

How Evolvable is Novelty Search?

David Shorten

Department of Computer Science
University of Cape Town
Cape Town, South Africa
Email: dshorten@cs.uct.ac.za

Geoff Nitschke

Department of Computer Science
University of Cape Town
Cape Town, South Africa
Email: gnitschke@cs.uct.ac.za

Abstract—This research compares the efficacy of novelty versus objective based search for producing evolvable populations in the maze solving task. Populations of maze solving simulated robot controllers were evolved to solve a variety of different, relatively easy, mazes. This evolution took place using either novelty or objective-based search. Once a solution was found, the simulation environment was changed to one of a variety of more complex mazes. Here the population was evolved to find a solution to the new maze, once again with either novelty or objective based search. It was found that, regardless of whether the search in the second maze was directed by novelty or fitness, populations that had been evolved under a fitness paradigm in the first maze were more likely to find a solution to the second. These results suggest that populations of controllers adapted under novelty search are less evolvable compared to objective based search in the maze solving task.

I. INTRODUCTION

Novelty search (NS) [1] is an adaptive search based approach, that replaces the traditional fitness function [2] of *Evolutionary Algorithms* (EAs). That is, NS replaces the notion of fitness, which rewards solutions based on how close they are to an objective, with one which rewards solutions based on their behavioral novelty. This approach was initially tested in a robot maze navigation task [3], where it evolved successful behaviors using fewer evaluations than a fitness based approach.

Subsequently, further research has been conducted using NS via applying it to *Artificial Neural Network* (ANN) controllers in simulated evolutionary robotics tasks. It was found that in the biped locomotion task domain that NS was able to outperform evolving gaits using objective based search [1]. However, it has been found by other authors that in other task domains the most effective approach is to use a combination of novelty and fitness incentives [4], [5].

For either natural or digital organisms, one can question the future potential and benefits of them deriving adaptable, robust, novel, or complex behaviors. This loose collection of concepts can be thought of as *evolvability*. Lehman and Stanley [6] studied this idea within the context of NS and investigated whether explicitly rewarding novelty would improve the evolvability of digital organisms in maze navigation and bipedal gait robot control tasks. In these experiments they were using the phenotypic diversity of offspring as the measure of evolvability. They found that organisms evolved under NS produced offspring with greater phenotypic diversity, so long as the task was not *fragile*. In these fragile tasks they found that self-adaptation of mutation parameters increased evolvability

under novelty-based search. Subsequently, Lehman and Stanley examined the emergence of evolvability, under the same definition, in abstract genetic drift models [7] and concluded that evolvability emerges in the absence of adaptive pressure.

These results are to be expected, given that if organisms are rewarded purely based on their phenotypic novelty then lineages which produce novelty will become dominant. However, this emphasis on the production of phenotypically varied offspring must be carefully evaluated. The conditions supporting this definition of evolvability are significant in the context of EC, given that an organism's ability to produce a phenotypically diverse range of offspring is not beneficial if all offspring are not suited to solve their given task.

Kirschner and Gerhardt [8], proponents of the *evolvability is variability* paradigm in biology, place emphasis on the fact that an important component of this type of evolvability is a reduction in potentially deleterious mutations. A problem in using variability as a definition for evolvability in EAs, is that the designers of experiments play a key role in defining detrimental mutations. For example, should a mutation which causes a maze navigating robot to scurry off to a corner and hide there be considered a deleterious or novel variation? In their seminal paper, Wagner and Altenberg [9], also proponents of the study of variability, go even further in emphasizing the importance of the nature of variation in offspring.

“Evolvability is the genome's ability to produce adaptive variants when acted upon by the genetic system... The situation is analogous to obtaining a verse of Shakespeare from monkeys banging away on typewriters. Typewriters make this far more likely than if the monkeys had pencil and paper. The typewriters at least constrain them to produce strings of letters. Similarly, the genotype-phenotype map constrains the directions of phenotypic change resulting from genetic variation.”

Much research in biology has centered around how the genotype-phenotype map is able to facilitate the emergence of adaptive phenotypes. Notable themes in this research are robustness, the ability of the phenotype to remain unchanged despite changes to the genotype [10], as well as modularity, or the independence of phenotypic traits [11].

There exists a multitude of definitions of evolvability in biological literature. The reader is referred to Pigliucci [11] for an overview of evolvability in nature. Within EC, a number of different definitions and associated metrics have been proposed, including those that focus exclusively on the

fitness of offspring [12], [13], [14]. Tarapore and Mouret [15] developed a metric which incorporated both the fitness and diversity of offspring. One metric, which excludes discussions of fitness and novelty, is that of Reisinger and Miikkulainen [16], which measured evolvability as the ability of organisms to detect deeper patterns in a dynamic fitness function.

