Objective Sampling Strategies for Generalized Locomotion Behavior with Lexicase Selection

Jared M. Moore[1] and Adam Stanton[2]

[1]School of Computing,
Grand Valley State University, Allendale, MI, USA
[2]School of Computing and Mathematics,
Keele University, Keele, ST5 5BG, UK
jared.moore@gvsu.edu, a.stanton@keele.ac.uk

Abstract

Controllers capable of exhibiting multiple behaviors is a long-standing goal in artificial life. Evolutionary robotics approaches have demonstrated effective optimization of robotic controllers, realizing single behaviors in a variety of domains. However, evolving multiple behaviors in one controller remains an outstanding challenge. Many objective selection algorithms are a potential solution as they are capable of optimizing across tens or hundreds of objectives. In this study, we use Lexicase selection evolving animats capable of both wall crossing and turn/seek behaviors. Our investigation focuses on the objective sampling strategy during selection to balance performance across the two primary tasks. Results show that the sampling strategy does not significantly alter performance, but the number of evaluations required varies significantly across strategies.

Introduction

Increasing the generality of robotic controllers is an important area of development in autonomous systems. This "general capability" can mean both that a controller performs well across variations of a single task (semi-generalized control), and/or competent in multiple, orthogonal task domains. Automatic design of effective controllers in problems requiring a combination of these two forms of generality, i.e. competency in multiple task domains where each task entails multiple related sub-tasks, is a challenging but achievable goal in evolutionary robotics (ER).

In previous work we have shown that Lexicase selection (Spector, 2012) is an effective many-objective selection operator realizing semi-generalized control in a quadruped wall-crossing task (Moore and Stanton, 2017). Evolved controllers crossed the majority of wall heights encountered, outperforming previous evolutionary strategies custom-designed for the task (Stanton and Channon, 2013). One hundred wall heights are treated as individual objectives within the wall-crossing task, resulting in a 100 objective search space (see Figure 1 for an example at maximum height). However, there is significant overlap between objectives since the overall task is to exhibit a semi-generalized locomotion behavior enabling crossing walls of any height to reach a target objective. Lexicase selection compares individuals in a tournament objective by objective. Once an individual is identified that is better than the others in an objective, that individual is selected. Given this ability to evaluate a subset of the population objective by objective, Lexicase can reduce computational overhead in high-dimensional spaces.

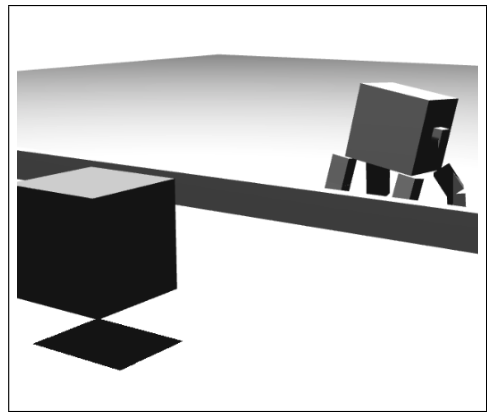

Figure 1: Agent facing the target, separated from it by the wall. In this case the wall is at maximum height (objective 100).

In this paper, we expand on our previous investigations by adding a second locomotion task, *turn and seek*. Turn and seek requires an animat to locate a target placed anywhere on an arc from $-90°$ to $90°$ in $1.8°$ increments. In the same way as in *cross wall*, the task is broken up into 100 objectives. Animats are evolved across all 200 possible objectives, also referred to as environments, attempting to elicit generalized control in both task domains. In this study, we investigate different methods of sampling from the 200 objectives during evolutionary time, assessing which strategies lead to high generalized performance and which are most computationally efficient. Five treatments sample from the 200 objective space in different ways. Although we expect all treatments to evolve some baseline level of effective performance, our hypothesis is that altering the sampling strategy will emphasize different evolutionary characteristics in Lexicase, resulting in performance differences in the final populations.

Results show that every two-task sampling strategy investigated evolves effective performance across the two tasks. Further, significant differences in performance do not arise between two-task treatments. However, the sampling strategy applied can significantly alter how many objectives, and thus how many evaluations, are conducted during an evolutionary run. We discuss this result through a novel analysis of the sub-objectives in each task, examining the *filtering power* of each objective and how differences between objectives assessed by this metric affect the operation of Lexicase selection. We conclude that due to implicit probabilistic bias towards sub-objectives separating populations more effectively in the multidimensional fitness space, naive Lexicase sampling of the full 200 objective space is as effective as the other treatments proposed for these two tasks. This is an important finding since a naive sampling strategy is more computationally efficient and avoids the creation of an additional parameter that must be specified at design time.

