Can the Problem-Solving Benefits of Quality Diversity Be Obtained Without Explicit Diversity Maintenance?

Ryan Boldi
University of Massachusetts Amherst
Amherst, MA
rbahlousbold@umass.edu

Lee Spector
Amherst College
Amherst, MA
lspector@amherst.edu

ABSTRACT

When using Quality Diversity (QD) optimization to solve hard exploration or deceptive search problems, we assume that diversity is extrinsically valuable. This means that diversity is important to help us reach an objective, but is not an objective in itself. Often, in these domains, practitioners benchmark their QD algorithms against single objective optimization frameworks. In this paper, we argue that the correct comparison should be made to *multi-objective* optimization frameworks. This is because single objective optimization frameworks rely on the aggregation of sub-objectives, which could result in decreased information that is crucial for maintaining diverse populations automatically. In order to facilitate a fair comparison between quality diversity and multi-objective optimization, we present a method that utilizes dimensionality reduction to automatically determine a set of behavioral descriptors for an individual, as well as a set of objectives for an individual to solve. Using the former, one can generate solutions using standard quality diversity optimization techniques, and using the latter, one can generate solutions using standard multi-objective optimization techniques. This allows for a level comparison between these two classes of algorithms, without requiring domain and algorithm specific modifications to facilitate a comparison.

CCS CONCEPTS

• **Mathematics of computing** → *Dimensionality reduction*; • **Theory of computation** → **Evolutionary algorithms**; • **Computing methodologies** → **Artificial intelligence**.

KEYWORDS

Quality Diversity, Multi-Objective Optimization, Dimensionality Reduction, Benchmarking

ACM Reference Format:
Ryan Boldi and Lee Spector. 2023. Can the Problem-Solving Benefits of Quality Diversity Be Obtained Without Explicit Diversity Maintenance?. In *Genetic and Evolutionary Computation Conference Companion (GECCO '23 Companion), July 15–19, 2023, Lisbon, Portugal.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3583133.3596336

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

ACM ISBN 979-8-4007-0120-7/23/07…$15.00
https://doi.org/10.1145/3583133.3596336

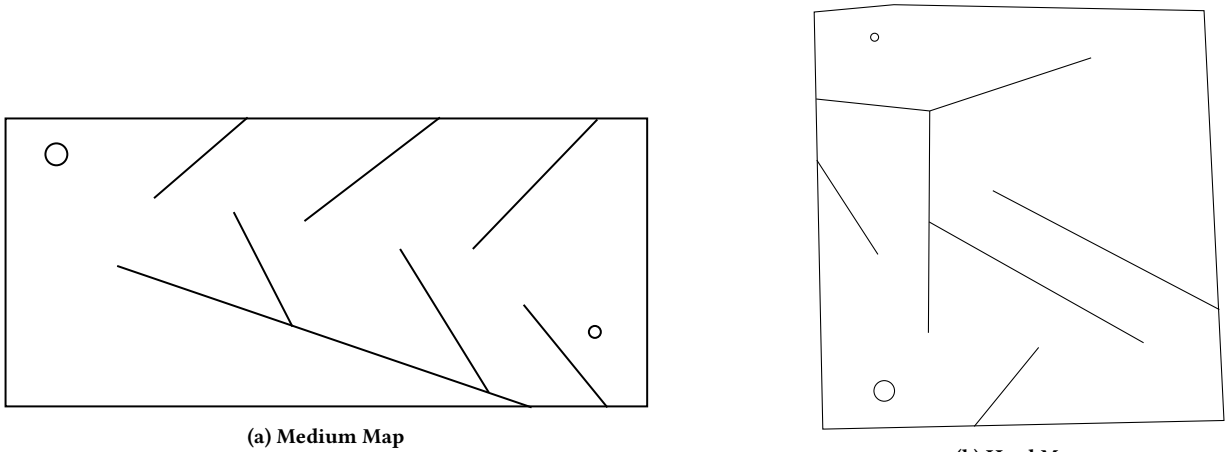
1 INTRODUCTION AND BACKGROUND

Quality Diversity (QD) optimization is a relatively recent advancement in the evolutionary computation literature where a diverse set of high performing individuals are maintained over the course of the run. This results in *divergent*, rather than the traditional *convergent* evolutionary search [21].

