# Critical Factors in the Performance of Novelty Search

Steijn Kistemaker
Informatics Institute
University of Amsterdam
Amsterdam, The Netherlands
steijn.kistemaker@student.uva.nl

Shimon Whiteson
Informatics Institute
University of Amsterdam
Amsterdam, The Netherlands
s.a.whiteson@uva.nl

## ABSTRACT

*Novelty search* is a recently proposed method for evolutionary computation designed to avoid the problem of deception, in which the fitness function guides the search process away from global optima. Novelty search replaces fitness-based selection with novelty-based selection, where novelty is measured by comparing an individual's *behavior* to that of the current population and an archive of past novel individuals. Though there is substantial evidence that novelty search can overcome the problem of deception, the critical factors in its performance remain poorly understood. This paper helps to bridge this gap by analyzing how the *behavior function*, which maps each genotype to a behavior, affects performance. We propose the notion of *descendant fitness probability* (DFP), which describes how likely a genotype's descendants are to have a certain fitness, and formulate two hypotheses about when changes to the behavior function will improve novelty search's performance, based on the effect of those changes on behavior and DFP. Experiments in both artificial and deceptive maze domains provide substantial empirical support for these hypotheses.

## Categories and Subject Descriptors

I.2.8 [**Problem Solving, Control Methods, and Search**]

## General Terms

Algorithms, Measurement, Performance, Experimentation

## Keywords

Evolutionary computation, problem of deception, novelty search, neural networks, neuroevolution

## 1. INTRODUCTION

One of the biggest challenges for evolutionary computation, and stochastic optimization in general, is how to tackle the *problem of deception*, in which the fitness function guides the search process towards local optima and away from global ones. According to Whitley, "the only challenging optimization tasks are problems involving some degree of deception" [21]. Numerous techniques have been proposed to overcome this problem. In fitness shaping [22, 3, 20, 19], the fitness function is altered so as to reward intermediate solutions on the path to a global optimum. However, it is only feasible if strong prior knowledge is available. In contrast, diversity maintenance techniques [1, 17, 14, 2, 11, 5], which aim to prevent premature convergence by maintaining heterogeneous populations, require less prior knowledge. However, they only mitigate the problem of deception and can easily fail in highly deceptive tasks.

Recently, Lehman and Stanley proposed *novelty search* [6], a radically different approach inspired by new insights about the role of non-adaptive processes in complexification in natural evolution [10, 12]. Essentially, it replaces fitness-based selection with novelty-based selection and in so doing bypasses the problem of deception. Novelty is measured by comparing an individual's *behavior* to that of the current population and an archive of past novel individuals. Behaviors typically describe some aspect of the individual's phenotype, e.g., how a robot moves around a maze.

To make novelty search feasible, a task-specific *behavior function* must be chosen to map each genotype to a behavior. Novelty is thus measured in phenotype space instead of genotype space. Choosing the behavior function requires prior knowledge about which behavior features affect fitness. However, unlike in fitness shaping, it is not necessary to understand *how* those features affect fitness, making novelty search more generally applicable.

Substantial evidence has accrued demonstrating that novelty search can successfully overcome deception and outperform fitness-based selection in deceptive problems [6, 16, 7, 8, 15]. However, there has been little work examining when novelty search works and why. In particular, not much is currently understood about how the behavior function influences performance and what properties of a behavior function are important for good performance.

The goal of this paper is to begin bridging this gap in our understanding of novelty search. To this end, we introduce the notion of *descendant fitness probability* (DFP), which describes how likely a genotype's descendants are to have a certain fitness. Intuitively, DFP is important because genotypes with similar DFPs will lead novelty search in similar directions and should thus have similar behaviors to prevent redundant exploration. Using DFP, we distinguish between four relationships between pairs of genotypes that depend on the equality or inequality of their behaviors and DFPs.


*GECCO'11,* July 12–16, 2011, Dublin, Ireland.

Next, we propose two specific hypotheses about the circumstances under which changes to the behavior function will reduce novelty search's evaluation costs, based on the effect of those changes on the relationships between genotype pairs. To test these hypotheses, we construct an artificial domain in which the effect of mutations on behavior is transparent, facilitating carefully controlled experiments. In addition, we consider several deceptive maze domains, including those used by Lehman and Stanley, that provide less controlled but more realistic tests of our hypotheses. Overall, our results in both the artificial and maze domains provide substantial empirical support for our hypotheses about the critical factors in the performance of novelty search, as well as raising intriguing questions for future research.

## 2. NOVELTY SEARCH

Stochastic optimization methods such as evolutionary computation typically use the fitness function as a guide for search, e.g., in each generation a population is constructed from the fittest members of the previous generation. Intuitively, this approach makes sense because good solutions are likely to be related to even better solutions. Ideally, the fitness function rewards intermediate solutions that light a path towards a global optimum. However, in many cases, this does not occur. Problems in which intermediate deleterious steps have to be made to reach the global optimum are called *deceptive* and pose a major challenge for optimization.

*Novelty search* [6] was recently proposed as a way to tackle deceptive problems. The authors, observing that the problem of deception is rooted in the fitness measure, propose the radical idea of simply ignoring fitness altogether. Instead of searching for fit individuals, it searches for novel ones. Unlike a random walk, novelty search has an explicit drive to continually discover novel behaviors.

Novelty is measured by comparing an individual's *behavior* to that of the current population and an archive of past novel individuals. Behaviors are represented using a *behavior characterization*, a set of $M$ features, which can be based on the genotype, the phenotype, or characteristics of the problem domain. The behavior of an individual $x$ is an assignment of values to these features and is determined using a *behavior function* $B : x \rightarrow \Re^M$. For simplicity, we assume that behavior features are real-valued. Typically, $B(x)$ is computed by applying the genotype to the task and extracting the behavior. Thus, in a stochastic domain, $B(x)$ can also be stochastic.