In this research, an organism’s evolvability is defined as the likelihood of its descendants solving sets of experimenter defined tasks. That is, evolvability is tantamount to an organism’s adaptability. Clearly, factors such as phenotypic diversity and fitness of offspring are correlated with such a measure. However, instead of imposing a set of features that will increase the likelihood of individual genotypes producing useful phenotypes (behaviors), this definition defers to an organism’s capability to adapt and thus survive in its given environment. It also has precedent within biological literature, although in nature it relates to whether descendants of organisms can adapt to changing environments or respond to selective pressure [11], [17]. Furthermore, it has been used as a metric for evolvability in simulations of gene regulatory networks [18].

This research tests NS using this definition of evolvability by evolving populations in one simulation environment, before transferring solutions to another environment and observing if a new solution can be found. Specifically, controller populations are evolved on a variety of mazes, using either objective or novelty based search. Once a maze solving controller has been found for a given maze, that controller is then placed in a different, more complex maze, and either objective (fitness) or novelty based search is once again used to adapt controller behavior. This process is similar to *incremental evolution* [19] and the transfer of solution representations to new learning environments in *transfer learning* [20]. In this context, however, it is used as an experimental technique rather than a learning mechanism. Such experimental techniques increases the relevance of the *evolvability as adaptability* metric, and elucidate which types of search are best at producing phenotypes able to adapt to changing environments.

Results of this study indicate that, regardless of whether future controller adaptation takes place using objective or novelty based search, populations evolved using a fitness function are more likely to find a solution to the maze to which they are transferred. This result was anticipated given that populations adapted under NS will contain a larger variety of behaviors, though not necessarily more beneficial behaviors. Although such novelty is useful in finding the stepping stones for certain types of highly deceptive tasks [1], for less deceptive tasks populations converging about global fitness peaks are more likely to produce useful offspring and thus task solutions.

II. METHODS

A. Novelty Search (NS)

It is widely accepted that EAs and more broadly, automated problem solving methods constructed within the purview of artificial intelligence, traditionally have an associated objective which algorithms are trying to attain [2]. Metrics are introduced for the purpose of describing how far a given solution is from an objective. However, the notion that natural evolution is not guided by objectives [21], has lead to the synthesis of EAs which are motivated not by how far the solutions are

from a desired objective, but rather how *novel* they are. This NS approach can, in principle, replace the fitness function of any EA, or even act as the guiding adaptive mechanism in other types of search methods.

A criticism of NS is that it is merely a random search [22]. However, given that simple behaviors emerge early in the search process, then these behaviors will be used as *stepping stones* for the discovery of more complex and beneficial behaviors. Ultimately, the desirable outcome is for the NS process to discover a behavior that is an optimal or near optimal solution. The theoretical advantage of NS is that it rewards a greater variety of behaviors than objective based search. This aids in the emergence of the stepping stones which might be necessary to achieve the desired solution [1].

Any implementation of NS requires a representation of behavioral solutions and an associated metric which assigns these behaviors a novelty score based on previously observed behaviors. The originally proposed and much used metric is that of *sparseness*, shown in equation 1 [1].

$$\rho(x) = \frac{1}{k} \sum_{i=0}^k \text{dist}(\mu_i) \quad (1)$$

Here the μ_i are the k nearest neighbors of x and *dist* is a distance measure. A common and intuitive way of representing a solution’s behavior is to use a vector of numbers. For example, in the maze navigation domain this vector could be the position of the robot at certain sampled points in time. If a vector of numbers is used then the distance metric is either the Euclidean distance between vectors or the average difference between their components. Currently, all research in the maze solving domain uses the average distance between components [1], [22]. Additionally, NS requires the use of an archive of previously observed behaviors, so that for every individual tested, there is a record of the area of the search space which it explored.

B. Neuro-Evolution of Augmenting Topologies (NEAT)

The underlying EA used in these experiments is *Neuro-Evolution of Augmenting Topologies* (NEAT) [23]. This method has been demonstrated as a powerful problem solver in a broad range of tasks [24], [23], [25], [26], and is typically used as the underlying adaptive method when NS is applied to solve evolutionary robotics tasks. For example, NEAT and NS have been applied to the maze solving domain [22], [1], and NEAT was also the method used in previous research on NS and evolvability [6].

NEAT evolves both the topology and the weights of ANNs, using the evolution of topology to increase controller performance, and as such falls into the category of *Topology and Weight Evolving Artificial Neural Networks* (TWEANNs) [27]. NEAT uses a complexification process, where evolutionary search begins with a population of simple ANN controllers which, over successive generations, become as topologically complex as they need to be to solve a given task.

