Overcoming Deceptive Rewards with Quality-Diversity

Arno Feiden
Fraunhofer SCAI
Sankt Augustin, Germany
arno.feiden@scai.fraunhofer.de

Jochen Garcke*
Institut für Numerische Simulation
Universität Bonn
Bonn, Germany
garcke@ins.uni-bonn.de

ABSTRACT

Quality-Diversity offers powerful ideas to create diverse, high-performing populations. Here, we investigate the capabilities these ideas hold to solve exploration-hard single-objective problems, in addition to creating diverse high-performing populations.

We find that MAP-Elites is well suited to overcome deceptive reward structures, while an Elites-type approach with an unstructured, distance based container and extinction events can even outperform it.

Furthermore, we analyse how the QD score, the standard evaluation of MAP-Elites type algorithms, is not well suited to predict the success of a configuration in solving a maze. This shows that the exploration capacity is an entirely different dimension in which QD algorithms can be utilized, evaluated, and improved on. It is a dimension that does not currently seem to be covered, implicitly or explicitly, by the current advances in the field.

CCS CONCEPTS

• **Computing methodologies** → **Continuous space search**; *Genetic algorithms*; Evolutionary robotics.

KEYWORDS

Quality-Diversity, Exploration, Deception, Maze, MAP-Elites

ACM Reference Format:
Arno Feiden and Jochen Garcke. 2023. Overcoming Deceptive Rewards with Quality-Diversity. In *Genetic and Evolutionary Computation Conference Companion (GECCO '23 Companion), July 15–19, 2023, Lisbon, Portugal.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3583133.3590741

1 INTRODUCTION

Reinforcement learning (RL) offers a flexible framework that can be applied to a variety of problems, including robotics, games, and natural language processing. While its foundations lay in optimization, it is often not necessary or even possible to find the optimal, but rather just a viable solution to a given problem setup. One can observe that the paradigm of optimization might even be a hindrance in some setups, for example when the objective is not suitably shaped for iterative improvement.

Balancing exploration and exploitation is a well-known challenge in reinforcement learning settings and exploration-hard problems with *sparse* or *deceptive* reward structures are a notable subject of ongoing research [5]. In sparse reward settings a reward occurs at the end of a sequence of non-rewarding actions. In deceptive reward settings, the reward landscape is rugged with local optima that have a large area of attraction that can trap an optimization algorithm.

Quality-Diversity (QD) algorithms emphasise finding a set of diverse solutions using the actual objective of the setup as a secondary guidance. These algorithms generate large and diverse populations to tackle the posed problem, but are typically less efficient. However, in exploration-hard settings they may be even more efficient than single-objective optimization methods, as the deceptive pull of the objective has less influence. Solving navigation tasks through different mazes is a classic problem requiring extensive exploration, that has been used to show just that [8].

In this paper, we study the capabilities of different approaches to train deep neural networks that solve mazes interpreted as a reinforcement learning problem with a deceptive reward structure. Solving the maze measures how well these algorithms are able to explore the solution space and avoid being trapped by local optima.

Thereby, we look at the potential of QD algorithms to solve such single objective problems even more effectively than goal-oriented algorithms. This potential may not be fully realized, because the output of QD algorithms is generally evaluated by looking at the whole population created and rarely reduced back to one solution.

We use a procedural generation of mazes to test learning algorithms for a large numbers of setups. The optimization task is to find a parametrization of a deep but relatively small neural network with a fixed architecture that acts as a policy for solving the maze. This provides a task that is complicated in exploration but relatively simple in neuroevolution. By re-running a similar setup in large numbers we can pick up even small performance differences in the different approaches.

We compare an RL algorithm, an *Evolutionary Strategy*, a classic *MAP-Elites* [9] (Multi-dimensional Archive of Phenotypic Elites) approach and also present a grid-less, *Voronoi-Elites* [7] approach, for which we include extinction events as a novel improvement. Using the latter approach, we observe the best performance in solving the maze and even get rid of the MAP-Elites requirement of a low-dimensional, bounded, well-aligned behaviour space.