A *distance metric* is used to calculate the distance between two behaviors. Although the distance metric can be arbitrarily complex, we assume the following form:

$$\text{dist}(x, y) = \frac{1}{M} \sum_{i=1}^{M} \delta_i(B_i(x), B_i(y))$$

where $B_i(x)$ is the value of the $i$-th behavior feature for individual $x$. The $\delta_i$ function is a feature-specific distance metric, e.g., Euclidean, minimum edit, or Hamming distance.

Using the behavior function and distance metric, the *novelty metric* calculates the average distance to the $K$ nearest neighbors in behavior space:

$$\rho(x) = \frac{1}{K} \sum_{i=1}^{K} \text{dist}(x, \mu_i)$$

where $\mu_i$ is the $i$-th nearest neighbor in the population or the archive according to the distance metric. A high average distance corresponds to low density and high novelty.

An adaptive threshold $\rho_{min}$ is used to determine which behaviors to include in the archive. If the novelty of a new individual is higher than the threshold ($\rho(x) > \rho_{min}$), it is added to the archive. To keep the size of the archive approximately constant, $\rho_{min}$ is increased by a fixed fraction if the number of added behaviors exceeds the $add_{max}$ threshold in a certain number of evaluations. If the number of added behaviors is lower than $add_{min}$ in a certain number of evaluations, $\rho_{min}$ is decreased by a fixed fraction.

Since reproduction occurs just as in traditional evolutionary algorithms, novelty search can be easily implemented by replacing the fitness function in such an algorithm with the novelty metric. Since the relationship between genotypes and behavior may be quite complex, novelty search cannot purposefully explore behavior space, i.e., it cannot systematically generate novel behaviors. It can only generate new genotypes through mutation and crossover and measure the novelty of the resulting behaviors. Thus, it uses novel individuals as a guide towards finding more novel individuals, just as fitness-based methods use fit individuals as a guide to finding even fitter ones. Lehman and Stanley [6] argue that this approach is feasible when "many points in the search space collapse to the same point in behavior space".

Novelty search was originally evaluated in two deceptive maze tasks, in which behavior is characterized by the robot's final position after a fixed number of time steps [6]. Novelty search greatly outperforms fitness-based optimization in these tasks. Later, novelty search was also shown to be effective at discovering neuromodulated neural networks for deceptive tasks that require learning [16, 15]. Mouret and Doncieux [13] show that behavioral diversity (the novelty metric with $\mu_i$ in population only) can overcome the bootstrap problem, i.e., a lack of fitness gradient during the early stages of evolution. Gomez [4] shows that, in partially observable tasks, using a behavioral distance measure to maintain diversity can significantly improve performance over standard genotypical distance measures.

In addition to confirming the efficacy of novelty search, Lehman and Stanley [8] have also analyzed its properties. In the maze domains and the deceptive Santa Fe Trail and Los Altos benchmarks, they demonstrate that novelty search evolves successful solutions more consistently than fitness-based optimization. They also show that, although novelty search finds more functionally complex solutions, the genetic complexity is consistently lower.

In a separate article, they show that, on a single deceptive maze, there is no significant decrease in performance when the dimensionality of the behavior characterization is increased [7]. They hypothesize that this is due to the combination of novelty search with NEAT [18], a neuroevolutionary method which starts with simple networks and builds up to larger ones. They conclude that "a high-dimensional behavior characterization is not a sufficient basis for predicting that novelty search should fail."

They also find that reducing the precision of the behavioral characterization has complex effects. In the maze domain, performance is not strongly affected by a reduction in precision, unless the reduction is too strong. They conclude that "conflation is harmful to the search for novelty if behaviors are conflated in a way that interferes with the

discovery of stepping stones." However, they do not offer a precise definition of stepping stones.

Finally, they note that when the size of the behavior space is unlimited, the reliability of novelty search greatly decreases. In their most recent paper, they address this problem by proposing *minimal criteria novelty search*, in which an individual's novelty is set to zero if it does not meet minimal performance criteria [9]. They demonstrate that this approach can speed novelty search by leveraging domain knowledge to shrink the behavior space.

# 3. DESCENDANT FITNESS PROBABILITY

The goal of this paper is to extend and refine Lehman and Stanley's work on analyzing the properties of novelty search. We aim for a deeper understanding of what factors affect its performance in various tasks. In particular, we focus on the behavior function as a critical element. While Lehman and Stanley assert that novelty search is feasible when "many points in the search space collapse to the same point in behavior space", we claim that novelty search is feasible when many *similar* points in the search space collapse to the same point in behavior space. In this section, we propose a precise notion of similarity between two genotypes based on *descendant fitness probability* (DFP), which describes how likely a genotype's descendants are to have a certain fitness.

The ultimate goal of optimization methods is to find the fittest genotypes. A proximate goal is to find genotypes with great potential: those whose descendants will be highly fit. Fitness-based methods use a genotype's fitness as a measure of its potential. The problem of deception arises because this measure may be misleading (e.g., at local optima).

Novelty search avoids this problem by ignoring fitness and searching for novel behaviors. However, since genotype space is typically large or infinite in size, the behavior function must map many genotypes to the same behavior to keep the search feasible. Once a genotype is discovered, others that map to the same behavior will not be considered novel and are thus less likely to be selected for reproduction. Intuitively, mapping such genotypes to the same behavior is good when the genotypes have similar potential, as it reduces the size of the search space, and bad when they have dissimilar potential, as important genotypes will be overlooked. The central question, then, is how to define potential.