The complexification process ensures that only topological innovations which are useful are preserved so as the dimensionality of the search space is minimized. Given that new topological mutations will normally be detrimental to the

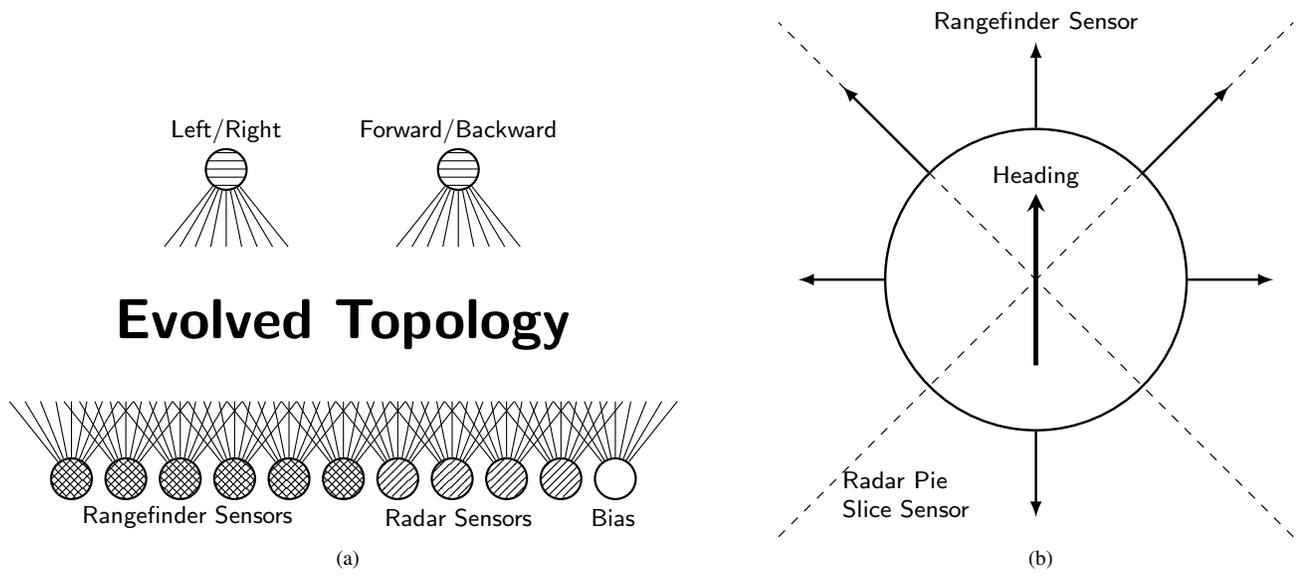


Fig. 1: (a) ANN controller used in the experiments. (b) Sensory configuration of robots in the simulated maze task. Both figures adapted from [1]

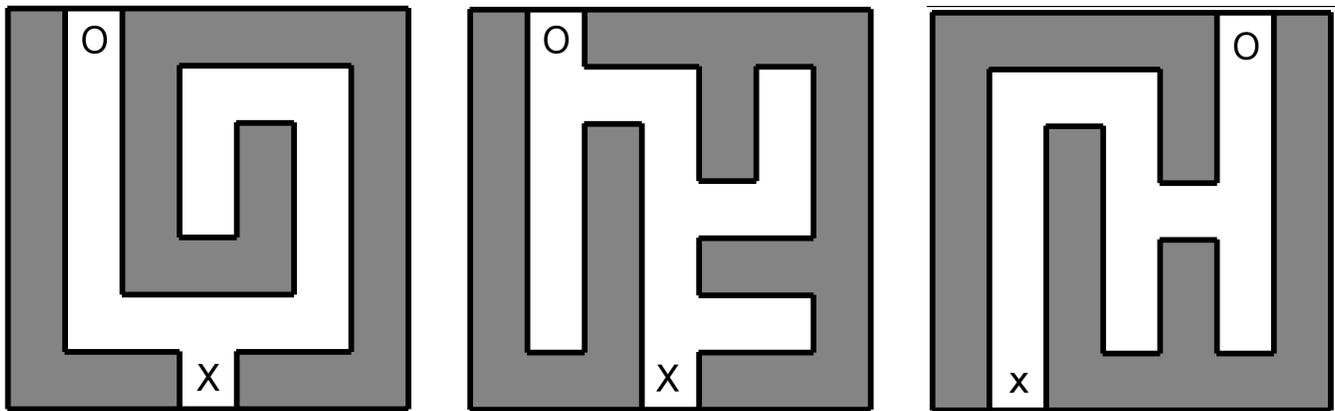


Fig. 2: Three sample mazes from set A. The navigator starts at the point marked X and its goal is to reach the point marked O