Background and Related Work

Evolutionary robotics techniques (Nolfi and Floreano, 2000), applied specifically to robot controller design, have demonstrated effective behaviors in legged animat locomotion (Baydin, 2012; Clune et al., 2009) and the transfer of controllers to reality (Ruud et al., 2016; Koos et al., 2010; Stanton, 2018) amongst many others. Multiple objective algorithms are increasingly being used to improve not only performance but also aspects of resilience to damage, behavioral robustness, and controller generalizability (Pinville et al., 2011). These secondary objectives enhance robotic systems, often by drawing on additional fitness metrics derived from biological observation (Moore and McKinley, 2016). Evolving multiple behaviors has arisen as a challenge for the field with approaches including behavioral diversity (Doncieux and Mouret, 2013) and evolving multiple movement behaviors in one platform such as walking, turning, and jumping (Huizinga and Clune, 2019).

Generalized behaviors encompass: the ability to learn and react to environmental information across multiple unique environments (Lehman et al., 2013), the capability to adapt and reconfigure due to damage (Kriegman et al., 2019), and the expression of multiple locomotive behaviors in one robot (Cully et al., 2015). In many cases, multi-objective (Deb et al., 2002) and many-objective algorithms like Lexicase selection enable evolving across multiple fitness metrics. In this paper, we expand on earlier investigations (Moore and Stanton, 2017, 2018, 2019) by adding a second meta-task to the wall crossing task. Lexicase selection allows us to evolve animats across 200 individual objectives. For these large search spaces, downsampling objectives for consideration during Lexicase selection (Helmuth and Spector, 2020; Hernandez et al., 2019) is an effective strategy to reduce computational overhead by only considering a subset of the objective space per generation. In this paper, we continue to employ downsampling, assessing individuals in up to 10 objectives per generation as in previous wall-crossing experiments.

Methods

Quadrupedal Animat Figure 1 shows the animat which has a cuboid torso with four legs placed at the lower corners. Each two segment leg is connected to the torso with 2-degree of freedom (DOF) hip joints allowing for up/down and side-to-side sweeping movement. The knees are 1-DOF allowing the legs to extend or contract toward the middle of the torso. Two sensors placed on the left and right side of the torso provide positional input to the controller for the relationship between the animat and the target. The animat is evolved on a flat high-friction surface.

Controller Animats are controlled by feed-forward artificial neural networks (ANN) as in (Moore and Stanton, 2020). 16 inputs provide: 2 periodic signals to promote oscillatory motion, 2 inputs for animat position relative to the target, and joint feedback for the 12 leg joints. A hidden layer comprising 12 nodes connects the inputs to the output nodes. 12 outputs provide a control signal for each DOF in the animat. Hips have 2-DOF requiring two inputs to report joint orientation. An animat's genome consists of 336 evolvable weights for each connection between nodes.

Tasks Two tasks are designed to elicit unique behaviors from animats in the simulation. The wall crossing task requires an animat to move to a target placed on the opposite side of a variable height wall. Wall heights range from very short to the height of the animat's hip over 100 gradations. Each wall height is an objective used in selection. Figure 2 shows the second task, turn and seek, requiring an animat to move toward a target placed on a semicircle spanning from the animat's left, to front, and finally to its right. 100 objectives are also created for this task advancing along the semicircular arc by $1.8°$.

Evolutionary Algorithm with Lexicase Selection Populations of 50 individuals are evolved for 5,000 generations with downsampled ϵ-Lexicase selection (La Cava et al., 2016). 20 replicate runs are conducted per treatment. Objectives are shuffled per generation and sampled from either the entire 200 objective space, or limited to a subset of the objective space depending on the treatment. During a selection event, 5 individuals are drawn randomly from the population competing in up to 10 objectives downsampled from the total possible 200. While the subset of individuals is being compared on an objective an ϵ of 10% is applied consistent with earlier work (Moore and Stanton, 2020). Effectively, any individual within 90% of the best individual's performance on that objective is considered tied and moves