There are often two reasons to apply quality diversity optimization. The first of them is to generate a large archive of qualitatively diverse individuals that solve certain problems. For example, finding diverse sets of robot behaviors [3, 12] or creating diverse video game or training scenarios [6, 7]. The second reason is generally to solve hard exploration problems that often have deceptive reward signals. For example, Lehman and Stanley [15] explore using Novelty Search (NS) to solve a deceptive maze, where there is a single goal, although many ways to solve this goal. They also used NS to evolve a controller for bipedal locomotion that outperformed fitness-based search with the particular fitness function and selection scheme studied. Some more recent examples are the QD-Maze brought forward by Pugh et al. [22], or the modular robotics domain used by Nordmoen et al. [19], which also have a single (real) objective. Despite the only goal for experimenters being solving the single objective, there is the assumption that diversity here is instrumental to solving the task. The intuition behind this makes sense: diverse low-performing solutions might be stepping stones that lead to high performing solutions in the future. This paper focuses on bench-marking Quality Diversity on the second use case, as an exploration algorithm that ultimately solves a single problem, although possibly in unexpected ways.

Multi-objective optimization (MOO) is a common paradigm in optimization literature that attempts to optimize for multiple objectives at the same time. For example, *non dominated sorting genetic algorithm* (NSGA2) [4] and *strength pareto evolutionary algorithm* (SPEA2) [24] both deal with situations where there are multiple different objectives that often have numerous trade-offs for each other. Lexicase selection, a parent selection technique, has also been found to be useful at optimizing for multiple objectives [13]. There are often many different solutions that can be made by having different trade-offs between objectives, which can be visualized as existing on a Pareto front. These solutions could be highly diverse, as individuals that trade off between objectives differently would likely have qualitatively different behavior [11, 14] These methods can therefore be used as a form of implicit quality diversity optimization: the quality comes from solving objectives, and the diversity comes from solving different combinations of objectives.

How do you determine whether a quality diversity algorithm is performing well in hard exploration or deceptive domains? Previous studies compare quality diversity to using single objective



(a) Medium Map

(b) Hard Map

Figure 1: Two deceptive mazes, adapted from [15]. In both maps, the large circle is the starting point of the robot and the small circle represents the goal. Both maps are deceptive in the sense that the path that minimizes euclidean distance to the goal will get stuck at local optima.

optimization in reaching the objective. For example, Lehman and Stanley [15] compare novelty search to single-objective fitness based optimization in maze domains. Pugh et al. [21] compare quality diversity techniques to using only a single distance metric as an objective. Gaier et al. [8] compare using MAP-Elites to a (single) image similarity objective in evolving a target image. There are notably few studies comparing it to multi-objective or multi-modal optimization. Vassiliades et al. [23] compared MAP-Elites and NS to other multi-modal selection schemes for a maze navigation task. Nordmoen et al. [19] compared MAP-Elites [16] to a single- and multi-objective optimization algorithm. However, the multi-objective algorithms simply use diversity as a secondary objective, which does not necessarily provide a fair ground between QD algorithms and MOOs as QD has access to all dimensions of diversity, where MOOs simply get access to an aggregated version of this information. Other work looked at combining QD with MOO, but not on comparing them to each other (partly due to them having different goals in the majority of the cases) [17, 20].