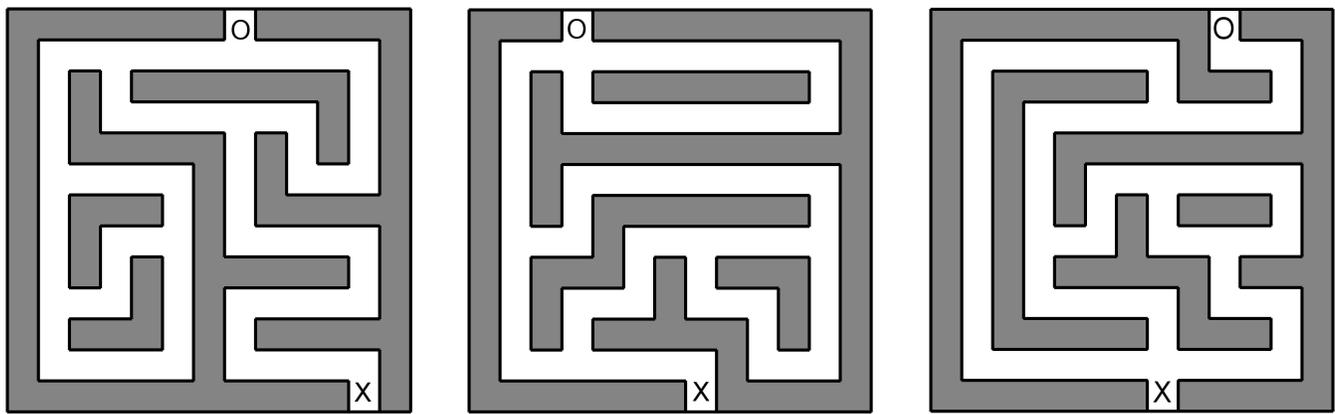


Fig. 3: Three sample mazes from set B. The navigator starts at the point marked X and its goal is to reach the point marked O

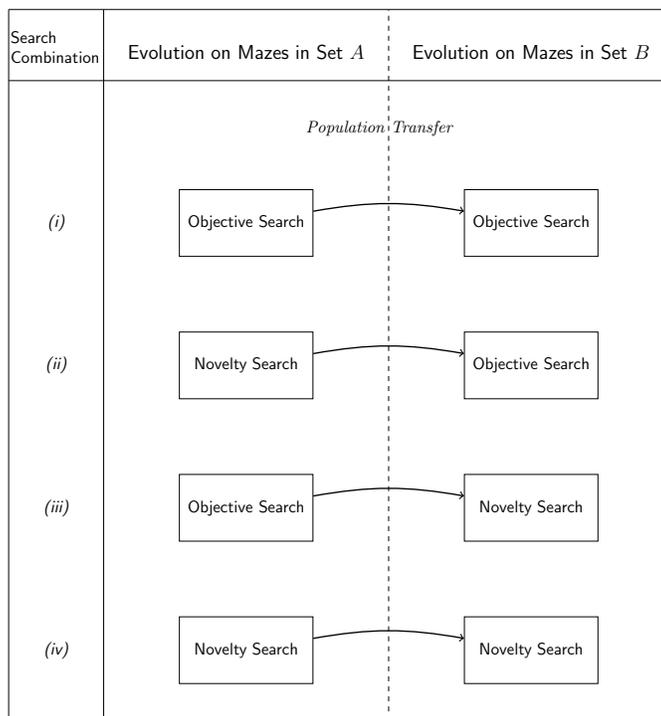


Fig. 4: The combinations of search strategies used in the experiments.

fitness of evolving ANNs, novel topologies are protected in niches, where they only compete with similarly novel controller topologies. This innovation protection mechanism gives novel ANN controllers time to optimize within their niche and thus become competitive with ANN controller topologies in other niches, thus improving overall solution quality.

Furthermore, NEAT partially solves the competing conventions problem in neuro-evolution [28], where ANNs which are structurally diverse can be functionally identical, complicating the development of effective crossover operators. To circumvent this, NEAT uses historical markings on genes so that crossover between ANNs can occur on the parts of their genotype encoding which are derived from common ancestors.

C. Maze Navigation Controllers

As in related research [1], the ANN controllers of simulated maze navigating robots are adapted using either objective or novelty based search applied to NEAT. The inputs to the ANN controller consist of six rangefinder sensors as well as four radar sensors. The rangefinder sensors indicate the distance to the closest obstacle in their direction. The radar sensors each correspond to a sensory quadrant which covers a quarter of the space radiating outwards from the robot. Each sensor indicates whether the goal falls within its quadrant. This sensory setup is displayed in figure 1 and further detailed elsewhere [1].

III. EXPERIMENTS

A. Maze Set Construction

An important goal in the design of experiments was to choose two sets of mazes, *A* and *B* such that, once a population had found a solution to a maze in set *A*, it would not contain

a solution to the maze in set *B* which it was transferred to. This is because, already having a solution to the target maze is a sign of population variability and capability to generalize, not evolvability.