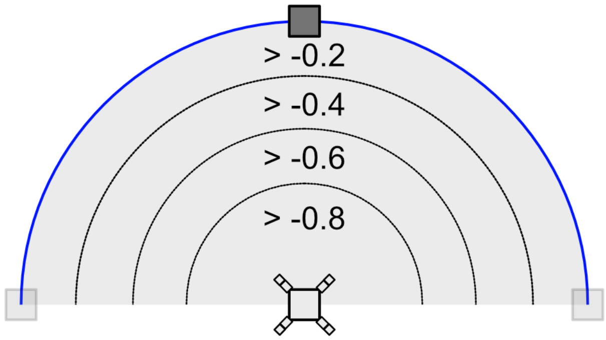

Figure 2: The turning task is broken down into 100 sub-objectives with the target box placed in the range of $-90° - 90°$. Fitness represents how far the animat is from the target at the end of the simulation. (Not to scale.)

on to the next objective in the selection process with the best individual. This ϵ value is effective in continuous objective space domains for metrics like distance to target as small performance differences don't substantially separate two individuals in terms of observed behavior. If all 10 objectives are exhausted and more than one individual remains, we randomly select one of the remaining individuals and record the selection as a tiebreak event.

Treatments Our primary focus is to examine how objectives should be sampled per generation from the two tasks to evolve effective generalized control. Three primary treatments alter the objective sampling mechanics:

1. *naive_2t* acts as our baseline and samples 10 objectives uniformly across the two tasks. The order of the sampled tasks is random, preventing one task from always being the first used in selection which could bias the algorithm. This is the normal operation of Lexicase selection.

2. *even-shuf_2t* samples 5 objectives from the wall-crossing task and 5 objectives from the turning task. The 10 objectives are then shuffled to randomize the order of occurrence during selection.

3. *flipN_2t* treatments sample 10 objectives from a single task per generation. After N generations, the task being sampled from flips and objectives are sampled from the next task. For example, for $N = 1$ we change objectives from wall-crossing to turning every other generation. Two treatments with $N = 1$ and $N = 50$ are evaluated.

Results and Discussion

One Task vs Two Task Performance In previous work we have evolved individuals for wall crossing across 100 unique environments (objectives). To provide a baseline for the other treatments in this study, *naive_1t* establishes a benchmark in both wall crossing and turning tasks. *naive_1t* individuals are only evolved on wall crossing as in previous investigations, but some of the behaviors for wall crossing carry over to turning as the target exists in both tasks. The best individuals for *naive_1t* are selected solely based on their wall crossing performance, whereas the *naive_2t* treatments described previously consider performance across both tasks.

Figures 3 and 4 plot the distributions of the best individual per replicate for the *naive_1t* and *naive_2t* treatments. Individuals in both treatments evolve high performance on low wall heights but differences arise on moderate wall heights. Performance in both tapers off as wall heights reach their upper limit. This is consistent with earlier results, as high wall heights require that the animat evolve a very specific behavior to cross. Still, *naive_1t* significantly outperforms *naive_2t* in wall crossing effectiveness across wall heights using a Wilcoxon rank-sum test with Bonferroni correction. This statistical test is used when reporting significance throughout this study. For turning, *naive_1t* demonstrates the similarities between the two tasks as targets placed directly in front of the animat are still solvable even though the individuals were not evolved for this task. However, performance tapers off drastically as the target moves to the sides with many individuals not exhibiting substantial movement toward the target. In this task, *naive_2t* significantly outperforms *naive_1t* evolving near perfect performance by generation 500.

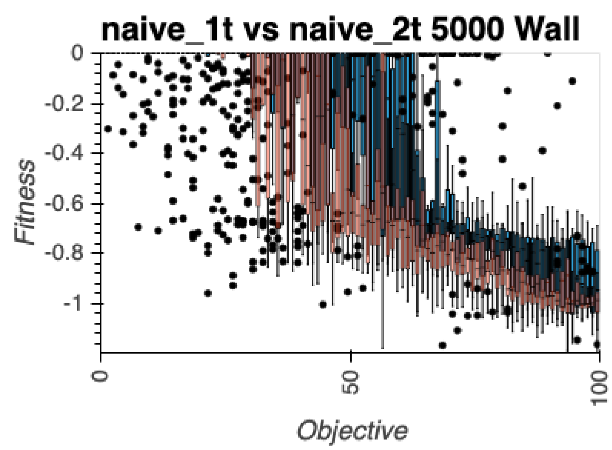

Figure 3: *naive_1t* (blue) and *naive_2t* (red) best individual per replicate in wall crossing task. Fitness scores below -0.4 correspond to individuals not being able to cross the wall obstacle.

