descriptor-Conditioned Gradients and Archive Distillation into a Single Policy

Maxence Faldor
m.faldor22@imperial.ac.uk
Imperial College London
London, United Kingdom

Félix Chalumeau
f.chalumeau@instadeep.com
InstaDeep
Cape Town, South Africa

Manon Flageat
manon.flageat18@imperial.ac.uk
Imperial College London
London, United Kingdom

Antoine Cully
a.cully@imperial.ac.uk
Imperial College London
London, United Kingdom

ABSTRACT

Quality-Diversity algorithms, such as MAP-Elites, are a branch of Evolutionary Computation generating collections of diverse and high-performing solutions, that have been successfully applied to a variety of domains and particularly in evolutionary robotics. However, MAP-Elites performs a divergent search based on random mutations originating from Genetic Algorithms, and thus, is limited to evolving populations of low-dimensional solutions. PGA-MAP-Elites overcomes this limitation by integrating a gradient-based variation operator inspired by Deep Reinforcement Learning which enables the evolution of large neural networks. Although high-performing in many environments, PGA-MAP-Elites fails on several tasks where the convergent search of the gradient-based operator does not direct mutations towards archive-improving solutions. In this work, we present two contributions: (1) we enhance the Policy Gradient variation operator with a descriptor-conditioned critic that improves the archive across the entire descriptor space, (2) we exploit the actor-critic training to learn a descriptor-conditioned policy at no additional cost, distilling the knowledge of the archive into one single versatile policy that can execute the entire range of behaviors contained in the archive. Our algorithm, DCG-MAP-Elites improves the QD score over PGA-MAP-Elites by 82% on average, on a set of challenging locomotion tasks.

CCS CONCEPTS

• **Computing methodologies** → **Evolutionary robotics**.

KEYWORDS

Quality-Diversity, Reinforcement Learning, Neuroevolution, MAP-Elites, Policy Gradient

ACM Reference Format:

Maxence Faldor, Félix Chalumeau, Manon Flageat, and Antoine Cully. 2023. MAP-Elites with Descriptor-Conditioned Gradients and Archive Distillation into a Single Policy. In *Genetic and Evolutionary Computation Conference (GECCO '23), July 15–19, 2023, Lisbon, Portugal.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3583131.3590503

This work is licensed under a Creative Commons Attribution International 4.0 License.
GECCO '23, July 15–19, 2023, Lisbon, Portugal
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0119-1/23/07.
https://doi.org/10.1145/3583131.3590503

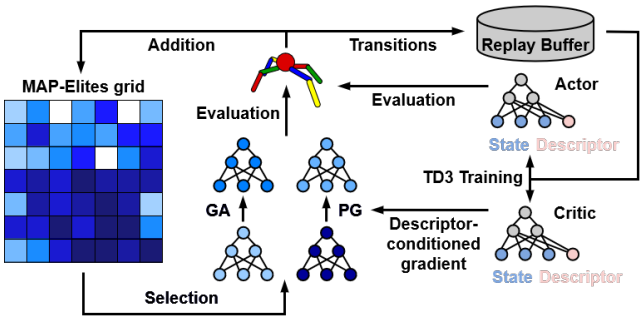


Figure 1: DCG-MAP-Elites performs a standard MAP-Elites loop of selection, variation, evaluation and addition. Two complementary variation operators are applied: (1) a standard Genetic Algorithm (GA) variation operator for exploration, (2) a Descriptor-Conditioned Policy Gradient (PG) variation operator for fitness improvement. Concurrently to the critic's training, the knowledge of the archive is distilled in the descriptor-conditioned actor as by-product.

1 INTRODUCTION

A fascinating aspect of evolution is its ability to generate a large variety of different species, each being adapted to their ecological niche. Inspired by this idea, Quality-Diversity (QD) optimization is a family of evolutionary algorithms that aims to generate a set of both diverse and high-performing solutions to a problem [3, 7, 32]. Contrary to traditional optimization methods that return a single high-performing solution, the goal of QD algorithms is to illuminate a search space of interest called *descriptor space* [27]. Producing a large collection of diverse and effective solutions provides multiple alternatives to solve a single problem which is useful in robotics to improve robustness, recover from damage [6] or reduce the reality gap [4]. Furthermore, conventional optimization methods are prone to get stuck in local optima whereas keeping diverse solutions to a problem can help to find stepping stones that lead to globally better solutions [27, 28]. Another benefit of diversity search is efficient exploration in problems where the reward signal is sparse or deceptive [2, 8, 31].

MAP-Elites [27] is a conceptually simple but effective QD optimization algorithm that has shown competitive results in a variety

of applications, to generate large collections of diverse skills. However, MAP-Elites relies on random variations that can cause slow convergence in large search space [5, 28, 31], making it inadequate to evolve neural networks with a large number of parameters.

In contrast, Deep Reinforcement Learning (DRL) [25, 26] algorithms combine reinforcement learning with the directed search power of gradient-based methods in order to learn a single solution. DRL can surpass human performance at video games [40], beat world champions in board games [35] and control complex robots in continuous action spaces [17], which is a long-standing challenge in artificial intelligence. Policy Gradient methods have shown state-of-the-art results to learn large neural network policies with thousands of parameters in high-dimensional state space and continuous action space [18, 24, 36].

PGA-MAP-Elites [28] is an extension of MAP-Elites that integrates the sample efficiency of DRL using the TD3 algorithm [15]. This algorithm uses a Policy Gradient (PG) variation operator for efficient fitness improvement, coupled with the usual Genetic Algorithm (GA) variation operator. The PG variation operator leverages gradients derived from DRL to improve fitness and drive mutations towards the global optimum and is supported by the divergent search of the GA variation operator for both exploration and optimization [10]. Other recent works have also introduced methods to combine the strength of QD algorithms with reinforcement learning [31, 38] on complex robotics tasks.

PGA-MAP-Elites achieves state-of-the-art performances in most of the environments considered so far in the literature [28, 31, 38]. However, the PG variation operator becomes ineffective in tasks where the global optimum is in an area of the search space that is not likely to produce offspring that are added to the archive. For example, consider a locomotion task where the fitness is the opposite of the energy consumption and the descriptor is defined as the final position of the robot. The global optimum for the fitness is the solution that does not move in order to minimize energy consumption. Thus, the PG variation operator will encourage solutions to stay motionless, collapsing their descriptors to a single point, the descriptor of the global optimum. Consequently, the PG variation operator generates offspring that are discarded and no interesting stepping stone is found, thereby hindering diversity.

In this work, we introduce Descriptor-Conditioned Gradients MAP-Elites (DCG-MAP-Elites) that builds upon PGA-MAP-Elites algorithm by enhancing the PG variation operator with a descriptor-conditioned critic that provides gradients depending on a target descriptor. The descriptor-conditioned critic takes as input a state and a descriptor to evaluate actions. With such a descriptor-conditioned critic, the PG variation operator can mutate solutions to produce offspring with higher fitness while targeting a desired descriptor, thereby avoiding the collapse of the descriptor onto a single point.

Furthermore, TD3 which is the DRL algorithm used by the PG variation operator, requires training an actor and a critic in parallel. We take advantage of this intertwined actor-critic training to make the actor descriptor-conditioned as well, allowing it to take actions based on the current state and on an input descriptor we want to achieve. Thus, instead of taking actions that maximize the fitness globally, the actor now takes actions that maximize the fitness while achieving a desired descriptor. At the end of training, we can condition the actor on a desired descriptor to execute a policy that takes actions that achieve the desired descriptor. On half of the tasks, we observe that the descriptor-conditioned actor can achieve the entire range of descriptors contained in the archive with a similar QD-score, negating the burden of dealing with a collection of thousands of solutions.