The mazes used in these experiments were generated using the *Daedalus* software of Walter Pullen [30]. The generation algorithms used by this program operate on a grid. All cells are initially impassable. Cells are incrementally cleared to become passable. The boundaries between cleared and uncleared cells then become the walls of the maze. Two types of mazes were generated. Mazes in set *A* were *perfect*. In these mazes every cell has a unique path to every other cell. This implies that the maze has no cycles and also that there is a unique path through the maze itself [31]. Mazes in set *B* were *braid* mazes. This class of maze can contain cycles but, unlike perfect mazes, they contain no dead ends [31]. Figures 2 and 3 show examples from these maze sets.

The reason for using these two different classes of mazes is that they require different strategies. A perfect maze can be solved easily using the *left-hand rule* [31]. In this strategy a human navigator would merely place their left hand on a wall and keep moving forward without removing this hand. This strategy will not work in a maze with cycles, which some braid mazes have. Using sets of mazes which require different solution strategies decreases the likelihood that once the population is transferred it will contain a solution to the new environment. In keeping with this trend, mazes in set *B* were made to be harder than those in set *A* by making them larger. Mazes in set *A* were constructed over a 7x7 grid, whereas a 13x13 grid was used for those in set *B*. See table I for a summary of the maze specifications.

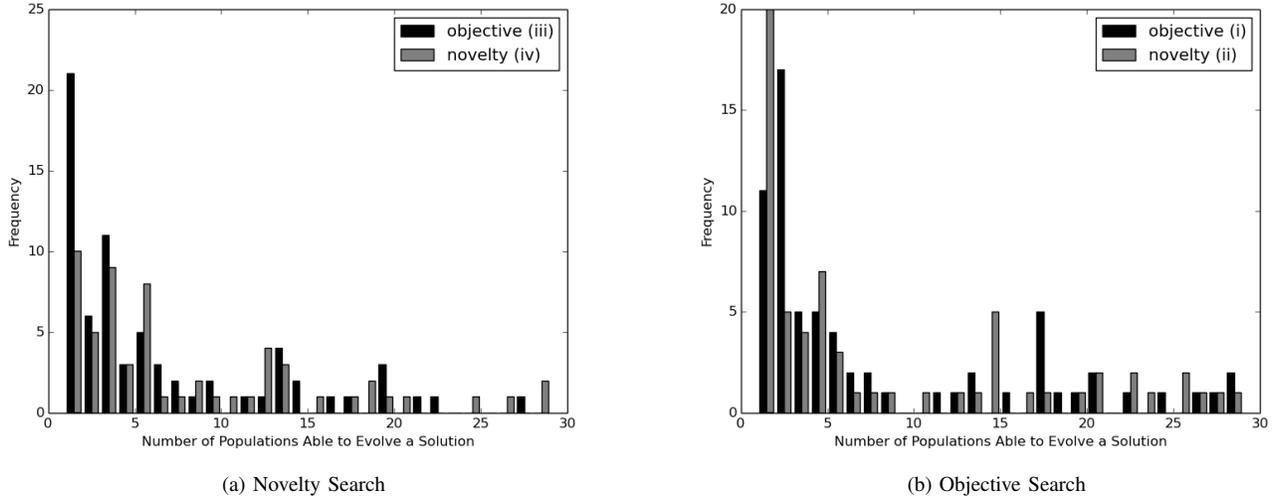


Fig. 5: The distribution of the number of populations which produced solutions for each maze in set B . To improve the scale, the incidence of zero solving populations is not displayed. Figure (a) displays runs where novelty search was used in the second maze and (b) displays runs where fitness search was used. The black bars represent objective based search in the preceding mazes and the grey bars represent novelty search. Note that the highest number of solution producing populations that any maze in maze set B could have was 35. This was the number of mazes in set A and thus the number of populations applied to solve each maze in set B for a given and preceding search method.

TABLE I: Parameters describing the two maze sets

Set Name	Number of Mazes	Maze Type	Grid Dimensions
A	35	Perfect	7x7
B	200	Braid	13x13

TABLE II: Average number of populations which were able to solve each maze in set B after 160 generations for populations previously evolved using novelty or objective based search. Standard deviations are in parentheses. All differences in averages are statistically significant ($p < 0.1$, Wilcoxon signed-rank test [29]), except between (ii) and (iii) and between (ii) and (iv)

Combination	Preceding Search Method (Maze Set A)	Utilized Search Method (Maze Set B)	Average Number of Populations (Maze Solving Behaviors)
(i)	Objective	Objective	3.55 (7.14)
(ii)	Novelty	Objective	3.28 (6.93)
(iii)	Objective	Novelty	3.19 (6.45)
(iv)	Novelty	Novelty	3.02 (6.25)

B. Experimental Runs

As in experiments of related research [6] these experiments applied either objective or novelty based NEAT to adapt the ANN controllers of maze navigating robots. Other than the population size and the number of simulation iterations, all parameters were identical to those used in related work [6] and included in the *novelty search C++* package¹. The population size was set to 25 in order to reduce the probability of an evolved population containing a solution to the second maze. The number of iterations per maze navigating simulation was set to 600, since sufficient navigation time for some of the

more difficult mazes was required. The novelty metric used was the ending position of the robot after a simulation, where novelty was calculated as the average difference between the x and y coordinates of two positions.