The inclusion of turning as a task and subsequent increase in the number of objectives in *naive_2t* does hinder performance in wall crossing. Since 200 objectives are now present and uniformly sampled from, the difference in performance may be that the wall crossing objectives aren't assessed as much during evolution. Accordingly, we conduct a

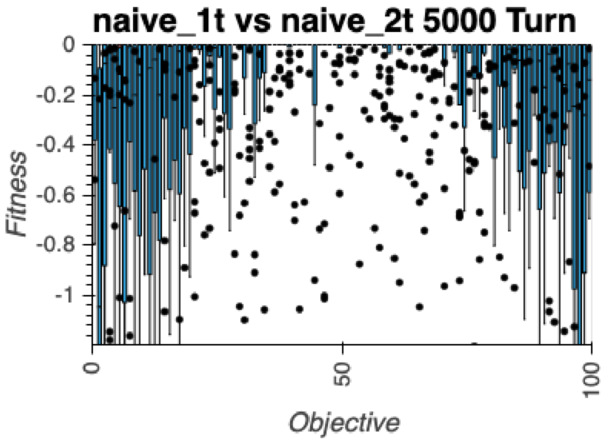

Figure 4: *naive_1t* (blue) and *naive_2t* (red) best individual per replicate in the turning task. *naive_2t* evolves perfect performance in this task so boxes are not visible.

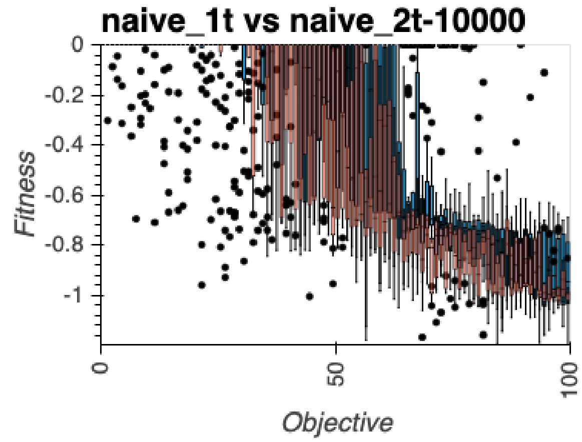

Figure 5: *naive_1t* (blue) and *naive_2t-10000* (red) best individual per replicate in the wall crossing task. No significant difference in performance between the two.

second *naive_2t* treatment evolving animats for 10,000 generations. The doubling of generations should compensate for the potential loss of wall crossing objective selection in *naive_2t*. We find that indeed, there is no significant difference in performance between *naive_1t* and *naive_2t-10000* in wall crossing, see Figure 5. However, there is also no significant difference between *naive_2t* and *naive_2t-10000* in wall crossing. *naive_2t-10000* falls in between *naive_1t* and *naive_2t* for wall crossing performance. A naive approach is capable of attaining similar performance in wall crossing, while evolving the ability to navigate to a target placed in a semicircle around it but only after twice the generations as a naive approach evolved only for wall crossing. Our goal with the remaining treatments is to investigate whether or not altering the objective sampling mechanism allows for effective two behavior performance to evolve in the same number of generations as one task.

Two Task Treatment Performance Figure 6 plots the performance in wall crossing, turning, and mean performance across tasks for all treatments conducted in this study at generation 5,000. As shown in the figure, all two task treatments evolve effective turning behaviors with no significant difference in performance between them. Furthermore, performance in wall crossing is similar for all two task treatments as well after 5,000 generations. The main performance difference remains that *naive_1t* does not evolve effective turning performance as it is not evolved in that task.

Lexicase Dynamics We next examine the dynamics of Lexicase selection across the treatments. Although performance is similar, the characteristics of the selection process vary. Figure 7 plots the number of environments per replicate considered across all selection events illustrating how much "deeper" Lexicase went into the subsampled objectives to separate individuals.

Figure 8 plots the number of simulations per replicate considered across all selection events quantifying how many individuals were evaluated in the subsampled objectives under consideration. Together, these two metrics indicate the computational efficiency of a treatment, and dynamics of the two tasks influence on Lexicase. As shown in both figures, the *flip* treatments have considerable disparity between the *wall* and *turn* tasks. Objectives are sampled from either *wall* or *turn* within a generation in contrast to *naive_2t* and *even-shuf_2t*. The high occurrence of environments and individual evaluations in the *flip* treatments in turning indicates that the task is not as efficient at separating individuals as wall crossing. That is, in wall crossing, individuals are separated within relatively few subsampled objectives resulting in fewer environments needing to be evaluated during selection.

naive_2t and *even-shuf_2t* do not show this disparity between tasks, presumably due to the mixing of both wall crossing and turning tasks being sampled for an individual selection event. Although *naive_2t* does not enforce that objectives be sampled from both tasks, in practice uniform sampling across two tasks with the same number of objectives results in an average of half and half split. Indeed, we do not observe a significant difference in the number of environments considered during selection, nor in the total number of individual evaluations between the two *non-flipping* treatments. Considering the disparities between *flipping* and *non-flipping* treatments in Figures 7 and 8, we hypothesize that the turning environment is not as effective a filter as wall crossing, resulting in more objectives having to be considered, and consequently more individual simulations having to be conducted.