In summary, we present two contributions: (1) we enhance the PG variation operator with a descriptor-conditioned critic, (2) we distill the knowledge of the archive into one single versatile policy at no additional cost. We compare our algorithm to four state-of-the-art QD algorithms on four high-dimensional robotics locomotion tasks. The results demonstrate that DCG-MAP-Elites has a QD-score 82% higher than PGA-MAP-Elites on average.

2 BACKGROUND

2.1 Problem Statement

We consider an agent sequentially interacting with an environment at discrete time steps t for an episode of length T. At each time step t, the agent observes a state s_t, takes an action a_t and receives a scalar reward r_t. We model it as a Markov Decision Process (MDP) which comprises a *state space* $\mathcal{S}$, a continuous *action space* $\mathcal{A}$, a stationary *transition dynamics distribution* $p(s_{t+1} \mid s_t, a_t)$ and a *reward function* $r\colon \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. In this work, a *policy* (also called *solution*) is a deterministic neural network parameterized by $\phi \in \Phi$, and denoted $\pi_\phi : \mathcal{S} \to \mathcal{A}$. The agent uses its policy to select actions and interact with the environment to give a trajectory of states, actions and rewards. The *fitness* of a solution is given by $F\colon \Phi \to \mathbb{R}$, defined as the expected discounted return $\mathbb{E}_{\pi_\phi}\left[\sum_{t=0}^{T-1} \gamma^t r_t\right]$.

The objective of QD algorithms in this MDP setting is to find the highest-fitness solutions in each point of the *descriptor space* $\mathcal{D}$. The descriptor function $D\colon \Phi \to \mathcal{D}$ is generally defined by the user and characterizes solutions in a meaningful way for the type of diversity desired. With this notation, our objective is to evolve a population of solutions that are both high-performing with respect to F and diverse with respect to D.

2.2 MAP-Elites

Multi-dimensional Archive of Phenotypic Elites (MAP-Elites) [27] is a simple yet effective QD algorithm that discretizes the descriptor space $\mathcal{D}$ into a multi-dimensional grid of cells called archive $\mathcal{X}$ and searches for the best solution in each cell, see Alg. 4. The goal of the algorithm is to return an archive that is filled as much as possible with high-fitness solutions. MAP-Elites starts by generating random solutions and adding them to the archive. The algorithm then repeats the following steps until a budget of I solutions have been evaluated: (1) a batch of solutions from the archive are uniformly selected and modified through mutations and/or crossovers to produce offspring, (2) the fitnesses and descriptors of the offspring are evaluated, and each offspring is placed in its corresponding cell if and only if the cell is empty or if the offspring has a better fitness than the current solution in that cell, in which case the current solution is replaced. As most evolutionary methods, MAP-Elites relies on undirected updates that are agnostic to the fitness objective. With a Genetic Algorithm (GA) variation operator, MAP-Elites performs a divergent search that may cause slow convergence in high-dimensional problems due to a lack of directed search power, and thus, is performing best on low-dimensional search space [28].

2.3 Deep Reinforcement Learning

Deep Reinforcement Learning (DRL) [25, 26] combines the reinforcement learning framework with the function approximation capabilities of deep neural networks to represent policies and value functions in high-dimensional state and action spaces. In contrast to black-box optimization methods like evolutionary algorithms, DRL leverages the structure of the MDP in the form of the Bellman equation to achieve better sample efficiency. The objective is to find an optimal policy π_ϕ, which maximizes the expected return or fitness $F(\pi_\phi)$. In reinforcement learning, many approaches try to estimate the action-value function $Q^\pi(s,a) = \mathbb{E}_\pi\left[\sum_{t=0}^{T-1}\gamma^t r_t|s,a\right]$ defined as the expected discounted return starting from state s, taking action a and thereafter following policy π.

The Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithm [15] is an actor-critic, off-policy reinforcement learning method that achieves state-of-the-art results in environments with large and continuous action space. TD3 indirectly learns a policy π_ϕ via maximization of the action-value function $Q_\theta(s,a)$. The approach is closely connected to Q-learning [15] and tries to approximate the optimal action-value function $Q^*(s,a)$ in order to find the optimal action $a^*(s) = \arg\max_a Q^*(s,a)$. However, computing the maximum over action in $\max_a Q_\theta(s,a)$ is intractable in continuous action space, so it is approximated using $\max_a Q_\theta(s,a) = Q_\theta(s,\pi_\phi(s))$. In TD3, the policy π_ϕ takes actions in the environment and the transitions are stored in a replay buffer. The collected experience is then used to train a pair of critics Q_{θ_1}, Q_{θ_2} using temporal difference and target networks $Q_{\theta_1'}$, $Q_{\theta_2'}$ are updated to slowly track the main networks. Both critics use a single regression target y, calculated using whichever of the two critics gives a smaller target value and using target policy smoothing by sampling a noise $\epsilon \sim \text{clip}(\mathcal{N}(0,\sigma), -c, c)$:

$$y = r(s_t,a_t) + \gamma \min_{i=1,2} Q_{\theta_i'}\left(s_{t+1}, \pi_{\phi'}(s_{t+1}) + \epsilon\right) \tag{1}$$

Both critics are learned by regression to this target and the policy is learned with a delay, only updated every Δ iterations simply by maximizing Q_{θ_1} with $\max_\phi \mathbb{E}\left[Q_{\theta_1}(s,\pi_\phi(s))\right]$. The actor is updated using the deterministic policy gradient:

$$\nabla_\phi J(\phi) = \mathbb{E}\left[\nabla_a Q_{\theta_1}(s,a)|_{a=\pi_\phi(s)} \nabla_\phi \pi_\phi(s)\right] \tag{2}$$

2.4 PGA-MAP-Elites

Policy Gradient Assisted MAP-Elites (PGA-MAP-Elites) [28] is an extension of MAP-Elites that is designed to evolve deep neural networks by combining the directed search power and sample efficiency of DRL methods with the exploration capabilities of genetic algorithms, see Alg. 5. The algorithm follows the usual MAP-Elites loop of selection, variation, evaluation and addition for a budget of I iterations, but uses two parallel variation operators: half of the offspring are generated using a standard Genetic Algorithm (GA) variation operator and half of the offspring are generated using a Policy Gradient (PG) variation operator. During each iteration of the loop, PGA-MAP-Elites stores the transitions from offspring evaluation in a replay buffer $\mathcal{B}$ and uses it to train a pair of critics based on the TD3 algorithm, described in Alg. 6. The trained critic is then used in the PG variation operator to update the selected solutions from the archive for m gradient steps to select actions that maximize the approximated action-value function, as described in Alg. 7. At each iteration, the critics are trained for n steps of gradients descents towards the target described in Eq. 1 averaged over N transitions of experience sampled uniformly from the replay buffer $\mathcal{B}$. The actor (also named greedy actor [28]) learns with a delay Δ via maximization of the critic according to Eq. 2.