In the experiments, four combinations of search types were tested (figure 4). In combination (i), the adaptation of maze solving behaviors in both sets A and B occurred using a fitness metric. In combination (ii), adaptation in set A occurred using a novelty metric, whereas in set B a fitness metric was used. Combination (iii) differed by using a fitness metric in set A , but used a novelty metric in set B . Combination (iv) used a novelty metric in sets A and B .

¹Available from <http://eplex.cs.ucf.edu/noveltysearch/userspage/>

The purpose of these combinations was to ascertain the impact of novelty or objective based search on the evolvability of populations. Also, four combinations allowed determination of whether, in the maze navigation task, evolvability was a general mechanism, or something specific to the type of objective used by a succeeding search process.

For each of the four search process combinations, and every pair of mazes (a, b) , where a is a maze from A and b is a maze from B , populations were randomly initialized and adapted to solve maze a . Once a solution had been found to maze a , the maze was changed and the evolved population used as a starting point for the adaptation of maze solving behaviors for maze b . There were 35 mazes in maze set A and 200 mazes in maze set B (table I). Thus, there were 7000 pairs of mazes (a, b) and so, over the four combinations, a total of 28 000 population transfers took place. Moreover, for each of the four combinations, each maze in set B received 35 populations. (equivalent to 35 runs for each combination).

For the evolution of behaviors on maze a , the search process was restarted after 4000 new genotypes had been generated, where each new genotype always replaced the least fit genotype. Thus, given a population size of 25, many generations passed before the search process was restarted. This was done since preliminary testing indicated that if a solution had not been found after the production of 4000 new genotypes, it was likely that evolution was stuck in a sub-optimal part of the search space and would take prohibitively long to find a solution. On maze b , objective or novelty based search were run for 160 generations, and whether a maze b solving behavior was found or not, was recorded.

Despite the use of methods described in section III-A, constructing maze sets *a priori* such that solutions evolved for set A never contained solutions for set B was found to be non-trivial. To address this, if for any run, a transferred population contained a solution to a maze b , then all runs on this maze were ignored. In total, 37 of the 200 mazes in set B were removed this way.

IV. RESULTS AND DISCUSSION

Results indicated that, on average, for any given maze b in maze set B , more populations adapted with objective based search on maze set A evolved solutions to maze b compared to populations adapted using NS on set A . This held regardless of whether maze b solving behaviors were adapted with objective or novelty based search. The difference between such maze set B solving populations was found to be statistically significant ($p < 0.1$, Wilcoxon signed-rank test [29]). These results are summarized in table II and figure 5.

One may note that in table II the standard deviations are relatively high compared to the average values. This results from the disparity in problem solving difficulty for different mazes. Hence there is an equal level of disparity between the number of populations that can and cannot evolve maze solving behaviors. Thus, pairwise statistical tests were applied to compare the number of times objective versus novelty based search solves a given maze in set B , where the population had previously been evolved for mazes in set A , using objective or novelty based search.

Figure 5 illustrates a distribution of the number of populations that evolved maze solving behaviors for each maze in set B . Figure 5 (a), left hand side, displays the results of the 35 runs, given either novelty or objective based search as the preceding method, where NS was applied to maze set B . Here the black bars represent objective search in set A (combination (iii)) and the grey bars represent NS in set A (combination (iv)). Figure 5 (b), right hand side, displays the results of the 35 runs, given either novelty or objective based search as the preceding search method, where objective based search was applied to maze set B . Here the black bars represent objective search in set A (combination (i)) and the grey bars represent NS in set A (combination (ii)). The vertical *frequency* axis of figure 5 indicates the number of mazes in set B for which a given number of populations (horizontal axis) was able to evolve a solution.

So, for example, in figure 5 (b), the right most black and grey bars (representing objective and novelty based search, respectively), indicate that only one maze in set B had 29 populations produce a solution, where these populations were previously adapted with NS in maze set A . Also, only two mazes in set B had 29 populations produce a solution, and these populations were previously adapted with objective based search in maze set A . However, in the left-most side of figure 5 (b), the black and grey bars indicate that 20 mazes in set B had only one population produce a solution, where these populations were previously adapted with NS in maze set A . Also, 11 mazes in set B had only one population produce a solution, where these populations were previously adapted with objective based search in maze set A .