We propose that descendant fitness probability is a good measure of a genotype's potential. Given a genotype, a current population, and selection and reproduction operators, it is possible to compute a probability distribution over the possible genotypes created as children. This calculation can be repeated recursively to compute a distribution over the grandchildren, great-grandchildren, etc. In principle, the fitness of each possible descendant can be measured and used to compute a distribution over the fitness of these descendants, which we refer to as DFP.

Unfortunately, this notion of DFP is not helpful in analyzing in novelty search. The main problem is that it is not fixed across time, since it depends on the current population. Thus, it cannot be used to determine which genotypes should be mapped to the same behavior, since such mappings are typically fixed across time. Therefore, we propose a simpler, heuristic measure of DFP that models only mutation. By assuming asexual reproduction, we render DFP independent of the current population. Our simpler definition is based on *descendant genotype probability* (DGP):

DEFINITION 1. *The descendant genotype probability* $\mathrm{D}_K^G(x, y, m)$ *is the probability that a genotype* $x$ *will have a descendant* $y$ *after* $K$ *generations assuming asexual reproduction with mutation operator* $m$*:*

$$\mathrm{D}_0^G(x, y, m) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

$$\mathrm{D}_K^G(x, y, m) = \sum_{z \in G} \mathrm{D}_{K-1}^G(x, z, m) \cdot \Pr(d_z = y | z, m)$$

*where* $d_z$ *denotes the direct descendant of* $z$ *and* $G$ *denotes the set of all possible genotypes.*

DEFINITION 2. *The descendant fitness probability* $\mathrm{D}_K^F(x, f, m)$ *is the probability that a genotype* $x$ *will have a descendant with fitness* $f$ *after* $K$ *generations assuming asexual reproduction with mutation operator* $m$*:*

$$\mathrm{D}_K^F(x, f, m) = \sum_{y \in Y} \mathrm{D}_K^G(x, y, m)$$

*where* $Y = \{z \in G | F(z) = f\}$ *and* $F : G \to \Re^+$ *is the fitness function.*

Note that we do not propose DFP as a tool for helping design behavior functions for real tasks. On the contrary, calculating even our simplified DFP is intractable on non-trivial problems. Furthermore, even if the DFP of each genotype was known, their fitnesses would also be known and thus no optimization problem would remain. Instead, we merely propose DFP as a critical ingredient for analyzing novelty search and use it to formulate hypotheses in Section 4.

Using DFP, we can now enumerate the four different relationships that pairs of genotypes can have. To illustrate these relationships, we use a running example with 16 genotypes laid out in a 4$x$4 grid as shown in Fig. 1. Each black square denotes a genotype, the bottom left square denotes (0;0), and each letter (and color) denotes a behavior. Thus, different genotypes with the same letter map to the same behavior. The optimal genotype is (3;3), and the fitness of the other genotypes is 6 minus the Manhattan distance to the optimal genotype. The offspring of a genotype will with equal probability be the same or any adjacent genotype.

DEFINITION 3. *Behavioral discrimination* $BD(x_1, x_2)$ *occurs between two different genotypes* $x_1$ *and* $x_2$ *iff they have different behaviors and different DFPs:*

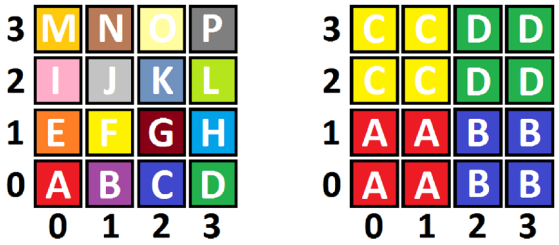
$$\mathrm{BD}(x_1, x_2) \Leftrightarrow (B(x_1) \neq B(x_2)) \wedge$$
$$\exists_{k,f}(\mathrm{D}_k^F(x_1, f, m) \neq \mathrm{D}_k^F(x_2, f, m))$$

Behavioral discrimination is illustrated in both grids in Fig. 1 between genotypes (1;0) and (2;0). Increasing the number of behaviorally discriminated genotype pairs increases the size of the behavior space.

DEFINITION 4. *Behavioral overdiscrimination* $BOD(x_1, x_2)$ *occurs between two different genotypes* $x_1$ *and* $x_2$ *iff they have different behaviors but the same DFP:*

$$\mathrm{BOD}(x_1, x_2) \Leftrightarrow (B(x_1) \neq B(x_2)) \wedge$$
$$\forall_{k,f}(\mathrm{D}_k^F(x_1, f, m) = \mathrm{D}_k^F(x_2, f, m))$$

Behavioral overdiscrimination is illustrated in both grids in Fig. 1 between genotypes (3;0) and (0;3), which have the same DFP due to symmetry about the diagonal. Intuitively, behavioral overdiscrimination is undesirable because pairs of genotypes with the same DFP are unnecessarily treated as different. Increasing the number of overdiscriminated genotype pairs increases the size of the behavior space.




(a) (b)

**Figure 1: Examples of behavioral discrimination and overdiscrimination (a and b) as well as weak behavioral aliasing and strong behavioral aliasing (b).**

DEFINITION 5. *Weak behavioral aliasing* $WBA(x_1, x_2)$ *occurs between two different genotypes* $x_1$ *and* $x_2$ *iff they have the same behavior and the same DFP:*

$$\text{WBA}(x_1, x_2) \Leftrightarrow (B(x_1) = B(x_2)) \wedge$$
$$\forall_{k,f}(\text{D}_k^F(x_1, f, m) = \text{D}_k^F(x_2, f, m))$$

Weak behavioral aliasing is illustrated in Fig. 1b, e.g., between genotypes (0;1) and (1;0), which were overdiscriminated in Fig. 1a but have the same DFP due to symmetry. Intuitively, weak behavioral aliasing is desirable because pairs of genotypes with the same DFP are collapsed into one in behavior space. Thus, we expect that, when the DFP of two genotypes is the same, weak behavioral aliasing is preferable to behavioral overdiscrimination. Increasing the number of weakly aliased genotype pairs decreases the size of the behavior space.