3 RELATED WORK

3.1 Scaling QD to Neuroevolution

The challenge of evolving diverse solutions in a high-dimensional search space has been an active research subject over the recent years. ME-ES [5] scales MAP-Elites to high-dimensional solutions parameterized by large neural networks. This algorithm leverages Evolution Strategies to perform a directed search that is more efficient than random mutations used in Genetic Algorithms. Fitness gradients are estimated locally from many perturbed versions of the parent solution to generate a new one. The population tends towards regions of the parameter space with higher fitness but it requires sampling and evaluating a large number of solutions, making it particularly data inefficient. In order to use the time step level information and hence improve data efficiency, methods that combine MAP-Elites with Reinforcement Learning [28, 30, 31, 38] have emerged and proved to efficiently evolve populations of high-performing and diverse neural network for complex tasks. PGA-MAP-Elites [28] uses policy gradients for part of its mutations, see section 2.4 for details. CMA-MEGA [38] estimates descriptor gradients with Evolution Strategies and combines the fitness gradient and the descriptor gradients with a CMA-ES mechanism [12, 19]. QD-PG [31] introduces a diversity reward based on the novelty of the states visited and derives a policy gradient for the maximization of those diversity rewards which helps exploration in settings where the reward is sparse or deceptive. PBT-MAP-Elites [30] mixes MAP-Elites with a population based training process [21] to optimize hyper-parameters of diverse RL agents. Interestingly, recent work [37] scales the algorithm CMA-MAE [13] to high-dimensional policies on robotics tasks with pure Evolution Strategies while showing comparable data efficiency to QD-RL approaches. It shows competitiveness but is still outperformed by PGA-MAP-Elites.

3.2 Conditioning the critic

None of the methods above take descriptors into account to compute policy gradients used to mutate solutions. In other words, they do not use descriptor-conditioned actor or critic as our method DCG-MAP-Elites does. The concept of descriptor-conditioned critic is related to the concept of Universal Value Function Approximators [33] and the most related field to Quality-Diversity that uses it is Skill Discovery Reinforcement Learning [1]. In VIC, DIAYN, DADS, SMERL [9, 16, 23, 34], conditioned actor-critics are used but the condition is a sampled prior and does not correspond to a real posterior like in DCG-MAP-Elites. Furthermore, those methods use diversity at the step level and not explicitly at the trajectory level like ours. Finally, they do not use an archive to store their population, resulting in much smaller sets of final policies. Ultimately, it has been shown that behaviors evolved by QD methods are competitive with skills learned by this family of methods [1], in regards to their use for adaptation and hierarchical learning.

3.3 Archive distillation

Distilling the knowledge of an archive into a single neural model is an alluring process that reduces the number of parameters outputted by the algorithm and enables generalization and interpolation/extrapolation. Although distillation is usually referring to policy distillation — learning the observation/action mapping from a teacher policy — we present archive distillation as a general term referring to any kind of knowledge transfer from an archive to another model, should it be the policies, transitions experienced in the environment, full trajectories or discovered descriptors.

To the best of our knowledge, two QD-related works use the concept of archive distillation. Go-Explore [8] maintains an archive of reached states and policies and then robustifies a global policy via imitation learning. Another interesting approach to archive distillation is to learn a generative policy network [22] over the policy contained in the archive. Our approach DCG-MAP-Elites distills the experience of the archive into a single versatile policy.

4 DCG-MAP-ELITES

Our new method Descriptor-Conditioned Gradients MAP-Elites (DCG-MAP-Elites) overcomes limitations of PGA-MAP-Elites by leveraging a descriptor-conditioned critic to improve the PG variation operator and concurrently distills the knowledge of the archive in a single versatile policy as a by-product of the actor-critic training. The pseudocode is provided in Alg. 1. The algorithm follows the usual MAP-Elites loop of selection, variation, evaluation and addition for a budget of I iterations. Two complementary and independent variation operators are used in parallel: 1) a standard GA operator 2) a descriptor-conditioned PG operator. At each iteration, the transitions from the evaluation step are stored in a replay buffer and used to train an actor-critic pair based on TD3.

Contrary to PGA-MAP-Elites, the actor-critic pair is descriptor-conditioned. In addition to the state s and action a, the critic $Q_\theta(s, a \mid d)$ also depends on the descriptor d and estimates the expected discounted return starting from state s, taking action a and thereafter following policy π *and* achieving descriptor d. Achieving descriptor d means that the descriptor of the trajectory generated by the policy π should have descriptor d. In addition to the state s, the actor $\pi_\phi(s \mid d)$ also depends on the descriptor d and maximizes the expected discounted return conditioned on achieving descriptor d. Thus, the goal of the descriptor-conditioned actor is to achieve the input descriptor d *while* maximizing fitness.

4.1 Descriptor-Conditioned Critic

Instead of estimating the action-value function with $Q_\theta(s, a)$, we want to estimate the descriptor-conditioned action-value function with $Q_\theta(s, a \mid d)$. When a policy π interacts with the environment for an episode of length T, it generates a trajectory τ, which is a sequence of transitions:

$$(s_0, a_0, r_0, s_1), \ldots, (s_{T-1}, a_{T-1}, r_{T-1}, s_T)$$

with descriptor $D(\pi) = d$. We extend the definition of a transition (s, a, r, s') to include the descriptor d of the policy (s, a, r, s', d). Thus, a trajectory τ with descriptor d gives a sequence of transitions:

$$(s_0, a_0, r_0, s_1, d), \ldots, (s_{T-1}, a_{T-1}, r_{T-1}, s_T, d)$$

Algorithm 1 DCG-MAP-Elites

Input: batch size b, number of GA variations $g \leq b$
  Initialize archive $\mathcal{X}$ with b random solutions and replay buffer $\mathcal{B}$
  Initialize critic networks $Q_{\theta_1}, Q_{\theta_2}$ and actor network π_ϕ
  $i \leftarrow 0$
  while $i < I$ **do**
    TRAIN_ACTOR_CRITIC$(Q_{\theta_1}, Q_{\theta_2}, \pi_\phi, \mathcal{B})$
    $\pi_{\psi_1}, \ldots, \pi_{\psi_b} \leftarrow$ SELECTION$(\mathcal{X})$
    $\pi_{\widehat{\psi}_1}, \ldots, \pi_{\widehat{\psi}_g} \leftarrow$ VARIATION_GA$(\pi_{\psi_1}, \ldots, \pi_{\psi_g})$
    $\pi_{\widehat{\psi}_{g+1}}, \ldots, \pi_{\widehat{\psi}_b} \leftarrow$ VARIATION_PG$(\pi_{\psi_{g+1}}, \ldots, \pi_{\psi_b}, Q_{\theta_1}, \mathcal{B})$
    ADDITION$(\pi_{\widehat{\psi}_1}, \ldots, \pi_{\widehat{\psi}_b}, \mathcal{X}, \mathcal{B})$
    $i \leftarrow i + b$
  function ADDITION$(\mathcal{X}, \mathcal{B}, \pi_\phi, \pi_{\widehat{\psi}} \ldots)$
    for $d' \in \mathcal{D}$ sampled from b solutions in $\mathcal{X}$ **do**
      $(f, \text{transitions}) \leftarrow F(\pi_\phi(\,.\mid d'))$
      INSERT$(\mathcal{B}, \text{transitions})$
    for $\pi_{\widehat{\psi}} \ldots$ **do**
      $(f, \text{transitions}) \leftarrow F(\pi_{\widehat{\psi}}), d \leftarrow D(\pi_{\widehat{\psi}})$
      INSERT$(\mathcal{B}, \text{transitions})$
      if $\mathcal{X}(d) = \emptyset$ or $F(\mathcal{X}(d)) < f$ **then**
        $\mathcal{X}(d) \leftarrow \pi_{\widehat{\psi}}$