The purpose of this research was to observe, for a given successive search method, whether using novelty or objective search in the first maze set would affect the task performance of a successively applied search method. Resultant data from these experiments allows one to ascertain, for a fixed previous search method, whether novelty or objective search is more effective in the given maze solving domain. Given that novelty search was used as the preceding search method, there was not a statistically significant difference in task performance between objective and novelty-based search on maze set B . However, given that objective search was used as the preceding strategy, it was found that objective search was able to produce more maze-solving behaviors compared to NS, with statistical significance, in maze set B .

This may seem to challenge previous related research results which indicated that NS derived maze navigation behaviors yielding higher task performance in the maze domain compared to objective based search [1]. However, the mazes used here (figures 2 and 3) were less deceptive than those used in the related research [1]. Also, the problem of evolving transferred populations is substantially different to the approach taken in previous research [1], which was to evolve randomly initialized populations.

Results also indicated that many populations performed poorly when transferred from maze set A to B . That is, for each of the combinations of preceding and subsequent search methods (figure II), all 35 populations transferred to a given maze in set B failed to find maze solving behaviors within the given generation limit, for an average of 97 mazes. This was found to be a result of maze construction, given that

mazes in set B were chosen to be very different from those in set A . Set B mazes were constructed to be larger and to contain fundamentally different structural features. Preliminary testing showed that this reduced the incidence of transferred populations already containing solutions to mazes in set B .

This result also indicates the importance of the tasks between which evolved populations are transferred being sufficiently similar, lest poor results be yielded after an evolved population is transferred. As elucidated in related research [32], [33], if the nature of solution search space is sufficiently different between tasks then transferred populations will most likely fail to evolve useful behaviors unless special genotype encoding [32] or shaping approaches [33] are used.

The results presented in figure 5 and table II do not refute NS as a valuable complement to current evolutionary adaptation approaches and its potential as a controller (behavioral) design method. Rather these results suggest that for certain types of tasks and environments, such as the mazes demonstrated in section III, NS produces populations of solutions that are not as evolvable as objective based search. That is, removing objectives from an evolutionary search process allows for the development of potentially useful stepping stones to desirable behaviors. However, such non-objective search processes operate at the cost of creating incentives for the development of undesirable behaviors, which may be unnecessarily propagated according to task and environment constraints.

Furthermore, the definition of evolvability in this research was tantamount to behavioral adaptability, which differed from the definition used by related research [6]. Ascertaining the computational mechanisms that consistently produce evolvable populations is essential for the progress of neuro-evolution [7], and more generally EC research [9]. As such, this research contributes to the open research question that aims to address the full scope of NS limitations, including the evolvability of populations produced by NS for a broad range of tasks. Future work will evaluate other evolvability metrics suggested for EAs [12], [13], [14], [15], [16] in the context of maze navigation as well as other tasks.

V. CONCLUSION

This research investigated the *evolvability* of objective versus novelty based methods for adapting ANN controllers of simulated robots that must successfully navigate sets of mazes. In this study, evolvability was equated with the adaptability of controller populations from which maze navigating behaviors were evolved. Evolvability was measured as the number of mazes solved on a large set of complex mazes, once solutions had been evolved for a relatively small set of simple mazes. Results indicated that when objective based search was applied to solve the first small simple maze set, and the population was transferred to the larger more complex maze set, then a significantly higher number of mazes could be solved, compared to the case where NS was applied to solve the first maze set. These results suggest that for certain types of tasks, NS produces less evolvable populations compared to objective based search. However, the evolutionary, task, and environmental conditions necessary for novelty versus objective based search to consistently produce beneficial phenotypes (problem

solving behaviors) over evolutionary time is the subject of ongoing research.

ACKNOWLEDGMENT

The financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the authors and are not necessarily to be attributed to the NRF. Computations were performed using facilities provided by the University of Cape Town's ICTS High Performance Computing team: <http://hpc.uct.ac.za>

