Quantifying Efficiency in Quality Diversity Optimization

Bryon Tjanaka
tjanaka@usc.edu
University of Southern California
Los Angeles, California, USA

Matthew C. Fontaine
mfontain@usc.edu
University of Southern California
Los Angeles, California, USA

Stefanos Nikolaidis
nikolaid@usc.edu
University of Southern California
Los Angeles, California, USA

ABSTRACT

In quality diversity (QD) optimization, the QD score is a holistic metric which sums the objective values of all cells in the archive. Since the QD score only measures the performance of a QD algorithm at a single point in time, it fails to reflect algorithm efficiency. Two algorithms may have the same QD score even though one algorithm achieved that score with fewer evaluations. We propose a metric called "QD score AUC" which quantifies this efficiency.

CCS CONCEPTS

• **General and reference** → **Metrics**; • **Computing methodologies** → **Machine learning**.

KEYWORDS

quality diversity, metrics, benchmarks

Quality diversity (QD) optimization seeks to find a diverse collection of high-performing solutions to a given problem [1, 3]. More formally, given an *objective function* $f(\boldsymbol{\phi})$ and k *measure functions* $m_i(\boldsymbol{\phi})$, the goal of QD is to find solutions which span the outputs of the measure functions while maximizing the objective function [2].

Solutions output by a QD algorithm are stored in an *archive*, a multi-dimensional grid of cells where the dimensions correspond to the measure function outputs $m_i(\boldsymbol{\phi})$ and each cell stores one solution and its associated objective value.

A common metric for QD algorithm performance is the QD score [3, 4], which sums the objective value of the solution in every archive cell. If an archive has M cells, its QD score is defined as

$$\text{QD score} = \sum_{i=1}^{M} f(\boldsymbol{\phi}_i) \tag{1}$$

Note that if a cell's solution $\boldsymbol{\phi}_i$ does not exist, then $f(\boldsymbol{\phi}_i)$ is defined as 0. Furthermore, to prevent individual solutions from decreasing the QD score, the objective value $f(\boldsymbol{\phi}_i)$ is always assumed to be non-negative (this is usually achieved via normalization).

QD algorithms are typically *iterative*, i.e. they generate a batch of solutions, evaluate the solutions' objectives and measures, and repeat. By looking at QD score, we only see a QD algorithm's performance at a certain iteration. In other words, QD score only reflects *current* performance, with no indication of *intermediate* performance.

To illustrate why it may be problematic to look only at the current performance, consider Fig. 1, which plots the QD score of two algorithms against the number of solutions they have evaluated so far. Although Algorithm 1 and Algorithm 2 attain identical final QD scores, we would prefer Algorithm 1 since its intermediate performance reflects greater *efficiency* — namely, Algorithm 1 achieves higher QD scores earlier in the run.

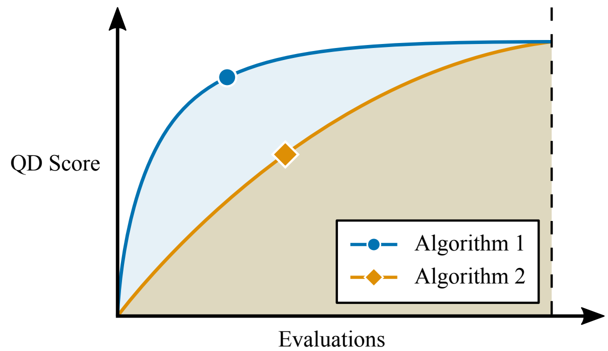

Figure 1: A plot of the QD score obtained by two fictional QD algorithms during a single run. While both algorithms obtain the same QD score at the end of the run, it is clear that Algorithm 1 is more efficient at QD optimization, since it achieves higher QD scores earlier. To quantify this difference, we can record the area under the QD score curve of each algorithm — we term this metric the "QD score AUC." Now, we see that Algorithm 1 is more efficient than Algorithm 2, since its QD score AUC is larger.

To quantify the efficiency of a QD algorithm, we propose the *QD score AUC* metric, which is the area under the curve (AUC) of the algorithm's QD score vs. evaluations plot. Drawing from prior work [5, 6], we formally define the QD score AUC as a Riemann sum:

$$\text{QD score AUC} = \sum_{i=1}^{N} (\lambda * \text{QD score at iteration } i) \tag{2}$$

where we assume that a QD algorithm runs for N iterations and evaluates a batch of λ solutions on each iteration. In cases where the QD algorithm generates solutions asynchronously rather than in batch, we set $\lambda = 1$.

The QD score AUC is most useful when two algorithms have similar final QD score. In such cases, the algorithm with the higher QD score AUC will be more efficient. Hence, we envision the QD score AUC being reported in works where the QD algorithms considered have similar final performance.

ACKNOWLEDGMENTS

The authors thank the anonymous reviewers for their invaluable feedback. This work was partially supported by the NSF NRI (#1053128) and NSF GRFP (#DGE-1842487).

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. https://doi.org/10.1007/978-3-030-66515-9_4

[2] Matthew C. Fontaine and Stefanos Nikolaidis. 2021. Differentiable Quality Diversity. *Advances in Neural Information Processing Systems* 34 (2021). https://proceedings.neurips.cc/paper/2021/file/532923f11ac97d3e7cb0130315b067dc-Paper.pdf

[3] 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), 40. https://doi.org/10.3389/frobt.2016.00040

[4] 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* (Madrid, Spain) *(GECCO '15)*. Association for Computing Machinery, New York, NY, USA, 967–974. https://doi.org/10.1145/2739480.2754664

[5] Konstantinos Sfikas, Antonios Liapis, and Georgios N. Yannakakis. 2021. Monte Carlo Elites: Quality-Diversity Selection as a Multi-Armed Bandit Problem. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Lille, France) *(GECCO '21)*. Association for Computing Machinery, New York, NY, USA, 180–188. https://doi.org/10.1145/3449639.3459321

[6] Bryon Tjanaka, Matthew C. Fontaine, Julian Togelius, and Stefanos Nikolaidis. 2022. Approximating Gradients for Differentiable Quality Diversity in Reinforcement Learning. arXiv:2202.03666 [cs.LG] https://dqd-rl.github.io https://dqd-rl.github.io.