However, the descriptor is only available at the end of the episode, therefore the transitions can only be augmented with the descriptor after the episode is done. In all the tasks we consider, the reward function is positive $r\colon \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}^+$ and hence, the fitness function F and action-value function are positive as well. Thus, for any sampled descriptor $d' \in \mathcal{D}$, we define the descriptor-conditioned critic as equal to the normal action-value function when the policy achieves the sampled descriptor d' and as equal to zero when the policy does not achieve the sampled descriptor d'. Given a transition (s, a, r, s', d), and $d' \in \mathcal{D}$,

$$Q_\theta(s, a \mid d') = \begin{cases} Q_\theta(s, a), & \text{if } d = d' \\ 0, & \text{if } d \neq d' \end{cases} \tag{3}$$

However, with this piecewise definition, the descriptor-conditioned action-value function is not continuous and violates the universal approximation theorem continuity hypothesis [20]. To address this issue, we introduce a similarity function $S\colon \mathcal{D}^2 \rightarrow]0, 1]$ defined as $S(d, d') = e^{-\frac{\|d-d'\|_{\mathcal{D}}}{l}}$ to smooth the descriptor-conditioned critic and relax Eq. 3 into:

$$\begin{aligned} Q_\theta(s, a \mid d') = S(d, d')\, Q_\theta(s, a) &= S(d, d')\, \mathbb{E}_\pi \left[\sum_{t=0}^{T-1} \gamma^t r_t \middle| s, a \right] \\ &= \mathbb{E}_\pi \left[\sum_{t=0}^{T-1} \gamma^t S(d, d') r_t \middle| s, a \right] \end{aligned} \tag{4}$$

With Eq. 4, we demonstrate that learning the descriptor-conditioned critic is equivalent to scaling the reward by the similarity $S(d, d')$ between the descriptor of the trajectory d and the sampled descriptor d'. Therefore, the critic target in Eq. 1 is modified to include the similarity scaling and the descriptor-conditioned actor:

$$y = S(d, d')\, r(s_t, a_t) + \gamma \min_{i=1,2} Q_{\theta_i'} \left(s_{t+1}, \pi_{\phi'}(s_{t+1} \mid d') + \epsilon \mid d' \right) \tag{5}$$

If the sampled descriptor d' is approximately equal to the observed descriptor d of the trajectory $d \approx d'$ then we have $S(d, d') \approx 1$ so the

reward is unchanged. However, if the descriptor d' is very different from the observed descriptor d then, the reward is scaled down to $S(d, d')\, r(s_t, a_t) \approx 0$. The scaling ensures that the magnitude of the reward depends not only on the quality of the action a with regards to the fitness function F, but also on achieving the descriptor d'. Given one transition (s, a, r, s', d), we can generate infinitely many critic updates by sampling $d' \in \mathcal{D}$. This is leveraged in the new actor-critic training introduced with DCG-MAP-Elites, which is detailed in Alg. 2 and section 4.3.

Algorithm 2 Descriptor-conditioned Actor-Critic Training

function TRAIN_ACTOR_CRITIC($Q_{\theta_1}, Q_{\theta_2}, \pi_\phi, \mathcal{B}$)
  for $t = 1 \rightarrow n$ **do**
    Sample N transitions $(s, a, r(s,a), s', d, d')$ from $\mathcal{B}$
    Sample smoothing noise ϵ
    $y \leftarrow S(d, d')\, r(s,a) + \gamma \min_{i=1,2} Q_{\theta'_i}\left(s', \pi_{\phi'}(s' \mid d') + \epsilon \mid d'\right)$
    Update both critics by regression to y
    if $t \bmod \Delta$ **then**
      Update actor using the deterministic policy gradient:
      $\frac{1}{N} \sum \nabla_a Q_{\theta_1}(s, a \mid d')|_{a=\pi_\phi(s|d')} \nabla_\phi \pi_\phi(s \mid d')$
      Soft-update target networks $Q_{\theta i'}$ and $\pi_{\phi'}$

4.2 Descriptor-Conditioned Actor and Archive Distillation

The learning of the critic requires training an actor π_ϕ to approximate the optimal action a^* as explained in section 2.3. However, in this work, the action-value function estimated by the critic is conditioned on a descriptor d. Hence, we don't want to estimate the best action globally, but the best action given that the policy should achieve descriptor d by the end of the trajectory. Therefore, the actor is extended to a descriptor-conditioned policy $\pi_\phi(s \mid d)$, that maximizes the descriptor-conditioned critic's value with $\max_\phi \mathbb{E}\left[Q_\theta(s, \pi_\phi(s \mid d) \mid d)\right]$. The actor is updated using the deterministic policy gradient, see Alg. 2:

$$\nabla_\phi J(\phi) = \frac{1}{N} \sum \nabla_a Q_{\theta_1}(s, a \mid d')|_{a=\pi_\phi(s|d')} \nabla_\phi \pi_\phi(s \mid d') \quad (6)$$

The policy $\pi_\phi(s \mid d)$ learns to suggest actions a that optimize reward $r(s,a)$ *while* generating a trajectory achieving descriptor d. Consequently, the descriptor-conditioned actor can exhibit a wide range of descriptors, effectively distilling some of the capabilities of the archive into a single versatile policy.

4.3 Actor-Critic Training

In section 4.1, we show that the new descriptor-conditioned critic target in Eq. 5 requires a sampled descriptor d'. At each iteration, the transitions $(s, a, r(s,a), s', d)$ generated from the evaluation of offspring are stored in a replay buffer $\mathcal{B}$. To learn the relation between being in state s, taking action a and achieving descriptor d', the critic needs to be trained on "positive" samples i.e. samples with $d' = d$ and "negative" samples i.e. samples with $d' \neq d$. Positive samples are easy to generate because we can sample d' as equal to the observed descriptor d of the trajectory. However, negative samples require to define an arbitrary sampling strategy that can impact the performance of the algorithm.

Moreover, learning to act from data without active interactions in the environment is difficult [29]. If the actor only learns passively on rollouts performed by offspring from policies stored in the archive, strong deviation from the state-action distribution can happen, preventing convergence. To solve this passive learning problem and generate negative samples at the same time, we evaluate the actor π_ϕ in the environment. At each iteration, we sample a batch of b descriptors d' in $\mathcal{D}$ and evaluate $\pi_\phi(\,.\mid d')$. The actor evaluation step gives b trajectories with transitions $(s_t, a_t, r_t, s_{t+1}, d, d')$ with d the observed descriptor achieved by the trajectory and d' the input descriptor that is sampled for evaluation. In general, the observed descriptor and the sampled descriptor are different, $d' \neq d$, providing negative samples to the critic training. To get the descriptors d', we uniformly sample with replacement b solutions π_ψ from the archive and take their descriptors with an additional Gaussian noise. The solutions have descriptors $D(\pi_\psi) = d_\psi$ and we compute a batch of descriptors $d' = d_\psi + \mathcal{N}(0_{\mathcal{D}}, \sigma_d I)$. The negative samples from the actor evaluation are stored in the replay buffer alongside the positive samples from offspring evaluation for which we take $d' = d$, as detailed in the ADDITION function of Alg. 1.