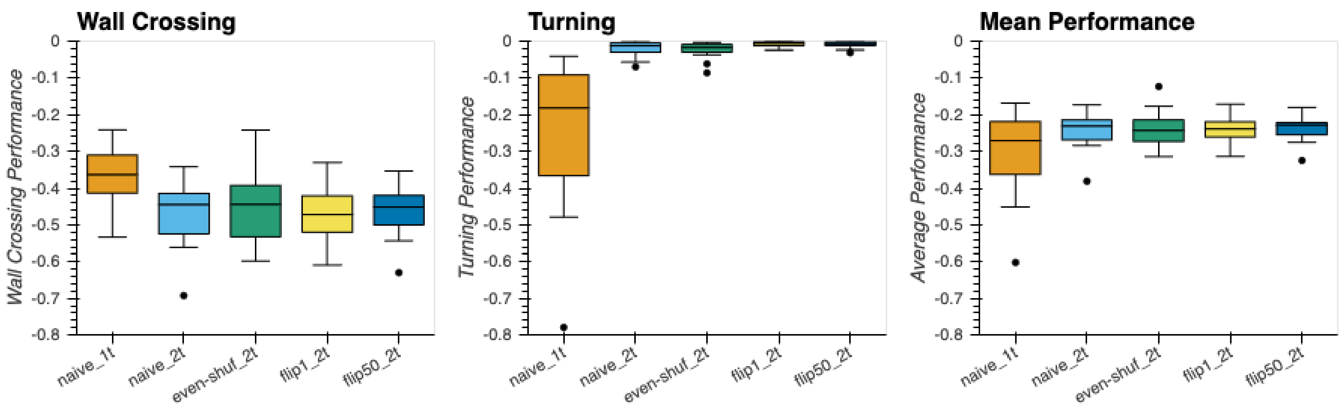

Figure 6: Performance of the best individual by average performance between the two tasks per replicate at the end of evolution across all 5,000 generation treatments.

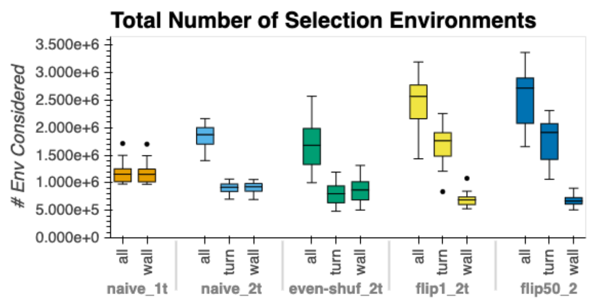

Figure 7: Total number of environments considered per replicate over evolutionary time. all is the aggregate of wall and turn tasks. Higher numbers indicate that selection typically required more environments per event to separate the best individual during the Lexicase operation.

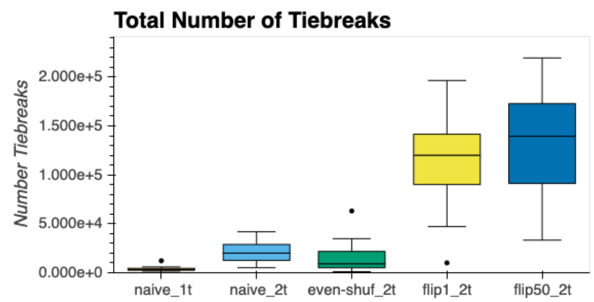

Figure 9: Number of tiebreaks across replicates per treatment. Tiebreaks indicate a failure to isolate one individual during the Lexicase selection process and are effectively a random selection event.

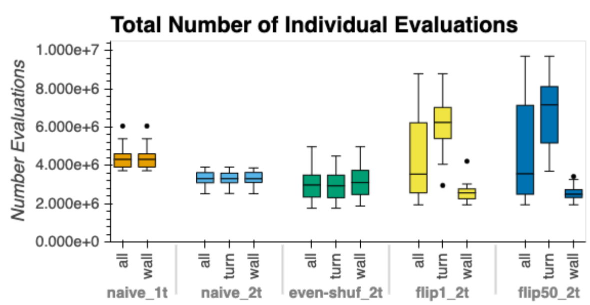

Figure 8: Total number of simulations per replicate over evolutionary time. all is the aggregate of wall and turn tasks. Simulations are the primary computational cost for this study.