DEFINITION 6. *Strong behavioral aliasing* $SBA(x_1, x_2)$ *occurs between two different genotypes* $x_1$ *and* $x_2$ *iff they have the same behavior but different DFPs:*

$$\text{SBA}(x_1, x_2) \Leftrightarrow (B(x_1) = B(x_2)) \wedge$$
$$\exists_{k,f}(\text{D}_k^F(x_1, f, m) \neq \text{D}_k^F(x_2, f, m))$$

Strong behavioral aliasing is illustrated in Fig. 1b, e.g., between genotypes (0;0) and (1;0). Intuitively, strong behavioral aliasing can be either good or bad depending on the degree of similarity between the DFPs. Increasing the number of strongly aliased genotype pairs decreases the size of the behavior space.

## 4. HYPOTHESES

In this section, we propose two hypotheses that use the above definitions to relate changes in the behavior function to the performance of novelty search. Because DFP is a property of the task, we assume that the designer of the behavior function cannot alter it. Consequently, the possible effects of changing the behavior function are limited to switching between behavioral overdiscrimination and weak behavioral aliasing or between behavioral discrimination and strong behavioral aliasing, as illustrated in Table 1. Each of our hypotheses predicts the effect on performance of one of these changes. We equate performance with *evaluation costs*, i.e., the number of genotype evaluations needed for novelty search to discover an optimal genotype.

HYPOTHESIS 1. *A change in the behavior function transforming pairs of BOD genotypes to WBA genotypes decreases expected evaluation costs.*

When behavioral overdiscrimination occurs, two genotypes map to different behaviors even though their DFP is the same. As a result, novelty search can regard both genotypes as novel and spend time exploring both their offspring.

| | | $B(x_1) \neq B(x_2)$ | | $B(x_1) = B(x_2)$ |
|---|---|---|---|---|
| $\forall_{k,f}$ | $\text{D}_k^F(x_1, f, m) = \text{D}_k^F(x_2, f, m)$ | $\text{BOD}(x_1, x_2)$ | $\Leftrightarrow$ | $\text{WBA}(x_1, x_2)$ |
| $\exists_{k,f}$ | $\text{D}_k^F(x_1, f, m) \neq \text{D}_k^F(x_2, f, m)$ | $\text{BD}(x_1, x_2)$ | $\Leftrightarrow$ | $\text{SBA}(x_1, x_2)$ |

**Table 1: Changing the behavior function can change genotype pairs from BOD to WBA and from BD to SBA.**

However, since their DFPs are the same, each has the same chance of producing fruitful descendants, making this exploration redundant. Since changing such genotype pairs into ones with weak behavioral aliasing avoids such redundancy, we expect it to reduce evaluation costs.

HYPOTHESIS 2. *A change in the behavior function transforming pairs of BD genotypes to SBA genotypes can decrease expected evaluations costs only when genotypes with dissimilar behaviors (before the change) but similar DFPs are grouped into a single behavior (after the change).*

When BD genotype pairs are transformed into SBA ones, genotypes with different DFPs become mapped to the same behavior. On the one hand, this reduces the size of the behavior space, simplifying the search problem. On the other hand, discriminative power is lost, since the genotypes given identical behaviors do not have identical DFPs.

The hypothesis states that, in order for this trade-off to produce an overall decrease in evaluation costs, two conditions must be met. First, the genotypes that become mapped to the same behavior must have previously had dissimilar behaviors. Thus, the genotypes would previously have been considered novel w.r.t. each other but are now treated as the same, yielding a smaller search problem.

The second condition is that the genotypes that become mapped to the same behavior must have similar DFPs. Clearly, reducing the size of behavior space will not be beneficial if genotypes with arbitrarily different DFPs are treated as the same. For the benefit of a smaller behavior space to be worth it, the loss of discriminative power must be minimized. In essence, Hypothesis 2 can be viewed as a non-binary version of Hypothesis 1. The DFPs need only be similar, not identical, but the behaviors must previously have been truly dissimilar, as opposed to merely not identical.

## 5. ARTIFICIAL DOMAIN EXPERIMENTS

As a first test of our hypotheses, we consider an artificial domain with a simple genotype representation in which the effect of mutations on behavior is predictable. The artificial domain consists of a two-dimensional grid world. A genotype $x$ consists of $P$ independent points on the grid, where each point is a pair of integers in $\{0, 1, \ldots, L\}$ for some constant L. The goal is to place one or more of these points in $(L-1; L-1)$, the cell farthest from the origin.

Unless stated otherwise, the genotype and behavior are exactly the same, i.e., each genotype point is a behavioral feature. This formulation makes it possible to measure the effect of behavioral changes in the absence of confounding factors, since they result directly from changes to genotypes. Nonetheless, we assume that the relationship between genotypes and behavior is not known to the designer and is not exploited by novelty search.

Though novelty search is typically used with NEAT, we employ a simple genetic algorithm to keep the underlying