REFERENCES

- [1] J. Lehman and K. Stanley, "Abandoning objectives: Evolution through the search for novelty alone," *Evolutionary Computation*, vol. 19, no. 2, pp. 189–223, 2011.
- [2] A. Eiben and J. Smith, *Introduction to evolutionary computing*. Springer, 2003.
- [3] J. Lehman and K. Stanley, "Exploiting open-endedness to solve problems through the search for novelty," in *ALIFE*, 2008, pp. 329–336.
- [4] G. Cuccu and F. Gomez, "When novelty is not enough," *Applications of Evolutionary Computation*, pp. 234–243, 2011.
- [5] J.-B. Mouret, "Novelty-based multiobjectivization," in *New Horizons in Evolutionary Robotics*. Springer, 2011, pp. 139–154.
- [6] J. Lehman and K. Stanley, "Improving evolvability through novelty search and self-adaptation," in *IEEE Congress on Evolutionary Computation*. IEEE, 2011, pp. 2693–2700.
- [7] —, "Evolvability is inevitable: Increasing evolvability without the pressure to adapt," *PloS one*, vol. 8, no. 4, p. e62186, 2013.
- [8] M. Kirschner and J. Gerhart, "Evolvability," *Proceedings of the National Academy of Sciences*, vol. 95, no. 15, pp. 8420–8427, 1998.
- [9] G. Wagner and L. Altenberg, "Perspective: Complex adaptations and the evolution of evolvability," *Evolution*, pp. 967–976, 1996.
- [10] A. Wagner, "Robustness and evolvability: a paradox resolved," *Proceedings of the Royal Society B: Biological Sciences*, vol. 275, no. 1630, pp. 91–100, 2008.
- [11] M. Pigliucci, "Is evolvability evolvable?" *Nature Reviews Genetics*, vol. 9, no. 1, pp. 75–82, 2008.
- [12] G. Hornby, H. Lipson, and J. Pollack, "Generative representations for the automated design of modular physical robots," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 4, pp. 703–719, 2003.
- [13] J. Grefenstette, "Evolvability in dynamic fitness landscapes: A genetic algorithm approach," in *Proceedings of the Congress on Evolutionary Computation*, vol. 3. IEEE, 1999.
- [14] J. Reisinger and R. Miikkulainen, "Acquiring evolvability through adaptive representations," in *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*. ACM, 2007, pp. 1045–1052.
- [15] D. Tarapore and J.-B. Mouret, "Comparing the evolvability of generative encoding schemes," in *Proceedings of ALife*. MIT Press, pp. 1–8.
- [16] J. Reisinger, K. Stanley, and R. Miikkulainen, "Towards an empirical measure of evolvability," in *Proceedings of the 2005 workshops on Genetic and evolutionary computation*. ACM, 2005, pp. 257–264.
- [17] T. Flatt, "The evolutionary genetics of canalization," *The quarterly review of biology*, vol. 80, no. 3, pp. 287–316, 2005.
- [18] A. Quayle and S. Bullock, "Modelling the evolution of genetic regulatory networks," *Journal of theoretical biology*, vol. 238, no. 4, pp. 737–753, 2006.
- [19] F. Gomez and R. Miikkulainen, "Incremental evolution of complex general behavior," *Adaptive Behavior*, vol. 5, no. 3–4, pp. 317–342, 1997.
- [20] M. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey," *Journal of Machine Learning Research*, vol. 10, pp. 1633–1685, 2009.

- [21] M. Lynch, "The frailty of adaptive hypotheses for the origins of organismal complexity," *Proceedings of the National Academy of Sciences*, vol. 104, no. Suppl 1, pp. 8597–8604, 2007.
- [22] R. Velez and J. Clune, "Novelty search creates robots with general skills for exploration," in *Proceedings of Genetic and Evolutionary Computation Conference*. Vancouver, Canada: ACM Press, 2014, p. To appear.
- [23] K. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary Computation*, vol. 10, no. 2, pp. 99–127, 2002.
- [24] P. Allen and P. Faloutsos, "Complex networks of simple neurons for bipedal locomotion," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE Press, 2009, pp. 4457–4462.
- [25] K. Stanley and R. Miikkulainen, "Evolving a roving eye for go," in *Proceedings of Genetic and Evolutionary Computation Conference*. ACM Press, 2004, pp. 1226–1238.
- [26] K. Stanley, B. Bryant, and R. Miikkulainen, "Real-time neuroevolution in the nero video game," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 6, pp. 653–668, 2005.
- [27] X. Yao, "Evolutionary artificial neural networks," *Journal of Neural Systems*, vol. 4, no. 3, pp. 203–222, 1993.
- [28] D. Floreano, P. Dürri, and C. Mattiussi, "Neuroevolution: from architectures to learning," *Evolutionary Intelligence*, vol. 1, no. 1, pp. 47–62, 2008.
- [29] B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes*. Cambridge, UK: Cambridge University Press, 1986.
- [30] W. Pullen. (2014) Think labyrinth: Daedalus. [Online]. Available: <http://www.astrolog.org/labyrinth/daedalus.htm>
- [31] J. Xu and C. Kaplan, "Image-guided maze construction," in *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3. ACM, 2007, p. 29.
- [32] P. Verbancsics and K. Stanley, "Evolving static representations for task transfer," *Journal of Machine Learning Research*, vol. 11, no. 1, pp. 1737–1769, 2010.
- [33] S. Doncieux, "Transfer learning for direct policy search: A reward shaping approach," in *Proceedings of ICDL-EpiRob conference*. Springer, 2004, pp. 1–6.