2 BACKGROUND AND RELATED WORK

2.1 Quality-Diversity

Quality-Diversity encompasses a family of optimization algorithms that aim to output a large collection of diverse, high-performing

*also with Fraunhofer SCAI

GECCO '23 Companion, July 15–19, 2023, Lisbon, Portugal

ACM ISBN 979-8-4007-0120-7/23/07.
https://doi.org/10.1145/3583133.3590741

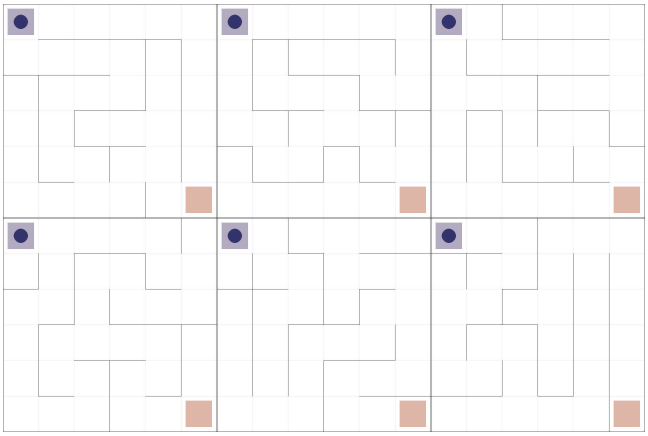

Figure 1: Starting position of six different maze setups with increasing complexity

solutions, instead of one, best-performing solution. These solutions are stored in a container, categorized by some function of their observed behaviour, the *behaviour characteristic* (BC). New solutions are developed through some evolutionary strategy. Cully and Demiris [4] distinguish QD algorithms along the axes type of container, selection of parents, and selection of offspring. MAP-Elites [9] type algorithms with a fixed, grid-based archive have sparked a lot of research and interest. MAP-Elites uses a distance measure to divide the behaviour space into cells that contain a (best performing) representative of the associated behaviour each. While the definition of good performance is given by the optimization problem, the definition of distance in behaviour space, i.e. what makes a collection diverse, is not imminently clear and typically handcrafted [3], [11].

2.2 Distinguishing Behaviour

The MAP-Elites setup requires an observation of behaviour, and some measure of difference between behaviour. Generally a projection or transformation of the observed data is used, which is called the behaviour descriptor or behaviour characteristic. This choice is an important, human decision [11]. Especially the grid-based approach of MAP-Elites requires a low-dimensional representation of the behaviour space, but it can be scaled to higher dimensional behaviour spaces as in Cluster- or CVT-Elites [14]. The behaviour space itself is usually handcrafted, but can also be constructed through dimension reduction or a secondary optimization. Grid-free approaches aim at curiosity, surprise, or novelty scores. Curiosity is associated with the idea of how useful of a stepping stone a certain solution is for further solutions. Surprise, how well an observing algorithm knows the data, and novelty how different the observed data is to previously observed data [1].

2.3 Evaluating QD Algorithms

Reinforcement learning algorithms can be easily evaluated by the peak performance reached, or by performance in relation to an expended resource, be it wall time, computational effort, or amount of interaction with the environment. The definition of performance is directly tied to the problem itself through the reward function. QD algorithms generally underperform in these categories. Typically the goal of QD algorithms is described by the *QD Challenge*, to find an archive of both diverse and high-performing solutions. Its success can be quantified by coverage of the behaviour space through the solutions of this archive weighted by their respective performance, the *QD Score* [4]. Clearly this is an advantageous technique to generate a repertoire of behaviours, if that is the goal. Other advantages are recognized but harder to measure: adaptability, i.e. recovery for a damaged robot [3] or improved exploration, i.e. in an environment with sparse or deceptive rewards [11].

2.4 Scaling to Complex Problems

Evolutionary search is not efficient in exploring the large parameter space associated with deep neural networks, a problem that is addressed by including gradient-based search [10] or exploiting the potential for parallel search [2]. While we will discuss a simple control problem with a complex reward structure, techniques exist to lift the employed techniques to more complex control tasks.

3 METHOD

3.1 Soft Actor Critic

Soft Actor Critic (SAC) [6] is an off-policy reinforcement learning algorithm that trains a stochastic policy. This randomness promotes exploration. The influence of the stochastic element itself is variable through the optimization process. We use the benchmark implementation from stable-baselines3 [12].

3.2 Evolutionary Strategy with Novelty

Evolutionary Strategy with Novelty-Search and Reward (NSR-ES) [2] is a representative of a *Novelty Search with Local Competition* (NSLC), the second big branch of Quality-Diversity algorithms [4]. A small population of 5 agents is updated by progressively picking one agent, repeatedly adding noise to its parametrization, and evaluating the change in incurred reward and novelty (difference in behaviour) compared to other agents. A gradient-like step is then applied to this agent, adding a weighted sum of the evaluated noise-updates, with the weights according to either both novelty and reward (NSR-ES), to only novelty (NS-ES) or only reward (ES).