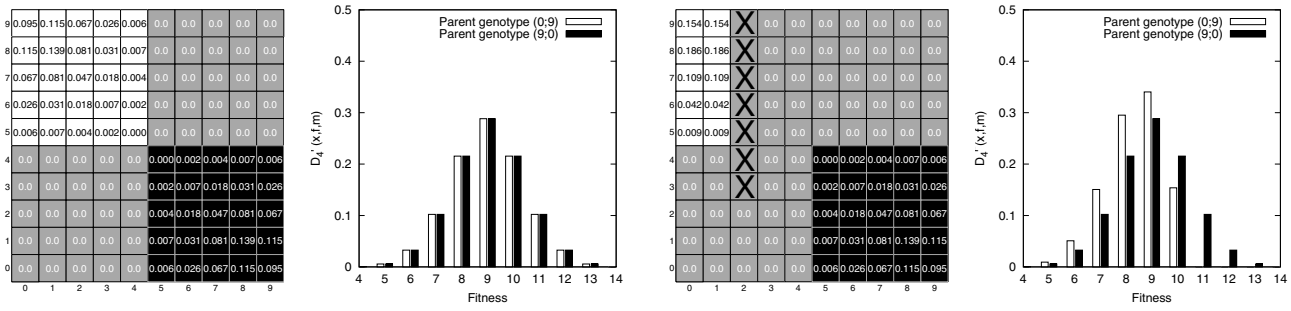



**Figure 2: DGPs and DFPs in the open grid (a and b) and the wall grid (c and d). DGP and DFP given parent genotype (0;9) is shown in white squares/bars and given parent genotype (9;0) in black squares/bars. Grey squares denote the same values for both parents.**

mechanisms as simple and predictable as possible. The initial population is filled with genotypes at the origin $(0;0)^P$. Offspring are created from a single parent genotype mutated with probability 1. Mutations are applied independently to all points. The distance function is Manhattan distance. In selection, each individual in the top ten has an equal probability of getting selected for reproduction, which is steady state: the offspring replaces the worst individual in the pool.

All results reported below are averaged over 70 independent runs, each limited to $2 \cdot 10^6$ genotype evaluations. Runs that did not discover the optimal genotype within this time were given evaluation costs of $2 \cdot 10^6$. Unless noted otherwise, the statistical significance of all observed performance differences was verified using a Student's t-test ($p < 0.05$).

## 5.1 Measuring DFP

To test our hypotheses, we need to know when two genotypes have the same or similar DFPs. Because of the simplicity of the artificial domain, it is feasible to compute the DFPs of every genotype in each variation of the artificial domain used in the experiments below.

The first variation, the *open grid* shown in Fig.2a, involves one genotype point in a $10 \times 10$ grid, i.e., $L = 10$ and $P = 1$. Genotype mutations are integer changes in $[-1, +1]$ and $F(x) = 18 - \text{Manhattan}(x, (9;9))$. We computed both DGP and DFP for $K = 4$ for each genotype. As expected, because this domain is symmetrical about the $(0;0)/(9;9)$ diagonal, behavioral overdiscrimination occurs: each point above the diagonal has the same DFP as the corresponding one below it. Figs. 2a and 2b show the DGPs and DFPs, respectively, for the points $(0;9)$ and $(9;0)$. Where computationally feasible, we use a second variation, the *large grid*, in which $L = 100$, $P = 1$, mutations are in $[-3, +3]$, and $F(x) = 198 - \text{Manhattan}(x_1, (99;99))$. Though not shown in the figure, the DFP calculations were qualitatively identical: DFPs are symmetrical about the diagonal.

The third variation, the *wall grid* shown in Fig. 2c, is exactly like the first but with some genotypes excluded, forming a wall in genotype space. As expected, this introduces an asymmetry, preventing behavioral overdiscrimination: the points above the diagonal no longer have the same DFP as those below it. Figs. 2c and 2d show that the DGPs and DFPs for the points $(0;9)$ and $(9;0)$ now differ.

## 5.2 Testing Hypothesis 1

To test Hypothesis 1 in the artificial domain, we use the large grid, but with four points ($P = 4$). However, only the first is relevant to the task, i.e., the goal is to place point 1 at $(99;99)$. We first treat each genotype point as a behavioral feature and then examine how novelty search's performance changes as genotype points $2 - 4$ are removed as behavioral features. Because these features are irrelevant, genotypes that differ only w.r.t. to such points have the same DFP and are thus behaviorally overdiscriminated. Consequently, Hypothesis 1 predicts that removing such behavioral features should reduce evaluation costs.

Figure 3 shows the results. Not surprisingly, they confirm our hypothesis. Since the size of the behavior space grows exponentially w.r.t. the number of behavior features, removing irrelevant dimensions greatly improves performance. With three behavior points, 21 of the 70 runs did not find an optimal genotype even after $2 \cdot 10^6$ evaluations. For four behavior points, 29 runs failed to do so.

As a second test of Hypothesis 1, we reduce the behavior space even further. Though there are no more irrelevant behavior features, the domain still contains behavioral overdiscrimination due to symmetry about the diagonal, as shown in Section 5.1. We can eliminate this overdiscrimination by changing the behavior function such that the genotype corresponding to each cell in the top-left part of the grid maps to the same behavior as its counterpart with identical DFP in the right-bottom part of the grid. In other words, the grid is virtually folded over the $(0;0)/(99;99)$ diagonal. Figure 3 shows these results also, marked '1p no BOD'. As expected, eliminating the remaining behavioral overdiscrimination results in another substantial reduction in evaluation costs, from $63,440$ to $27,427$ on average.

So far, these results show only that reducing the behavior space can improve performance. To examine the importance of creating weak behavioral aliasing, we must compare against a scenario in which strong behavioral aliasing is induced instead. We can create such a scenario by folding the grid over the $(0;99)/(99;0)$ diagonal instead. This results in a behavior space of exactly the same size as in the '1p no BOD' scenario but with different DFPs mapped to the same behavior. Figure 3 shows these results also, marked '1p SBA'. As expected, performance is quite poor in this scenario because novelty search cannot properly discriminate between genotypes with different potential. In fact, the optimal genotype was found in only 6 of the 70 runs.

## 5.3 Testing Hypothesis 2

To test Hypothesis 2 in the artificial domain, we conduct two experiments that create strong behavioral aliasing, one by grouping genotypes with similar behaviors, and one by grouping genotypes with dissimilar behaviors.