When comparing QD's problem solving ability to single objective or limited multi-objective optimization paradigms, the information accessible to both systems is not the same, preventing a fair comparison. In this paper, we consider the problem of providing quality diversity algorithms with the same information as that available to multi-objective optimization. This allows for both MOOs and QD to be compared faithfully in their ability to maximize objective(s) where diversity is instrumental (extrinsic), as opposed to being a goal that we are trying to optimize in and of itself (intrinsic).

To do this, we propose the use of a dimensionality reduction technique to generate the behavioral descriptors for the quality diversity methods, and a slightly augmented version of this model to generate the objective values for MOO. This allows for a faithful comparison between both techniques as they will have access to the same performance signal.

2 MOTIVATION

To motivate the use of comparison scheme like this, we discuss the potential effects of learning the fitness function from hand-written measures (as it usually is in the literature), and provide a motivating example that shows how optimizing for combinations of these objectives through multi-objective optimization could result in the maintenance of diversity *automatically*.

A common motivating example for quality diversity algorithms aimed at solving deceptive single objective problems is one of the deceptive hard maze. Two mazes that are of medium and hard difficulty can be found in Figure 1. These mazes are both deceptive as an individual greedily moving towards the goal will get stuck at sub-optimal traps. For this reason, diversity must be emphasized to ensure that the goal is actually reached. This fits into scope of the problems discussed in this paper as the sole objective is to reach the goal, yet diversity is intstrumental in reaching this objective.

Consider what an arbitrary QD algorithm could do in this scenario. For example, MAP-elites [16] could be implemented with hand written behavioral descriptors such as 1) distance to goal on x axis, 2) distance to goal on y axis, 3) total distance travelled, and perhaps 4) number of turns taken. Using measure functions like these, solving deceptive mazes like above would be relatively straightforward, as demonstrated by Pugh et al. [21].

Why not single objectives? Single objectives often rely on an aggregation of sub-objectives that each represent qualitatively different goals. This aggregation procedure results in a loss of information regarding the performance of an individual.

Instead, we use multi-objective optimization directly on the sub-objectives to help facilitate the discovery of high performing individuals. With MOOs, then, finding solutions that optimize combinations of the sub-objectives creates diversity in the population similar to that with QD. However, the diversity that results from this is implicitly optimized for, as opposed to it being explicitly optimized for like in QD. What this means is QD maintains diversity

as an objective in-and-of itself, whereas MOOs only optimize for their given objectives.

In order to use multi-objective optimization on this domain using the scheme we present in this paper, we make an assumption that the fitness function (proximity to goal) can be approximated as a linear combination of these 4 measures. It is important to note that the raw value of the fitness function need not matter, as long as it matches the original fitness function in ranking the individuals. It is clear that both x and y distance are negatively correlated with proximity to the goal. For the sake of the example, let us suppose that the total distance travelled could also have a negative (yet smaller in magnitude) correlation to proximity to the goal. Finally, let us say that the number of turns is neutral to fitness (individuals that turn more on average perform no better than those that turn less).

We then set each feature as its own objective for multi-objective optimization. Each individual is evaluated on all 4 of these metrics, and individuals are selected based on the multi-objective optimization algorithm being used. Consider an individual that has perfectly matched the x location of the goal. This individual will likely be selected regardless of its y location, distance, or turn values. Consider an individual that is very far from the goal, yet has covered a large amount of distance. This individual would also be selected, regardless of its distance to the goal. These different trade offs result in a diversity of individuals that solve different combinations of the sub-problems. When using a multi-objective selection scheme like Lexicase selection, this diversity is maintained automatically until convergence to a final goal. [10]

Importantly, these systems have access to the same information as the quality diversity optimization algorithms. However, the key difference here is that MOOs could solve these deceptive problems *without explicitly maintaining diversity*. If QD algorithms can outperform MOOs in this domain, this would mean that the diversity that that specific QD algorithm maintains is instrumental in overcoming the deception in this domain. In the next section, we discuss how to extend this comparison scheme to any domain, regardless of the existence of human-written behavioral descriptors.