4.4 Descriptor-Conditioned PG variation

Once the critic $Q_\theta(s, a \mid d)$ is learned, it can be used to improve the fitness of solutions in the archive, as described in Alg. 3. A parent solution π_ψ with descriptor $D(\pi_\psi) = d_\psi$ is selected from the archive. Then, we apply the PG variation operator, using the descriptor d_ψ to condition the critic, and thus, to apply m descriptor-conditioned gradient steps using the deterministic policy gradient:

$$\nabla_\psi J(\psi) = \frac{1}{N} \sum \nabla_a Q_{\theta_1}(s, a \mid d_\psi)|_{a=\pi_\psi(s)} \nabla_\psi \pi_\psi(s) \quad (7)$$

The goal is to improve the fitness of the solution while remaining in the same cell as the parent.

Algorithm 3 Descriptor-conditioned PG Variation

function VARIATION_PG($\pi_\psi \ldots, Q_{\theta_1}, \mathcal{B}$)
  for $\pi_\psi \ldots$ **do**
    $d_\psi \leftarrow D(\pi_\psi)$
    for $i = 1 \rightarrow m$ **do**
      Sample N transitions $(s, a, r(s,a), s', d, d')$ from $\mathcal{B}$
      Update actor using the deterministic policy gradient:
      $\frac{1}{N} \sum \nabla_a Q_{\theta_1}(s, a \mid d_\psi)|_{a=\pi_\psi(s)} \nabla_\psi \pi_\psi(s)$
  return $\pi_{\hat{\phi}} \ldots$

5 EXPERIMENTS

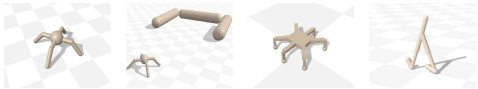
5.1 Evaluation Tasks

We evaluate DCG-MAP-Elites on four QDGym tasks [28] implemented in Brax [14] and derived from standard DRL benchmarks for robot locomotion. We call these tasks "Ant-Omni", "AntTrap-Omni", "Hexapod-Omni", and "Walker-Uni". Three omnidirectional tasks and one unidirectional task are considered for evaluation. Ant-Omni, AntTrap-Omni and Hexapod-Omni are omnidirectional tasks in which the objective is for the robot to reach all possible points in the plane while minimizing energy consumption. Walker-Uni is a unidirectional task in which the objective is for the robot

to discover all possible ways to walk forward while maximizing a trade-off between speed and energy consumption. Walker-Uni is exactly the same tasks used in PGA-MAP-Elites paper [28], Ant-Omni and Hexapod-Omni have been used by Flageat and Lim et al. [11] and AntTrap-Omni is adapted from QD-PG paper [31]. The difference is the elimination of the forward reward term in the reward function. The goal of AntTrap-Omni is to reach every point in the plane while minimizing energy consumption in a deceptive environment. This new task is designed to show that our algorithm performs well in environment where there is a discontinuity on the fitness landscape in descriptor space caused by the trap. The trap causes a discontinuity that the descriptor-conditioned critic has to learn. Indeed, points on both sides of the trap will be close in the descriptor space, but distant in terms of trajectory to achieve these descriptors. We detail these tasks in Tab. 1.

PGA-MAP-Elites has previously shown state-of-the-art results on some of these tasks, in particular Walker-Uni but tends to struggle in omnidirectional tasks. In those tasks, the global optimal of the fitness function is a solution that prevents the robot to move, which is directly opposed to the objective of discovering how to reach different locations. Hence, most of the offspring generated by the PG variation will tend to move less and travel a shorter distance. Instead, DCG-MAP-Elites aims to improve the energy consumption while maintaining the ability to reach distant locations.

Table 1: Evaluation Tasks

	Ant	AntTrap	Hexapod	Walker
				
State	Position and velocity of body and joints			
Action	Torques for each joint			
Type	Omni	Omni	Omni	Uni
Descriptor	Final (x, y) position			Feet contact
Fitness	Surviving reward + Energy consumption penalty			
State dim	29	29	48	19
Action dim	8	8	18	6
Episode len	250	250	250	1000
Parameters	21,384	21,384	25,106	19,846

5.2 Baselines

We compare DCG-MAP-Elites with four algorithms: CVT-MAP-Elites [39], MAP-Elites-ES [5], PGA-MAP-Elites [28] and QD-PG [31]. We could not compare to PBT-MAP-Elites [30] as its implementation is not yet available. We use QDax implementations and each experiment is replicated 20 times with random seeds, over one million evaluations. The source code of DCG-MAP-Elites is available at github.com/adaptive-intelligent-robotics/DCG-MAP-Elites including a container in which all the experiments can be replicated.

5.3 Evaluation Metrics

We consider three main metrics to evaluate the archives and two additional metrics to evaluate the descriptor-conditioned actor. We compute p-values based on the Wilcoxon rank-sum test with a Bonferroni correction for all metrics.

- **QD-score:** The sum of fitness of all solutions stored in the archive. In the task considered, fitnesses are always positive, which avoid penalizing algorithms for finding additional solutions. This score encompasses both the quality and diversity of the population.
- **Coverage:** The proportion of filled cells in the archive, indicating descriptor space illumination.
- **Maximum fitness:** The fitness of the best solution in the archive.

We consider two additional metrics to evaluate the descriptor-conditioned policy. The purpose of these metrics is to quantify how similar the archive and the descriptor-conditioned policy are in terms of QD-score and coverage of the descriptor space.

- **Descriptor-conditioned QD-score:** The sum of fitness achieved by the descriptor-conditioned policy evaluated on each descriptor of the solutions in the archive.

$$\sum_{\pi_\psi \in \mathcal{X}} F(\pi_\phi(\,.\mid D(\pi_\psi)))$$

- **Descriptor error mean:** (i) for the archive, the average difference over all solutions, between the stored descriptor and the reevaluated descriptor, (ii) for the descriptor-conditioned policy, the average difference over all solutions, between the stored descriptor and the reevaluated descriptor achieved by the policy conditioned on the stored descriptor. ($|\mathcal{X}|$ number of solutions in the archive).

$$\frac{1}{|\mathcal{X}|} \sum_{\pi_\psi \in \mathcal{X}} ||D(\pi_\phi(\,.\mid D(\pi_\psi))) - D(\pi_\psi)||_{\mathcal{D}}$$

5.4 Results

5.4.1 Archive. The experimental results on Fig. 2 demonstrate that DCG-MAP-Elites achieves the best archive in the two Ant tasks in terms of QD-score ($p < 10^{-6}$) and coverage ($p < 10^{-4}$) compared to all other baselines. On the two other tasks, the QD-score of our algorithm is similar to PGA-MAP-Elites, the previous state-of-the-art, while achieving a significantly better coverage than all baselines ($p < 10^{-3}$). The coverage metric shows that DCG-MAP-Elites surpasses the exploration capabilities of QD-PG on all tasks ($p < 10^{-4}$). We also show that our method still benefits from the exploration power of the GA operator even in deceptive environment like AntTrap-Omni.

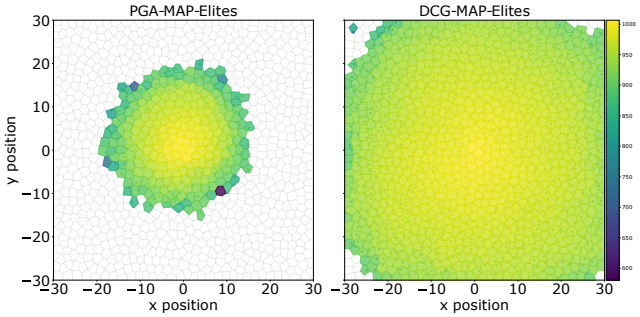


Figure 3: Final archives for PGA-MAP-Elites and DCG-MAP-Elites on Ant-Omni after one million evaluations.