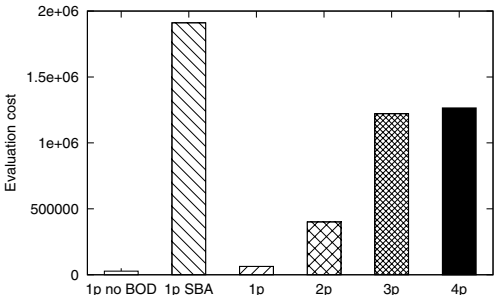



**Figure 3: Effects on average evaluation costs when reducing behavioral overdiscrimination.**

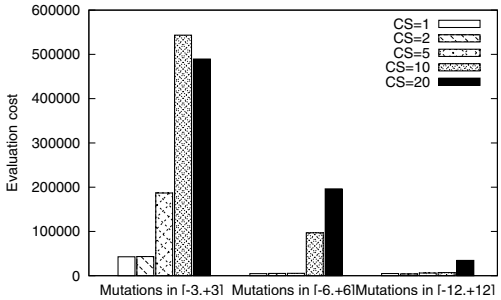



**Figure 4: Effects on average evaluation costs when SBA is created by grouping similar behaviors.**

The first experiment uses the large grid. To create strong behavioral aliasing, we lay a coarser grid over the genotype grid and map each cell in the same square 'supercell' of the coarser grid to the same behavior. Because the genotypes that are grouped are near each other in the grid, they have similar behaviors before they are grouped.

The left portion of Fig. 4 shows the results for cell sizes (CS), i.e., the length and width of each supercell, 1, 2, 5, 10 and 20. As predicted by Hypothesis 2, introducing SBA by grouping similar behaviors does not improve performance. Though the size of the behavior space is reduced, novelty search is not sped up, since the genotypes that now map to the same behavior were already considered similar by novelty search and thus unlikely to be redundantly explored.

The results also show that such grouping can hurt performance, as CS $> 5$ leads to significantly higher evaluation costs. Understanding why requires examining the relationship between CS and the magnitude of genotype mutations. Thus, we repeat the above experiment with mutation ranges $[-6, +6]$ and $[-12, +12]$. The results are also shown in Fig. 4. As before, grouping similar behaviors never reduces evaluation costs. However, whether doing so increases evaluation costs depends on whether the supercell's size exceeds the magnitude of the genotype mutations. When the mutation range is $[-3, +3]$, performance drops suddenly as CS increases from 2 to 5. For mutation range $[-6, +6]$ it drops as CS increases from 5 to 10, and for mutation range $[-12, +12]$ it drops as CS increases from 10 to 20. When many genotypes are grouped together in one supercell, novelty search in that supercell reduces to a random search. If the mutations are small compared to the supercell's size, the chance of entering a new supercell through random search, and thus discovering novel behavior, is also small.

The second experiment, in which we create strong behavioral aliasing by grouping dissimilar behaviors, uses the wall grid. However, the genotype consists of three points ($P = 3$), all relevant. Thus, the goal is to get genotype points 1, 2 and 3 equal to (9;9) and $F(x) = 54 - \sum_{i=1}^{3} \text{Manhattan}(\phi_{x_i}, (9;9))$. To make the experiments computationally feasible, we consider any $x$ for which $F(x) \geq 53$ to be optimal.

As with the '1p no BOD' setting in Section 5.2, we reduce the behavior space by folding the grid over the $(0;0)/(9;9)$ diagonal. Thus, many points that are far away from each other obtain the same behavior, leading to the grouping of dissimilar behaviors. Because of the wall, the grouped genotypes have different DFPs. Since Hypothesis 2 predicts that the similarity in DFP is important, we also consider two variations in which the grouped behaviors have more or less similarity in their DFPs. To increase similarity in DFP, we fold the grid along the same diagonal but only for the subgrid with corners at (3;3) and (9;9). Because none of the genotypes in this subgrid are 'blocked' by the wall, our analysis in Section 5.1 found that they have more similar DFPs with their counterparts across the diagonal. To decrease similarity in DFP, we again fold along the same diagonal but now only for the regions outside this subgrid.

The results, shown in Figure 5, demonstrate that transforming pairs of BD genotypes to pairs of SBA ones can significantly reduce evaluation costs. However, as predicted by Hypothesis 2, performance depends on the similarity in DFP of the grouped genotypes. In 'SBA in sub', the grouped genotypes have relatively similar DFPs and evaluation costs are reduced by more than a factor of two. However, comparing 'No SBA' to 'SBA out sub' and comparing 'SBA in sub' to 'SBA in full' shows that the addition of groupings with dissimilar DFP leads to small increases in evaluation costs. Due to high variance in evaluation costs between test runs, these differences were not significant.

These results demonstrate that creating strong behavioral aliasing can improve performance, but only up to a limit, after which the discriminative power of the behavior function is too low and performance begins to decrease again. The results thus confirm the prediction of Hypothesis 2 that changing the behavior function such that previously dissimilar behaviors are grouped together will yield a performance benefit only when the genotypes grouped have similar DFPs.

# 6. MAZE DOMAIN EXPERIMENTS

In the artificial domain, the effects of mutations on behavior and DFP are easy to predict. As a result, we are able to run well controlled tests of our hypotheses that isolate the critical factors. However, such tests are also limited because neither the setting nor the algorithm are realistic. Therefore, in this section we present the results of additional experiments conducted in more realistic settings (deceptive maze domains) with a more realistic algorithm (novelty search based on NEAT).

We consider the three deceptive mazes shown in Fig. 6: the *medium* and *hard mazes* introduced by Lehman and Stanley [6] and a new maze we call the *star maze*. In these domains, a robot must navigate the maze using six range sensor and a rough compass giving the direction towards the goal (i.e., front, left, right, back). The robot has 400 time steps to navigate from the start to the goal, after which its fitness is calculated as $F(x) = b_f - d_g$ where $b_f$ is a constant bias to make sure fitness is always positive and $d_g$ the Euclidean distance between the robot and the goal. Any solution for which $d_g < 5$ is considered optimal.