3 COMPARISON SCHEME

In this section, we will bring together all the ideas presented in this paper into a simple scheme to fairly compare quality diversity algorithms with objective-based algorithms. We operate under the assumption that humans have little intuition about the domain, and that reasonable choices behavioral descriptors or objectives will therefore not be obvious in advance.

3.1 From Phenotype to Measures

Consider learning of a set of measures $m(\theta)$ as an unsupervised learning task. This can be learned with a variational autoencoder (VAE, [5]) as done in some previous work in QD [2, 9].

Figure 2 outlines an example autoencoder architecture that could be used to learn a latent embedding of a given phenotype. Using reconstruction loss, this autoencoder can be used to learn a compressed representation of a phenotype in lower-dimensional space. This makes it possible for QD algorithms such as MAP-Elites to cover a large set of behavioral niches without needing to store a high dimensional archive.

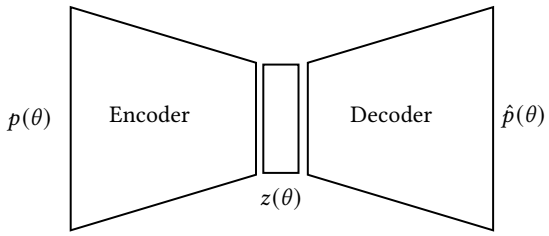


Figure 2: Variational autoencoder architecture used to learn a latent embedding $z(\theta)$ of a phenotype $p(\theta)$. As this is a variational autoencoder, $z(\theta)$ is sampled from a distribution that is parameterized by the output of the encoder (not pictured here for simplicity).

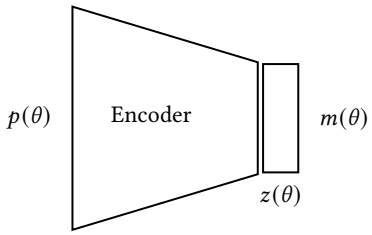


Figure 3: Using the encoder of a VAE to predict measures $m(\theta)$ of a phenotype $p(\theta)$

Figure 3 shows how the architecture outline in Figure 2 can be used to generate the qualitative measures $m(\theta)$ that can be used to emphasize diversity using one of many QD techniques.

3.2 From Measures to Fitness

In order to facilitate a comparison between quality diversity techniques and multi-objective optimization techniques, one should try to present them both with as close to the same information as possible.

In order to apply multi-objective optimization, we need to extract the sub-objectives from both the phenotype $p(\theta)$ and the ground truth fitness function $f(\theta)$. This can be done by simply augmenting the dimensionality reduction architecture used to generate the measures. We assume that the ground truth fitness can be approximated by a linear combination of the measure functions. Although this is a large assumption, it is one that is commonly made in the reward learning literature [1]. In inverse reinforcement learning (IRL), where the task is to learn a reward function that explains an expert trajectory, it is often the case that this reward function is learned as a linear combination of state features ϕ_s [18]. More recent work in IRL uses a set of pretraining algorithms to learn the state features (either with an autoencoder, or predicting a forward dynamics model, etc) before learning the final linear combination of these features that results in a reward function [1].

[1]It is possible to relax this assumption to be that fitness is a non-linear function of the measures by incorporating a second multi-layer perception that learns the fitness from the output of the encoder. However, a similar de-aggregation procedure would be necessary on the penultimate layer of the newly added MLP.