Figure 9 plots the number of tiebreaks per treatment across replicates. Significant differences arise across treatments except between *naive_2t* and *even-shuf_2t*, and between *flip1_2t* and *flip50_2t*. The addition of turning significantly increases the number of tiebreaks between *naive_1t* and *naive_2t*, while switching between tasks in the *flip* treatments apparently further increases the number of tiebreaks.

Figure 10 plots the percentage of individuals in the selection subset filtered out by the objective in the left column, and the number of times an objective occurs in the right column. As shown in the figures, medium wall height objectives are effective filters in wall crossing. Short walls do not act as strong selectors as most individuals can cross them provided they evolve some form of locomotion, while tall walls are also weak selectors since most individuals cannot cross them. Middle wall heights are thus the better selectors as effective wall crossers make it past, while those without robust locomotion are filtered out. This filtering dynamic appears universal across all treatments. Turning demonstrates a different filtering characteristic where far left and far right targets are the strongest selectors while targets place to the front of the animat are less effective. Objectives placed in front and front-sides of the animat likely have strong similarity to the low wall heights in wall crossing as the essential task is to navigate to a target placed in front or nearly straight forward to the animat in both tasks. Sensory information would also be similar to wall crossing in these objectives. Thus the shape of the target placement manifests in the filtering plot.

The right side of Figure 10 shows the number of times an objective was used in the selection process for each treatment. For *naive_1t*, *naive_2t*, and *even-shuf_2t*, the objective sampling is uniform across objectives. The *flip* treatments, however, show a much different sampling. Here, turning objectives arise far more often as compared to wall crossing objectives. This disparity potentially indicates a difference in difficulty between the two tasks, as well as a selection effectiveness difference as more turning objectives have to be considered during selection.

Discussion

In this study, we examine three objective sampling strategies for two quadrupedal locomotion tasks. It appears that success in wall crossing does translate mildly to the turning task as *naive_1t* demonstrates upward drift in performance over time. Both tasks require locomotion and it seems likely that for two or more locomotion tasks there is going to be some behavioral overlap. Here we hypothesize that the overlap manifests as the low selective efficacy in the middle turning objectives as the target is placed at nearly the same location as-in wall crossing. No matter the wall crossing objective, there would be some pressure toward navigating to the target as in the turning task. This overlap may also aid in the prevention of catastrophic forgetting as *flip50_2t* does not suffer a significant decrease in performance despite alternating periods of 50 generations where wall crossing or turning are not used in selection.

There is not a large performance difference across two task treatments. Furthermore, doubling the number of generations to allow for approximately the same number of selections in wall crossing does not significantly improve performance for *naive_2t*. This indicates that the addition of a second task does not implicitly require a subsequent increase in evolutionary time. We note that the task relatedness is likely a factor here to consider as well, and subsequent investigation is planned to introduce conflicting tasks. Still, given that *even sampling* and *flipping* treatments require some parameterization it may be better to stay with a *naive* uniform sampling strategy and not overconstrain Lexicase.

Conclusions and Future Work

Evolving robots capable of accomplishing multiple tasks is a critical step towards generalized behavior. In this study, we evaluate Lexicase selection's performance on a two-task quadrupedal locomotion problem. The different objective sampling strategies all evolve effective behaviors across the two tasks, with a slight loss in performance on wall crossing versus individuals only evolved for wall crossing. Minimal performance differences arise across sampling strategies suggesting that a naive uniform sampling approach is acceptable in this domain. Choosing a naive sampling approach also has the added benefit of removing additional parameters from the search algorithm, reducing the need for user specified constraints.

In future work, we plan to expand the number of tasks from two up to a range of five to ten, substantially increasing the objective space. Additional tasks will also introduce conflicting pressures requiring compromises in performance between objectives. We will continue to examine performance, and characteristics of Lexicase selection in these broader spaces. Furthermore, although not necessarily the goal of this paper, quantifying the filtering efficacy of the objectives suggests that an *adaptive* Lexicase selection could monitor what objectives are stronger filters and up or down-weight those objectives. Changing the occurrence rate of those objectives would alter the explore versus exploit dynamics of the algorithm potentially increasing performance or leading to increased diversity.

Acknowledgements

The authors would like to thank William La Cava, Thomas Helmuth, Edward Pantridge, Emily Dolson, and Lee Spector for their conversations about various aspects of Lexicase selection. We would also like to acknowledge Keele University and Grand Valley State University for supporting this research.