The experimental results confirm that DCG-MAP-Elites is able to overcome the limits of PGA-MAP-Elites on omnidirectional tasks

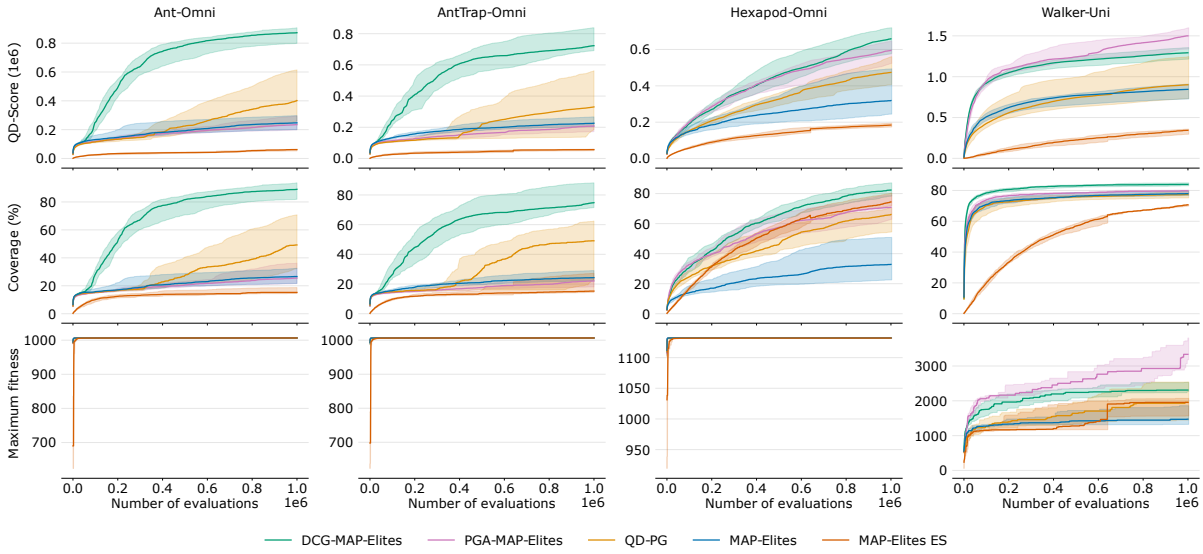


Figure 2: QD-score, coverage and maximum fitness for DCG-MAP-Elites and the baselines on all tasks. Each experiment is replicated 20 times with different random seeds. The solid line is the median and the shaded area is the first and third quartiles.

while still performing on par when evaluated on the unidirectional task (Walker-Uni) where no significant improvement was expected. Thus confirming the interest of having a descriptor-conditioned gradient to make the PG variation operator fruitful in a wider range of tasks. Overall, DCG-MAP-Elites shows best (or competitive) performance on all metrics and tasks, hence proving to be the first successful effort in the QD-RL literature to achieve well on both the unidirectional and omnidirectional tasks. Previous efforts were usually adapted to either one or the other [28, 31, 38]. Nevertheless, our results show that there is still room for improvement. In particular, we can see that DCG-MAP-Elites was not able to outperform the baselines with the same margin on Hexapod-Omni that it has been able to do on the other omnidirectional tasks. We hypothesize that the environment dynamics and the higher-dimensional search space of Hexapod-Omni makes it much more challenging.

Table 2: Median of the QD-score and Descriptor Error Mean (DEM) after one million evaluations for the archive and for the descriptor-conditioned policy over 20 replications.

Metric	Type	Ant	AntTrap	Hexapod	Walker
QD-score (10^5)	Archive	8.7	7.2	6.5	9.8
	Policy	8.3	7.0	0.96	5.2
DEM	Archive	4.04	3.44	0.078	0.12
	Policy	6.25	5.8	1.39	0.27

5.4.2 Descriptor-Conditioned Policy. In Tab. 2 we provide the final QD-score and Descriptor Error Mean (DEM) of the archive and the descriptor-conditioned policy, see section 5.3 for definitions. First, on Ant-Omni and AntTrap-Omni, the descriptor-conditioned policy achieves the same QD-score as the archive, measured by the descriptor-conditioned QD-score (decreases by less than 3%). This shows that the single-conditioned policy is able to restore the quality of the archive although having compressed the information in a single network. Second, it is interesting to cross this observation with the fact that, at the same time, the Descriptor Error Mean obtained by the conditioned policy is reasonable. We observe a DEM of 6.25, which is less than 8% of the typical size of the environment (i.e. the diagonal of the square considered). The reader can refer to Fig. 3 to get an idea of what that represents. In addition, the DEM of the archive is of 4.04 (4.7% of size of the environment), showing an inevitable stochasticity in the evaluation that is related to the simulator used [14]. Those two combined observations (QD-score and DEM) tend to show that our archive and our descriptor-conditioned policy have similar properties on Ant-Omni and AntTrap-Omni and hence that the versatile policy could be a practical summary of our archive. Third, we can observe that on the Hexapod-Omni and Walker-Uni tasks, the descriptor-conditioned policy struggles to recover the QD-score of the archive (reduced by more than 40% in both cases). On the Hexapod-Omni, the DEM of the descriptor-conditioned policy is significantly larger than the typical error obtained by the archive (1.39 vs 0.078). This task is more challenging than Ant-Omni because of a higher number of dimensions (see Tab. 1) and we hypothesize that the optimization algorithm used (TD3) could be the limitation. For Walker-Uni, the DEM of the archive (0.12) shows that there is also an inherent variability of the descriptor, we are hence less surprised by the DEM obtained by the single-conditioned policy (0.27) although it could make it less usable in practice.

Overall, the results show that our single descriptor-conditioned policy can already be seen as a promising summary of our archive, showing very similar properties on half our tasks. The other tasks prove that there is still room for improvement of our method.

5.4.3 Ablations. We perform additional ablation experiments to show the importance of three components of DCG-MAP-Elites, namely: using negative samples for the critic's training, actively collecting experiences with the actor, and conditioning the actor on the descriptor. In **Ablation (1)**, we remove the actor evaluation