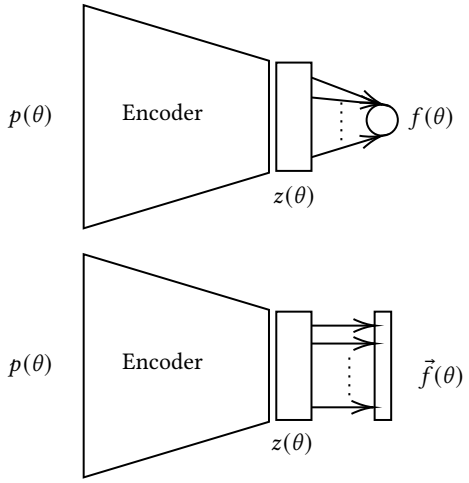


Figure 4: Top: Learning the weights for a linear combination of features that sums to an approximation for the true fitness function $f(\theta)$. These weights can then be used to predict fitness from measures $m(\theta) = z(\theta)$, or even straight from the phenotype $p(\theta)$. Bottom: De-aggregation of last layer of learned fitness model to result in a set of sub-objectives $\vec{f}(\theta)$ that sum to an approximation of the ground truth fitness $f(\theta)$.

Figure 4 gives an overview of the training procedure of the fitness prediction model. First, after having trained the autoencoder to a satisfactory level, we freeze the encoder weights, and add a final layer that leads to a single fitness value. We can update the weights of this last layer (through back-propagation, or any other linear regression technique), to accurately predict the fitness value. Then, once the weights are determined, we can *de-aggregate* them into a vector of sub-fitness values. These values correspond well to sub-objectives for multi-objective optimization algorithms.

3.3 Bringing It All Together

Given a genotype $g(\theta)$, we can create a phenotype $p(\theta)$ and a gound truth fitness value $f(\theta)$. This is domain specific but does not depend on which optimization algorithm being used. Examples of the phenotype could be the trajectory of a reinforcement learning agent, raw sensory information from a robot, or any other characterization of how the genotype behaves after translation. Then, we train a dimensionality reduction model (such as an autoencoder) to learn a lower dimensional representation of the phenotype $z(\theta)$.

Quality Diversity. Using the lower dimensional representation, we can assign the measures to simply be the values of the latent variables (i.e. $m(\theta) = z(\theta)$). Then, we can assign the fitness value to be the ground truth fitness $f(\theta)$. This is very similar to the procedure done by Cully [2] to automatically discover the measure functions.

Multi-objective Optimization. We can then augment our measure function to include a final layer that predicts the ground truth fitness. Once this model is trained to convergence, we simply de-aggregate the final node, resulting in an element-wise multiplication of the measures and their corresponding weights. These values $\vec{f}(\theta)$ should sum approximately to the ground truth fitness $f(\theta)$. These fitness functions can then be used as the multiple objectives for any MOO scheme.

Fair Comparison? What metric do we use to evaluate how these methods are performing? Since the domain is one that ultimately does have a single objective, we should measure the performance of the algorithms by how well they solve this objective. Including a comparison about diversity would not be yield a fair performance comparison as only QD methods explicitly optimize for diversity. So, in the context of solving hard exploration problems, we should be measuring how well each algorithm solves these problems. The authors suggest simply using the ground truth fitness function of the individuals produced through both optimization paradigms.

3.4 Practical Details

In practice, using the scheme we presented requires some choices to be made by the user. In this section, we discuss such details and considerations that experimenters attempting to use this comparison scheme should be aware of.

Training the Models. In order to train the models, one needs to amass a significant amount of training data. Cully [2] solves this issue by using the archive of individuals produced through a run of MAP-elites to train the dimensionality reduction model. As we are attempting to form a comparison scheme between two classes of algorithms, it is important that the model used does not vary much between these two systems. In order to this, the authors see 2 options:

(1) Pre-trained: Generate the training data from a series of prior evolutionary runs (or perhaps exhaustively enumerating all of genotype space, if tractable), and train the models using that. Use the entire dataset to train the VAE, and then do a second pass to learn the weights of the linear layer given the ground truth fitness as a label.
(2) Incremental: Train the models with some random phenotypes from randomly sampling from genotype space, and fine-tune the models *differently* for each system, using the phenotypes cached throughout their respective evolutionary runs and the ground truth fitness.