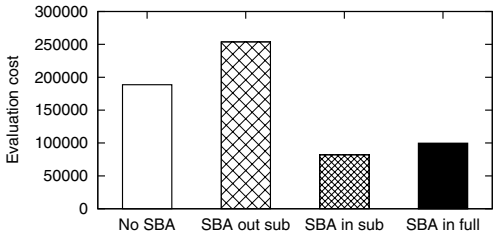



**Figure 5: Effects on average evaluation costs when SBA is created by grouping dissimilar behaviors.**

We consider two different behavior functions. In the *one-point* function, an individual's behavior is simply its $(x; y)$ position after 400 steps. In the *eight-point* function, its position is measured once every 50 timesteps. The robot is controlled by a neural network evolved with novelty-based NEAT using the same parameter settings as in [7]. Results are averaged over 70 independent runs, each of which is terminated after $2 \cdot 10^5$ genotype evaluations if no optimal solution has been found.

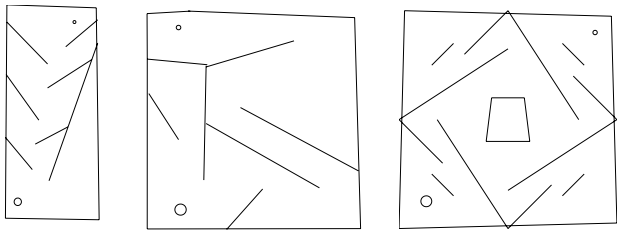
Because it is unlikely for two different neural networks to have exactly the same DFP, we focus only on testing Hypothesis 2 in this setting. Doing so is challenging because we need a way to measure when genotypes have similar DFPs and behaviors. Unlike in the artificial domain, it is no longer feasible to directly compute DFP because the genotypes contain continuous parameters, mutations are governed by continuous probability distributions, and, due to complexification, NEAT does not search in a fixed-dimensional space.

However, even without computing DFP exactly, we can still identify cases where it is reasonable to assume that two genotypes have similar DFP. In particular, since fitness depends only on the robot's final position, we assume that the one-point behavior function is approximately optimal. Since an optimal behavior function assigns similar behaviors to genotypes with similar DFPs, we can conclude that similarity in one-point behavior implies similarity in DFP.

Determining when genotypes have similar behaviors is also challenging because behaviors are affected, not only by the behavior function but by domain constraints, e.g., narrow passageways in the maze, and the neural network topology, which restricts which behaviors are reachable. For example, in the artificial domain, we can safely assume that removing behavior features causes dissimilar behaviors to be grouped together, since behavior features are independent. In contrast, in the maze domain, changing from eight-point behavior to one-point behavior may only group similar behaviors because, depending on the shape of the maze, the robot's positions at different times may be highly correlated.

Therefore, in each maze, we use the average correlation between the one-point and eight-point behaviors as an approximate measure of similarity. In particular, we record the behavior points encountered when using eight-point behavior and measure the average Pearson correlation coefficient, a standard measure of linear correlation, between the eighth point and each of the first seven points. A high value implies that switching from eight-point behavior to one-point behavior only groups behaviors that are already similar, whereas a low value implies that doing so groups dissimilar behaviors. Thus, Hypothesis 2 predicts that one-point behavior should outperform eight-point behavior in mazes with relatively low correlation between behavior points.

To test this prediction, we evaluated both eight-point and one-point behavior on each of the three deceptive mazes. The results are shown in Fig. 7. Using the results of the eight-point runs, we computed the average correlation, shown below each maze in the $x$-axis of Fig. 7.



(a) Medium (b) Hard (c) Star

**Figure 6: Deceptive mazes, in which the robot must navigate from the large circle to the small one.**

In the hard maze, the average correlation is high and there is no significant performance difference between eight-point and one-point behavior. This result is consistent with that of Lehman and Stanley [7], who demonstrate that a high-dimensional behavior characterization is not a sufficient basis for predicting that novelty search will fail. They hypothesize that this is due to the use of NEAT, which starts with small networks and complexifies. In domains where smaller networks cause stronger correlation between behavioral features, such complexification may mitigate the effects of these extra features. While our results do not rule out such a hypothesis, the high correlation between features does suggest another explanation: one-point behavior does not perform better because the behaviors it groups were already similar under eight-point behavior.

This explanation is further supported by the medium maze results, in which the average correlation is lower, and one-point behavior performs better. Though Lehman and Stanley conducted experiments in the medium maze, they do not present results for eight-point behavior in this domain. The presence of a performance difference in the medium maze and the absence of one in the hard maze suggests that correlation between the behavior points is indeed a critical factor in novelty search's performance. In addition, given our assumptions that individuals with similar one-point behavior have similar DFP and that low average correlation implies that switching from eight-point to one-point behavior groups dissimilar behaviors, these results also confirm Hypothesis 2.

After obtaining these results, we investigated several other maze domains in order to determine how broadly Hypothesis 2 holds. Each produced results consistent with those from the medium and hard mazes. The only exception was the star maze: though its average correlation is only slightly higher, at 0.55, than in the medium maze, switching to one-point behavior does not reduce evaluation costs. On the contrary, it substantially increases them. Thus, the star maze is an intriguing and puzzling domain for novelty search.