References

Baydin, A. G. (2012). Evolution of central pattern generators for the control of a five-link planar bipedal walking mechanism. *Paladyn. Journal of Behavioral Robotics*, 3(1):45–53.

Clune, J., Beckmann, B. E., Ofria, C., and Pennock, R. T. (2009). Evolving coordinated quadruped gaits with the HyperNEAT generative encoding. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 2764–2771, Trondheim, Norway. IEEE.

Cully, A., Clune, J., Tarapore, D., and Mouret, J.-B. (2015). Robots that can adapt like animals. *Nature*, 521(7553):503–507.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182 –197.

Doncieux, S. and Mouret, J. B. (2013). Behavioral diversity with multiple behavioral distances. In *Proceedings of the 2013 IEEE Congress on Evolutionary Computation*, pages 1427–1434, Cancun, Mexico. IEEE.

Helmuth, T. and Spector, L. (2020). Explaining and exploiting the advantages of down-sampled lexicase selection.

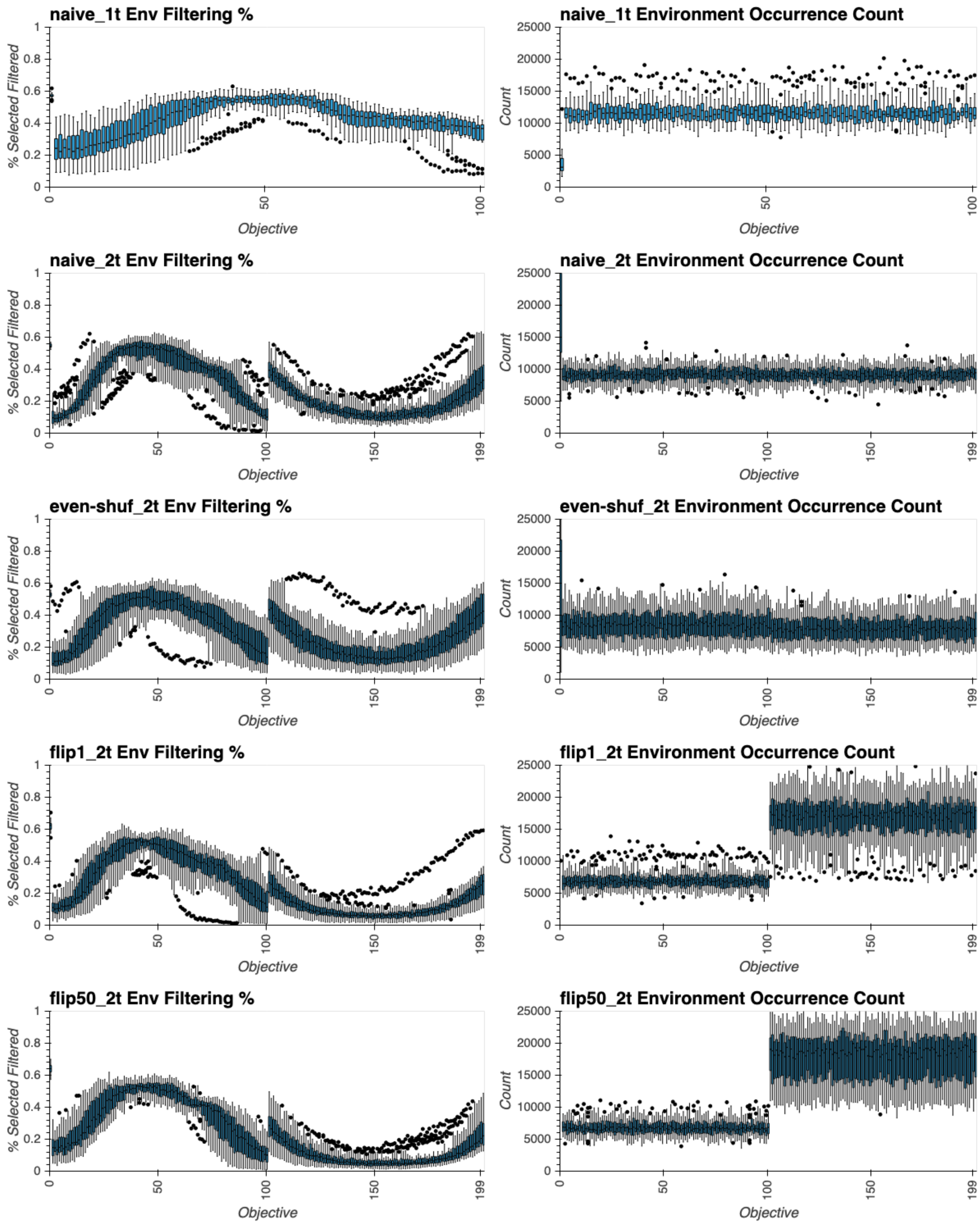

Figure 10: (Left) Each objectives typical filtering percentage when used for selection across replicates. (Right) Occurrence of each environment in the selection process across replicates.