In order to facilitate a fair comparison, the authors recommend method (1). This is because method (2) results in two different models being used to extract the information from phenotype used for selection. This could result in the lack of clarity regarding each system's performance: was it due to actually solving the domain better, or due to generating better training data for the models to later perform better using?

Dimensionality of the latent layer. Another practical consideration to be made is the extent to which we shrink dimensions using the dimensionality reduction technique. If the bottleneck layer is too small, it could result in poor reconstruction capability. If the bottleneck layer is too large, it could prohibit effective learning by the optimization algorithms. This is a domain specific consideration, but could be addressed through a series of hyperparameter optimization runs. For example, one could run AURORA [2] using various latent layer sizes, and take the latent layer size that results

in the highest QD-Score. As this is simply a benchmark, the cost of training the model can be amortized over many benchmarking tasks.

4 CONCLUSION AND FUTURE WORK

We have presented a method to compare quality diversity techniques to objective based techniques for a variety of hard exploration or deceptive problems. The key insight is that the comparison to single objective optimization is not a fair comparison due to the aggregation that occurs when we aggregate an individual's performance to be a single number. Each individual solves different sub-objectives to different extents. Optimizing for different combinations of all these sub-objectives could correspondingly also extrinsically optimize for diversity without needing to explicitly be tasked to. This means we can compare the quality of solutions that are created by explicitly optimizing for diversity to directly optimizing for the sub-objectives in their ability to use extrinsic diversity in their favor.

We present a dimensionality reduction technique based on an autoencoder to automatically learn the measure functions from a given phenotype. Then, fitness is approximated as a weighted sum of these measure values. Finally, we remove the summation operation, and set the sub-objectives to simply be an element-wise multiplication of the measure and its weight (i.e. positive weighted measures are increased, negatives decreased). We can perform multi-objective optimization on the sub-objectives, and quality diversity straight on the measures (with access to the ground truth fitness).

Future work in improving this benchmark should address the limitation that the sub-objectives might not be a function of the measures. This could be addressed by training a new fitness prediction architecture that does not re-use weights from the measure predicting autoencoder. Simply de-aggregating the penultimate layer on this model would similarly allow for multiple sub-objectives that are learned straight from the phenotype.

ACKNOWLEDGMENTS

The authors acknowledge Antoine Cully, Bill Tozier, Edward Pantridge, Li Ding, Maxime Allard, Nic McPhee, Thomas Helmuth, and the members of the PUSH Lab at Amherst College for discussions and inspiration that helped shape this work. Furthermore, the authors would like to thank the anonymous reviewers for their careful reading and insightful comments and discussion.

This material is based upon work supported by the National Science Foundation under Grant No. 2117377. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] Daniel S. Brown, Russell Coleman, Ravi Srinivasan, and Scott Niekum. 2020. Safe Imitation Learning via Fast Bayesian Reward Inference from Preferences. http://arxiv.org/abs/2002.09089 arXiv:2002.09089 [cs, stat].

[2] Antoine Cully. 2019. Autonomous skill discovery with Quality-Diversity and Unsupervised Descriptors. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 81–89. https://doi.org/10.1145/3321707.3321804 arXiv:1905.11874 [cs].

[3] Antoine Cully and Jean-Baptiste Mouret. 2015. Evolving a behavioral repertoire for a walking robot. *Evolutionary Computation* (2015).

[4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (2002), 182–197. https://doi.org/10.1109/4235.996017

[5] Carl Doersch. 2016. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908* (2016).