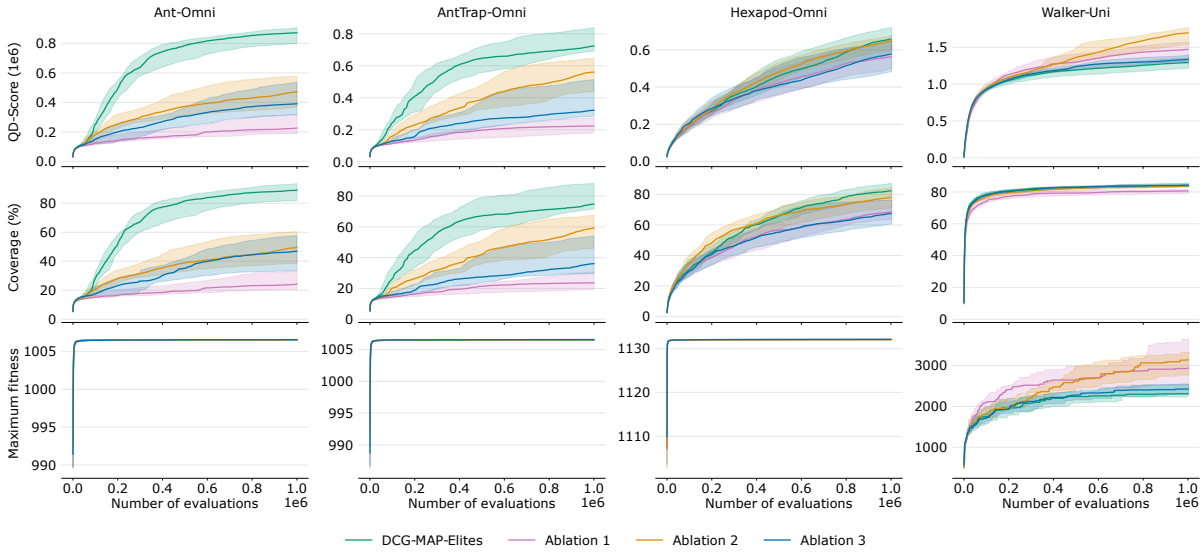


Figure 4: QD-score, coverage and maximum fitness for DCG-MAP-Elites and the ablations on all tasks. Each experiment is replicated 20 times with different random seeds. The solid line is the median and the shaded area is the first and third quartiles. Ablation (1) is DCG-MAP-Elites without actor evaluation and without negative samples, Ablation (2) is DCG-MAP-Elites without actor evaluation but with negative samples, Ablation (3) is DCG-MAP-Elites without a descriptor-conditioned actor.

in the environment. This allows to study the impact of both passive learning and of lack of negative samples, see section 4.3 for a detailed explanation. The lack of active learning deprives the learning of negative samples that are necessary to learn the descriptor-conditioned action-value function. We observe that ablation (1) performs worse than DCG-MAP-Elites on all omnidirectional tasks ($p < 0.01$). However it performs similarly on Walker-Uni, which can be explained by the fact that non-conditioned PG emitter already prove efficient in solving this task. This first ablation result highlight the importance of active learning and negative examples for the critic's training. To complement this result, in **Ablation (2)**, we remove the actor evaluation but we generate negative samples artificially by sampling in the descriptor space. We show that ablation (2) is performing significantly worse than DCG-MAP-Elites on Ant tasks ($p < 10^{-3}$) but performs similarly on Hexapod-Omni and Walker-Uni. However, ablation (2) is performing significantly better than ablation (1) on Ant tasks ($p < 10^{-5}$), demonstrating the importance of both active learning *and* negative samples. Finally, in **Ablation (3)** we replace the descriptor-conditioned actor $\pi_\phi(\,.\mid d)$ with a normal actor $\pi_\phi(\,.\,)$. Ablation (3) is performing significantly worse on all omnidirectional tasks ($p < 0.05$), demonstrating the importance of the descriptor-conditioned actor for the critic's training. Ablation (3) performs equivalently to DCG-MAP-Elites on Walker-Uni.

6 CONCLUSION

In this work, we introduce DCG-MAP-Elites and demonstrate the benefits of having descriptor-conditioned gradients to evolve populations of large neural networks. We concurrently train a descriptor-conditioned actor, as a by-product of the critic's training, that can achieve a wide range of descriptors. Our method performs better than PGA-MAP-Elites by a significant margin in omnidirectional tasks while maintaining similar performance to PGA-MAP-Elites in unidirectional tasks. We also show that the synergy between the fitness improvement capabilities of the PG variations and the exploration capabilities of the GA variations is preserved, even in deceptive environments. On some tasks, the descriptor-conditioned policy demonstrates properties that are similar to the discrete archive, summarizing its capabilities into one single neural network and acting as a continuous archive. We think that distilling the archive into a single policy is a promising method as it enables to have less redundancy compared to a discrete archive in which most of the solutions can be similar, especially between close cells. The descriptor-conditioned policy could also negate the burden of dealing with archive of thousands of solutions in practical applications.

The benefits of combining DRL methods with MAP-Elites come with the limitations of MDP settings. Specifically, we are limited to evolving differentiable solutions and the foundations of DRL algorithms rely on the Markov property and full observability. In this work in particular, we face challenges with the Markov property because the descriptors depend on full trajectories. Thus, the scaled reward introduced in our method depends on the full trajectory and not only on the current state and action. The performance of the descriptor-conditioned policy also shows that there is room for improvement to better distill the knowledge of the archive.

For future work, we would like to investigate the generalization capabilities of the descriptor-conditioned policy trained with DCG-MAP-Elites and try to produce solutions with descriptors that are not in the archive, effectively performing descriptor space interpolation. In our method, the critic attempts to mutate solutions to produce offspring with higher fitness while keeping their descriptors constant. We think that we could use the descriptor-conditioned critic to mutate solutions to produce offspring towards different descriptors, thereby explicitly promoting diversity.

REFERENCES

[1] Felix Chalumeau, Raphael Boige, Bryan Lim, Valentin Macé, Maxime Allard, Arthur Flajolet, Antoine Cully, and Thomas Pierrot. 2022. Neuroevolution is a Competitive Alternative to Reinforcement Learning for Skill Discovery. https://doi.org/10.48550/arXiv.2210.03516 arXiv:2210.03516 [cs].

[2] Felix Chalumeau, Thomas Pierrot, Valentin Macé, Arthur Flajolet, Karim Beguir, Antoine Cully, and Nicolas Perrin-Gilbert. 2022. Assessing Quality-Diversity Neuro-Evolution Algorithms Performance in Hard Exploration Problems. http://arxiv.org/abs/2211.13742 arXiv:2211.13742 [cs].

[3] Konstantinos Chatzilygeroudis, Antoine Cully, Vassilis Vassiliades, and Jean-Baptiste Mouret. 2020. Quality-Diversity Optimization: a novel branch of stochastic optimization. http://arxiv.org/abs/2012.04322 arXiv:2012.04322 [cs, math, stat].

[4] Konstantinos Chatzilygeroudis, Vassilis Vassiliades, and Jean-Baptiste Mouret. 2018. Reset-free Trial-and-Error Learning for Robot Damage Recovery. *Robotics and Autonomous Systems* 100 (Feb. 2018), 236–250. https://doi.org/10.1016/j.robot.2017.11.010 arXiv:1610.04213 [cs].

[5] Cédric Colas, Joost Huizinga, Vashisht Madhavan, and Jeff Clune. 2020. Scaling MAP-Elites to Deep Neuroevolution. *CoRR* abs/2003.01825 (2020), 67–75. arXiv:2003.01825 https://arxiv.org/abs/2003.01825

[6] Antoine Cully, Jeff Clune, Danesh Tarapore, and Jean-Baptiste Mouret. 2015. Robots that can adapt like animals. *Nature* 521, 7553 (May 2015), 503–507. https://doi.org/10.1038/nature14422 arXiv:1407.3501 [cs, q-bio].

[7] Antoine Cully and Yiannis Demiris. 2017. Quality and Diversity Optimization: A Unifying Modular Framework. http://arxiv.org/abs/1708.09251 arXiv:1708.09251 [cs].

[8] Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. 2021. Go-Explore: a New Approach for Hard-Exploration Problems. http://arxiv.org/abs/1901.10995 arXiv:1901.10995 [cs, stat].

[9] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. 2018. Diversity is All You Need: Learning Skills without a Reward Function. http://arxiv.org/abs/1802.06070 arXiv:1802.06070 [cs].