3.3 MAP-Elites

We use the pyribs [13] implementation of MAP-Elites [9] as a baseline. Here, we take the approach of an underlying genetic algorithm with Gaussian noise added to the parameters of a random parent to form a child organism. The MAP-Elites algorithm is varied over the definition and meshing of the behaviour space, following the idea of alignment of the behaviour characteristic with the goal [11]. A child organism is made a candidate for a bin by categorization through a behaviour characteristic. If this bin is occupied by another candidate, the candidate with better performance takes over as representative for this bin.

3.4 Voronoi-Elites with Extinction Events

We utilize a 1-nearest-neighbour strategy to create a grid-free Elites algorithm, also conceptualized as Voronoi-Elites [7]. Here,

Algorithm 1 Voronoi-Elites with Extinction Events

Require: With N normal population size, $\tilde{N}$ extinction population size, d distance measure, f fitness function, n_e extinction frequency, m_g number of generations, m_b batch size, Gaussian noise $\mathcal{N}(\mu, \sigma)$.

Initialize population container P with N random organisms.
for $i \in [1, ..., m_g]$, $j \in [1, ..., m_b]$ **do**
  $x \leftarrow$ sample P + sample $\mathcal{N}(\mu, \sigma)$.
  Add x to P
  while $(|P| > N) \mid (i \mod n_e = 0 \ \& \ |P| > \tilde{N})$ **do**
    $x, y \leftarrow \arg\min_{x,y \in P, x \neq y} d(x, y)$
    Remove $\arg\min_{\xi \in \{x,y\}} f(\xi)$ from P
return P

an archive of fixed size holds a set of agents characterized by their behaviour. The behaviour space comes equipped with a distance measure. New organisms are created by randomly picking an organism in the archive and adding Gaussian noise to its parametrization. It is then identified with their behaviour and added to the archive. Whenever the archive has gotten bigger than its fixed size, the two organisms closest in the behaviour space compete and the one with lower performance is deleted from the archive.

We find that *extinction events*, a regular shrinking of the population size, act as a catalyst for improved exploration in this setup. The algorithm naturally induces a way to reduce the total size, by progressively letting the closest organisms in the archive compete. This improves the focus of the random search in the parameter space to efficiently find a solution. It also introduces two additional hyperparameters, the frequency of extinction events n_e and smaller population size during the extinction event $\tilde{N}$, see algorithm 1.

4 EXPERIMENT

4.1 Deceptive Maze

All maze tasks are setup as a 6-by-6 Cartesian grid. Between any grid cell may be a wall. The state-space is continuous as $[0, 6]^2$. The agent starts in the middle of cell $(0, 0)$ and wins with 1 point when getting anywhere in the cell $(5, 5)$. There is always a way to the goal. For every learning task, we normalize the state-space. The action-space is $[-0.5, 0.5]^2$, every action changing the state along this vector. A wall in the path will stop any further movement in this direction, but keeps intact any movement along it. The final reward of an episode that is not won after 72 steps is defined as 1 minus the normalized linear distance to the goal: $1 - \frac{|x_{\text{goal}} - x_{\text{pos}}|}{|x_{\text{goal}} - x_{\text{start}}|}$. This generates a deceptive trap with every cul-de-sac and makes the reward of an episode defined only by the last observed state of the episode.

4.2 Solving the Maze

We compare the different algorithms regarding their ability to overcome the deceptive reward structure. The underlying optimization problem is to find a parametrization of a feed-forward neural net with two hidden layers of size 16 each. This architecture is fixed for all learning tasks.

Table 1: Average mazes solved

Method	BC	% Solutions	Avg. reward
SAC	n/a	16.3 / 18.3	0.58
ES	n/a	17.7 / 60	0.524
NS-ES	last position	19.7 / 64.2	0.243
NSR-ES	last position	21.9 / 63.5	0.481
ME	last position	86.8	0.969
	mean	87.8	0.974
	direction	86.8	0.967
VE	last position	94.2	0.991
	mean	91.5	0.983
	direction	89.5	0.973
	full trajectory	90	0.981

We judge the success of the algorithms by their ability to solve 100 generated mazes, i.e. to find a solution that deterministically moves the agent from start to goal. To better understand the Elites-type algorithms, we also measure the QD score normalized by the population size.