[6] Sam Earle, Justin Snider, Matthew C. Fontaine, Stefanos Nikolaidis, and Julian Togelius. 2022. Illuminating Diverse Neural Cellular Automata for Level Generation. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Boston, Massachusetts) *(GECCO '22)*. Association for Computing Machinery, New York, NY, USA, 68–76. https://doi.org/10.1145/3512290.3528754

[7] Matthew C. Fontaine and Stefanos Nikolaidis. 2022. Evaluating Human–Robot Interaction Algorithms in Shared Autonomy via Quality Diversity Scenario Generation. *ACM Transactions on Human-Robot Interaction* 11, 3 (Sept. 2022), 1–30. https://doi.org/10.1145/3476412

[8] Adam Gaier, Alexander Asteroth, and Jean-Baptiste Mouret. 2019. Are quality diversity algorithms better at generating stepping stones than objective-based search?. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. ACM, Prague Czech Republic, 115–116. https://doi.org/10.1145/3319619.3321897

[9] Luca Grillotti and Antoine Cully. 2022. Unsupervised Behaviour Discovery with Quality-Diversity Optimisation. http://arxiv.org/abs/2106.05648 arXiv:2106.05648 [cs].

[10] Thomas Helmuth, Nicholas Freitag McPhee, and Lee Spector. 2016. Effects of Lexicase and Tournament Selection on Diversity Recovery and Maintenance. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion (GECCO '16 Companion)*. Association for Computing Machinery, New York, NY, USA, 983–990. https://doi.org/10.1145/2908961.2931657

[11] Thomas Helmuth, Nicholas Freitag McPhee, and Lee Spector. 2016. Lexicase Selection for Program Synthesis: A Diversity Analysis. In *Genetic Programming Theory and Practice XIII*, Rick Riolo, W.P. Worzel, Mark Kotanchek, and Arthur Kordon (Eds.). Springer International Publishing, Cham, 151–167. https://doi.org/10.1007/978-3-319-34223-8_9 Series Title: Genetic and Evolutionary Computation.

[12] Seungsu Kim and Stéphane Doncieux. 2017. Learning highly diverse robot throwing movements through quality diversity search. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. ACM, 1177–1178.

[13] William La Cava, Thomas Helmuth, Lee Spector, and Jason H Moore. 2019. A probabilistic and multi-objective analysis of lexicase selection and ε-lexicase selection. *Evolutionary Computation* 27, 3 (2019), 377–402.

[14] Marco Laumanns, Lothar Thiele, Eckart Zitzler, and Kalyanmoy Deb. 2002. Archiving With Guaranteed Convergence And Diversity In Multi-objective Optimization. In *Annual Conference on Genetic and Evolutionary Computation*.

[15] Joel Lehman and Kenneth O. Stanley. 2011. Abandoning Objectives: Evolution Through the Search for Novelty Alone. *Evolutionary Computation* 19, 2 (June 2011), 189–223. https://doi.org/10.1162/EVCO_a_00025

[16] 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].

[17] Jean-Baptiste Mouret and Glenn Maguire. 2020. Quality diversity for multi-task optimization. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*. 121–129.

[18] Andrew Y. Ng and Stuart J. Russell. 2000. Algorithms for Inverse Reinforcement Learning. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML '00)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 663–670.

[19] Jørgen Nordmoen, Frank Veenstra, Kai Olav Ellefsen, and Kyrre Glette. 2021. MAP-Elites enables Powerful Stepping Stones and Diversity for Modular Robotics. *Frontiers in Robotics and AI* 8 (2021).

[20] Thomas Pierrot, Guillaume Richard, Karim Beguir, and Antoine Cully. 2022. Multi-objective quality diversity optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 139–147.

[21] 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 (July 2016). https://doi.org/10.3389/frobt.2016.00040

[22] Justin K. Pugh, L. B. Soros, Paul A. Szerlip, and Kenneth O. Stanley. 2015. Confronting the Challenge of Quality Diversity. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*. ACM, Madrid Spain, 967–974. https://doi.org/10.1145/2739480.2754664

[23] Vassilis Vassiliades, Konstantinos Chatzilygeroudis, and Jean-Baptiste Mouret. 2017. Comparing multimodal optimization and illumination. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. ACM, Berlin Germany, 97–98. https://doi.org/10.1145/3067695.3075610

[24] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. 2001. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. (2001).