In *The 2020 Conference on Artificial Life: A Hybrid of the European Conference on Artificial Life (ECAL) and the International Conference on the Synthesis and Simulation of Living Systems (ALIFE)*, number 32, pages 341–349, Montreal, Quebec, Canada.

Hernandez, J. G., Lalejini, A., Dolson, E., and Ofria, C. (2019). Random subsampling improves performance in lexicase selection. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 2028–2031, New York, NY, USA. Association for Computing Machinery.

Huizinga, J. and Clune, J. (2019). Evolving multimodal robot behavior via many stepping stones with the combinatorial multi-objective evolutionary algorithm.

Koos, S., Mouret, J. B., and Doncieux, S. (2010). Crossing the reality gap in evolutionary robotics by promoting transferable controllers. In *Proceedings of the 2010 ACM Genetic and Evolutionary Computation Conference*, pages 119–126, Portland, Oregon, USA. ACM.

Kriegman, S., Walker, S., Shah, D. S., Kramer-Bottiglio, R., and Bongard, J. (2019). Automated shapeshifting for function recovery in damaged robots. In Antonio Bicchi, H. K.-G. and Hutchinson, S., editors, *In Proceedings of the Robotics: Science and Systems XV*, Freiburg im Breisgau, Germany.

La Cava, W., Spector, L., and Danai, K. (2016). Epsilon-lexicase selection for regression. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016*, pages 741–748, Denver, Colorado, USA. ACM.

Lehman, J., Risi, S., D'Ambrosio, D., and Stanley, K. O. (2013). Encouraging reactivity to create robust machines. *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems*, 21(6):484–500.

Moore, J. M. and McKinley, P. K. (2016). *A Comparison of Multiobjective Algorithms in Evolving Quadrupedal Gaits*, pages 157–169. Springer International Publishing, Aberystwyth, UK.

Moore, J. M. and Stanton, A. (2017). Lexicase selection outperforms previous strategies for incremental evolution of virtual creature controllers. In *Proceedings of the 14th European Conference on Artificial Life*, pages 290–297, Lyon, France. MIT Press.

Moore, J. M. and Stanton, A. (2018). Tiebreaks and diversity: Isolating effects in lexicase selection. In *Proceedings of the 16th International Conference on the Simulation and Synthesis of Living Systems*, pages 590–597, Tokyo, Japan. ACM.

Moore, J. M. and Stanton, A. (2019). The limits of lexicase selection in an evolutionary robotics task. In *The 2019 Conference on Artificial Life: A Hybrid of the European Conference on Artificial Life (ECAL) and the International Conference on the Synthesis and Simulation of Living Systems (ALIFE)*, pages 551–558, Newcastle-Upon-Tyne, UK. MIT Press, MIT Press.

Moore, J. M. and Stanton, A. (2020). When specialists transition to generalists: Evolutionary pressure in lexicase selection. *Artificial Life Conference Proceedings*, (32):719–726.

Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics: The Biology, Intelligence and Technology of Self-Organizing Machines*. The MIT Press.

Pinville, T., Koos, S., Mouret, J.-B., and Doncieux, S. (2011). How to promote generalisation in evolutionary robotics: The progab approach. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, pages 259–266, Dublin, Ireland. ACM.

Ruud, E. L., Samuelsen, E., and Glette, K. (2016). Memetic robot control evolution and adaption to reality. In *Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–7, Athens, Greece. IEEE.

Spector, L. (2012). Assessment of problem modality by differential performance of Lexicase selection in genetic programming: A preliminary report. In *Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation*, pages 401–408, Philadelphia, Pennsylvania, USA. ACM.

Stanton, A. (2018). Stochastic ontogenesis in evolutionary robotics. In *Proceedings, ALife'18*, pages 214–221. MIT Press.

Stanton, A. and Channon, A. (2013). Heterogeneous complexification strategies robustly outperform homogeneous strategies for incremental evolution. In *Proceedings of the 12th European Conference on Artificial Life*, pages 973–980, Taormina, Italy.