We represent non-Elites algorithms through soft-actor critic and evolutionary search. These algorithms may see a solution once, but they will not immediately keep this solution. We therefore give two numbers in table 1: the number of mazes for which the algorithm outputs a solution and the number of mazes for which the algorithm sees a solution at any point. For Elites-type algorithms these are identical.

For the Elites algorithms, we run the same experiment for several behaviour characteristics (BC), which are functions of a rollout of the agent: *Last position* the last state, *full trajectory* all the states as a time series, *mean* the mean values of state and action over time. The Euclidean distance of BCs of agents defines the distance of those agents. *Direction* splits the rollout in 5 equally long parts and encodes the dominant direction of movement, north, east, west, or south. The distance between two agents is then defined as the count of disagreements on those 5 parts.

Voronoi-Elites use population size 72, extinction frequency 20, extinction population size 20. MAP-Elites use a regular Cartesian grid, *last position* a 2-dimensional grid with 2304 cells, *direction* a 5-dimensional grid with 1024 cells, *mean* a 4-dimensional grid with 4096 cells. All use an initial Gaussian noise $\mathcal{N}(\mathbf{0}, \mathbf{1})$ to vary the parameters of the parents, that act as the weights of the neural network.

We run the learning process for 2000 generations with a batch size of 100, observing 14.4M state-action pairs for both the ES and the Elites algorithms. For the comparison with SAC, we run learning for the same amount of timesteps. We repeat the experiment with six different seeds and average over the results, totaling in 600 trials for each method.

4.3 Results

MAP-Elites is well suited to overcome the deceptive reward and solve the maze, while the gradient-based methods only see solutions in less than two thirds of mazes. The pull of the gradient traps the algorithms in local optima and prevents sufficient exploration.

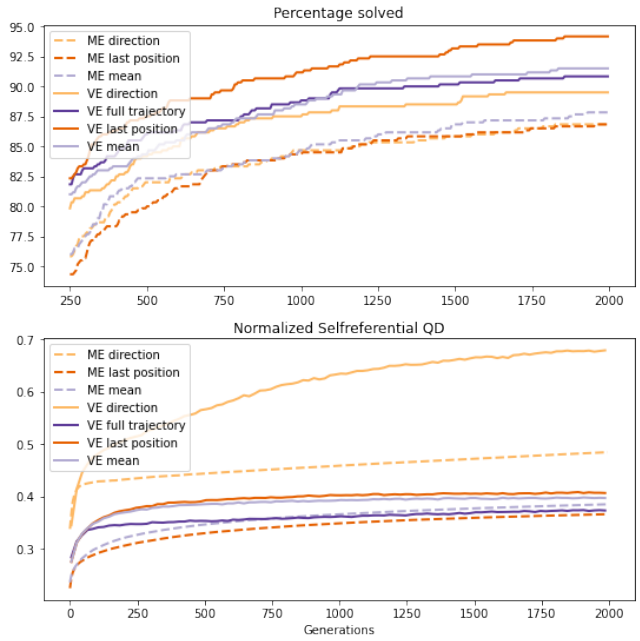

Figure 2: Comparison of Elites-type algorithms

Voronoi-Elites with extinction events improves the performance of the MAP-Elites approach, see table 1 and figure 2. The most successful runs of Voronoi-Elites leaves 4 mazes unsolved, the most successful run of classic MAP-Elites leaves 10 unsolved. The BC type influences the capability of the algorithm to solve the maze, but the choice of underlying algorithm matters more.

The BC type does strongly affect the QD score. We find that difference in performance of the different Elites-type algorithms is not reflected in their respective QD scores, the prime evaluator of QD algorithms, see also figure 2: The *direction* BC achieves a high QD score, because it allows the solutions to clump nearer to the goal, but performs worse or equally to the other BCs in actually solving the mazes. MAP-Elites with the *last position* BC achieves the lowest QD score, MAP-Elites with *direction* BC highest of MAP-Elites, but both solve the same number of mazes. The 3 Voronoi-Elites with QD scores in between the two, all solve more mazes.

The average score proves to be a similarly deceptive indicator, when looking at SAC and the ES-algorithms: The NS-ES approach finds significantly lower scoring organisms than SAC or ES but will still return more solutions to the deceptive problem (see table 1).