[10] Manon Flageat, Felix Chalumeau, and Antoine Cully. 2023. Empirical analysis of PGA-MAP-Elites for Neuroevolution in Uncertain Domains. *ACM Transactions on Evolutionary Learning* 3, 1 (2023), 1–32. http://arxiv.org/abs/2210.13156

[11] Manon Flageat, Bryan Lim, Luca Grillotti, Maxime Allard, Simón C. Smith, and Antoine Cully. 2022. Benchmarking Quality-Diversity Algorithms on Neuroevolution for Reinforcement Learning. http://arxiv.org/abs/2211.02193 arXiv:2211.02193 [cs].

[12] Matthew C. Fontaine and Stefanos Nikolaidis. 2021. Differentiable Quality Diversity. https://doi.org/10.48550/arXiv.2106.03894 arXiv:2106.03894 [cs].

[13] Matthew C. Fontaine and Stefanos Nikolaidis. 2023. Covariance Matrix Adaptation MAP-Annealing. https://doi.org/10.48550/arXiv.2205.10752 arXiv:2205.10752 [cs].

[14] C. Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. 2021. Brax – A Differentiable Physics Engine for Large Scale Rigid Body Simulation. http://arxiv.org/abs/2106.13281 arXiv:2106.13281 [cs].

[15] Scott Fujimoto, Herke van Hoof, and David Meger. 2018. Addressing Function Approximation Error in Actor-Critic Methods. http://arxiv.org/abs/1802.09477 arXiv:1802.09477 [cs, stat].

[16] Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. 2016. Variational Intrinsic Control. http://arxiv.org/abs/1611.07507 arXiv:1611.07507 [cs].

[17] Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. 2016. Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates. http://arxiv.org/abs/1610.00633 arXiv:1610.00633 [cs].

[18] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. http://arxiv.org/abs/1801.01290 arXiv:1801.01290 [cs, stat].

[19] Nikolaus Hansen. 2016. The CMA Evolution Strategy: A Tutorial. http://arxiv.org/abs/1604.00772 arXiv:1604.00772 [cs, stat].

[20] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. 1989. Multilayer feedforward networks are universal approximators. *Neural Networks* 2, 5 (1989), 359–366. https://doi.org/10.1016/0893-6080(89)90020-8

[21] 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. 2017. Population Based Training of Neural Networks. https://doi.org/10.48550/arXiv.1711.09846 arXiv:1711.09846 [cs].

[22] Marija Jegorova, Stéphane Doncieux, and Timothy Hospedales. 2019. Behavioural Repertoire via Generative Adversarial Policy Networks. In *2019 Joint IEEE 9th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. https://doi.org/10.1109/ICDL-EpiRob44920.2019 arXiv:1811.02945 [cs, stat].

[23] Saurabh Kumar, Aviral Kumar, Sergey Levine, and Chelsea Finn. 2020. One Solution is Not All You Need: Few-Shot Extrapolation via Structured MaxEnt RL. http://arxiv.org/abs/2010.14484 arXiv:2010.14484 [cs].

[24] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2019. Continuous control with deep reinforcement learning. http://arxiv.org/abs/1509.02971 arXiv:1509.02971 [cs, stat].

[25] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing Atari with Deep Reinforcement Learning. http://arxiv.org/abs/1312.5602 arXiv:1312.5602 [cs].

[26] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (Feb. 2015), 529–533. https://doi.org/10.1038/nature14236 Number: 7540 Publisher: Nature Publishing Group.

[27] Jean-Baptiste Mouret and Jeff Clune. 2015. Illuminating search spaces by mapping elites. http://arxiv.org/abs/1504.04909 arXiv:1504.04909 [cs, q-bio].

[28] Olle Nilsson and Antoine Cully. 2021. Policy gradient assisted MAP-Elites. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, Lille France, 866–875. https://doi.org/10.1145/3449639.3459304

[29] Georg Ostrovski, Pablo Samuel Castro, and Will Dabney. 2021. The Difficulty of Passive Learning in Deep Reinforcement Learning. http://arxiv.org/abs/2110.14020 arXiv:2110.14020 [cs].

[30] Thomas Pierrot and Arthur Flajolet. 2023. Evolving Populations of Diverse RL Agents with MAP-Elites. https://openreview.net/forum?id=CBfYffLqWqb

[31] Thomas Pierrot, Valentin Macé, Félix Chalumeau, Arthur Flajolet, Geoffrey Cideron, Karim Beguir, Antoine Cully, Olivier Sigaud, and Nicolas Perrin-Gilbert. 2022. Diversity Policy Gradient for Sample Efficient Quality-Diversity Optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 1075–1083. https://doi.org/10.1145/3512290.3528845 arXiv:2006.08505 [cs].

[32] Justin K. Pugh, Lisa B. Soros, and Kenneth O. Stanley. 2016. Quality Diversity: A New Frontier for Evolutionary Computation. *Frontiers in Robotics and AI* 3 (2016). https://www.frontiersin.org/articles/10.3389/frobt.2016.00040

[33] Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. 2015. Universal Value Function Approximators. In *Proceedings of the 32nd International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 37)*, Francis Bach and David Blei (Eds.). PMLR, Lille, France, 1312–1320. https://proceedings.mlr.press/v37/schaul15.html

[34] Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. 2020. Dynamics-Aware Unsupervised Discovery of Skills. http://arxiv.org/abs/1907.01657 arXiv:1907.01657 [cs, stat].

[35] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529, 7587 (Jan. 2016), 484–489. https://doi.org/10.1038/nature16961 Number: 7587 Publisher: Nature Publishing Group.

[36] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. 2014. Deterministic Policy Gradient Algorithms. In *Proceedings of the 31st International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 32)*, Eric P. Xing and Tony Jebara (Eds.). PMLR, Bejing, China, 387–395. https://proceedings.mlr.press/v32/silver14.html

[37] Bryon Tjanaka, Matthew C. Fontaine, Aniruddha Kalkar, and Stefanos Nikolaidis. 2022. Training Diverse High-Dimensional Controllers by Scaling Covariance Matrix Adaptation MAP-Annealing. https://doi.org/10.48550/arXiv.2210.02622 arXiv:2210.02622 [cs].

[38] Bryon Tjanaka, Matthew C. Fontaine, Julian Togelius, and Stefanos Nikolaidis. 2022. Approximating Gradients for Differentiable Quality Diversity in Reinforcement Learning. https://doi.org/10.48550/arXiv.2202.03666 arXiv:2202.03666 [cs].

[39] Vassilis Vassiliades, Konstantinos Chatzilygeroudis, and Jean-Baptiste Mouret. 2017. Using Centroidal Voronoi Tessellations to Scale Up the Multi-dimensional Archive of Phenotypic Elites Algorithm. http://arxiv.org/abs/1610.05729 arXiv:1610.05729 [cs].

[40] Oriol Vinyals, Igor Babuschkin, Wojciech M. Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H. Choi, Richard Powell, Timo Ewalds, Petko Georgiev, Junhyuk Oh, Dan Horgan, Manuel Kroiss, Ivo Danihelka, Aja Huang, Laurent Sifre, Trevor Cai, John P. Agapiou, Max Jaderberg, Alexander S. Vezhnevets, Rémi Leblond, Tobias Pohlen, Valentin Dalibard, David Budden, Yury Sulsky, James Molloy, Tom L. Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Yuhuai Wu, Roman Ring, Dani Yogatama, Dario Wünsch, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy Lillicrap, Koray Kavukcuoglu, Demis Hassabis, Chris Apps, and David Silver. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 7782 (Nov. 2019), 350–354. https://doi.org/10.1038/s41586-019-1724-z Number: 7782 Publisher: Nature Publishing Group.