5 DISCUSSION

We present a problem that on the one hand does not require a complicated, very deep neural network and parametrization thereof, but on the other hand features a reward landscape that is complex within these boundaries. While gradient-based approaches are not successful even if they feature some kind of exploration, we find that MAP-Elites is well equipped to tackle this problem setup.

Voronoi-Elites with extinction events improves on the performance. This is especially interesting, because these distance based containers are somewhat overlooked in research [1]. The presented algorithm simplifies the application, as the behaviour space now need neither be low-dimensional nor bounded. However, extinction events require two new hyperparameters, extinction frequency and extinction population size, which need understanding and tuning.

We find that the ability to overcome deceptive reward structures is not adequately measured by the conventional ways the success of reinforcement learning or Quality-Diversity algorithms is measured. The focus on the QD Challenge obfuscates the capability in single-objective optimization. Since advances in MAP-Elites algorithms are typically measured along QD scores, their potential to explore and avoid local optima is generally not sufficiently measured. If neither the QD score nor the performance regarding reward indicate success in overcoming deceptive reward structures, how can an algorithm even be tested in that regard?

Genetic algorithms are not considered an effective way to evolve complex neural networks, and although it is difficult to extrapolate the observed success of Elites-type algorithms to complex problems, we conjecture that with a small base population and a grid-free archive, other ways to generate offspring may be feasible.

Of special interest is the influence of the extinction events: If its success can be replicated in other grid-free setups but also why extinction events help finding a solution. Further analysis of the phenomenon may give a deeper understanding of the strengths and weaknesses of these algorithms with unstructured containers.

REFERENCES

[1] Konstantinos Chatzilygeroudis, Antoine Cully, Vassilis Vassiliades, and Jean-Baptiste Mouret. 2021. *Quality-Diversity Optimization: A Novel Branch of Stochastic Optimization.* Springer International Publishing, Cham, 109–135.

[2] Edoardo Conti, Vashisht Madhavan, Felipe Petroski Such, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. 2018. Improving Exploration in Evolution Strategies for Deep Reinforcement Learning via a Population of Novelty-Seeking Agents. In *NeurIPS'18.* 5032–5043.

[3] Antoine Cully, Jeff Clune, Danesh Tarapore, and Jean-Baptiste Mouret. 2015. Robots that can adapt like animals. *Nature* 521 (05 2015), 503–507.

[4] Antoine Cully and Y. Demiris. 2017. Quality and Diversity Optimization: A Unifying Modular Framework. *IEEE Transactions on Evolutionary Computation* 22 (2017), 245–259.

[5] Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth Stanley, and Jeff Clune. 2021. First return, then explore. *Nature* 590 (02 2021), 580–586.

[6] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In *ICML'18 (PMLR, Vol. 80).* 1861–1870.

[7] Alexander Hagg, Mike Preuss, Alexander Asteroth, and Thomas Bäck. 2020. An Analysis of Phenotypic Diversity In Multi-Solution Optimization. In *BIOMA 2020.* Springer, 43–55.

[8] Joel Lehman and Kenneth O. Stanley. 2011. Abandoning Objectives: Evolution Through the Search for Novelty Alone. *Evolutionary Computation* 19, 2 (2011), 189–223.

[9] Jean-Baptiste Mouret and Jeff Clune. 2015. Illuminating search spaces by mapping elites. arXiv:1504.04909

[10] Thomas Pierrot, Valentin Macé, Felix 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 *(GECCO '22).* 1075–1083.

[11] Justin K. Pugh, L. B. Soros, Paul A. Szerlip, and Kenneth O. Stanley. 2015. Confronting the Challenge of Quality Diversity *(GECCO '15).* 967–974.

[12] Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. 2021. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *JMLR* 22, 268 (2021), 1–8.

[13] Bryon Tjanaka, Matthew C. Fontaine, David H. Lee, Yulun Zhang, Trung Tran Minh Vu, Sam Sommerer, Nathan Dennler, and Stefanos Nikolaidis. 2021. pyribs: A bare-bones Python library for quality diversity optimization. https://github.com/icaros-usc/pyribs.

[14] Vassilis Vassiliades, Konstantinos Chatzilygeroudis, and Jean-Baptiste Mouret. 2016. Using Centroidal Voronoi Tessellations to Scale Up the Multidimensional Archive of Phenotypic Elites Algorithm. *IEEE Transactions on Evolutionary Computation* 22 (2016), 623–630.