There are many possible explanations for this counterintuitive result. One is that Hypothesis 2 is incomplete, and that the star maze represents a corner case not accounted for in our current formulation. Another is that Hypothesis 2 is correct but our assumption that its conditions hold in the star maze is not. In particular, we assume that a one-point behavior function is approximately optimal and that the behaviors that become grouped therefore have similar DFPs. While this seems obvious in the hard and medium mazes, it may not hold in the star maze. In particular, we specu-

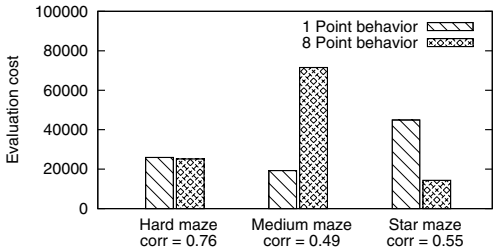



**Figure 7: Maze domain results.**

late that the presence of many narrow corridors in the star maze is an important factor. Though lack of space presents its presentation here, we conducted an analysis of the robot trajectories in the star maze and found that many individuals get stuck in these corridors, leading to many different solutions with nearly identical one-point behavior. Thus, it may be that an individual's entire trajectory, and not just its end point, is an important factor in its offsprings' potential.

# 7. DISCUSSION AND FUTURE WORK

Overall, the results presented in this paper offer several new insights about the critical factors in the performance of novelty search. At a high level, the results demonstrate that the design of a good behavior function is essential to novelty search's success. At a lower level, the results isolate both similarity in descendant fitness probability and behavior as central characteristics of the behavior function. The genotypes mapped to the same behavior should have similar DFPs or the benefit of a smaller behavior space will be offset by the lack of discriminative power. In addition, the genotypes that become grouped should previously have had dissimilar behaviors or the change in the behavior function will not substantially speed the search.

On the one hand, these findings are good news for novelty search, as they identify a broad range of scenarios in which the behavior function can supply substantial leverage for speeding search. On the other hand, these findings are bad news, as they underscore the difficulty of finding a good behavior function in practice. Doing so with confidence requires reasoning about DFP, which is infeasible to compute in realistic problems. In general, finding a good behavior function may be as difficult as solving the optimization problem itself. This problem is less severe for reasoning about behaviors, since designers often have good intuition about what behaviors are similar. However, the results shown in Section 6 illustrate that such reasoning can also be tricky. Even domain experts may have difficulty explaining why average correlation is higher in the hard maze than in the medium maze, much less why the star maze benefits from a larger behavioral characterization.

Thus, the irony is that novelty search, which strives to operate independently of fitness, relies on a form of prior knowledge that is inherently connected to it. While it remains a radical and useful approach to addressing the problem of deception, these results show that it is, at least with respect to its use of prior knowledge, not so different from other optimization methods after all.

Several avenues for future work are suggested by the results presented here. First, additional study of the star domain and variations thereof could shed new light on how broadly our hypotheses hold and how qualitative changes to a domain affect the optimal behavior function. Second, further study of DFP, including alternative heuristics, could arm designers with valuable intuition to aid in the selection of a behavior function. Finally, other aspects of novelty search remain to be analyzed. For example, while Lehman and Stanley have considered two different rules for adding individuals to the archive [6, 8], others are possible and the effect of such choices has never been investigated.

# 8. REFERENCES

[1] P. Darwen and X. Yao. Every niching method has its niche: Fitness sharing and implicit sharing compared. In *PPSN IV*, volume 1141, pages 398–407, 1996.

[2] K. A. De Jong. *An analysis of the behavior of a class of genetic adaptive systems*. PhD thesis, University of Michigan, Ann Arbor, 1975.

[3] F. Gomez and R. Mikkulainen. Incremental evolution of complex general behavior. In *Adaptive Behavior*, volume 5, pages 317–342. MIT Press, 1997.

[4] F. J. Gomez. Sustaining diversity using behavioral information distance. In *GECCO-09*, pages 113–120, 2009.

[5] G. R. Harik. Finding multimodal solutions using restricted tournament selection. In *ICGA-95*, pages 24–31, 1995.

[6] J. Lehman and K. O. Stanley. Exploiting open-endedness to solve problems through the search for novelty. In *ALIFE-XI*, 2008.

[7] J. Lehman and K. O. Stanley. Abandoning objectives evolution through the search for novelty alone. In *Evolutionary Computation*. 2010.

[8] J. Lehman and K. O. Stanley. Efficiently evolving programs through the search for novelty. In *GECCO-10*, pages 837–844, 2010.

[9] J. Lehman and K. O. Stanley. Revising the evolutionary computation abstraction: minimal criteria novelty search. In *GECCO -10*, pages 103–110, 2010.

[10] M. Lynch. The frailty of adaptive hypotheses for the origins of organismal complexity. *Proceedings of the National Academy of Sciences*, 104(Suppl 1):8597–8604, 2007.

[11] S. W. Mahfoud. *Niching methods for genetic algorithms*. PhD thesis, University of Illinois, Urbana-Champaign, 1995.

[12] T. Miconi. Evolution and complexity: the double-edged sword. *Artificial life*, 14(3):325–344, 2008.

[13] J.-B. Mouret and S. Doncieux. Overcoming the bootstrap problem in evolutionary robotics using behavioral diversity. In *CEC-09*, pages 1161–1168, 2009.

[14] A. Petrowski. A clearing procedure as a niching method for genetic algorithms. In *IEEE IECC*, pages 798–803, 1996.

[15] S. Risi, C. Hughes, and K. Stanley. Evolving plastic neural networks with novelty search. *Adaptive Behavior*, 2010.

[16] S. Risi, S. D. Vanderbleek, C. Hughes, and K. O. Stanley. How novelty search escapes the deceptive trap of learning to learn. In *GECCO-09*, pages 153–160, 2009.

[17] B. Sareni and L. Krahenbuhl. Fitness sharing and niching methods revisited. *Evolutionary Computation, IEEE Transactions on*, 2(3):97–106, 1998.

[18] K. O. Stanley and R. Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2):99–127, 2002.

[19] E. Uchibe, M. Yanase, and M. Asada. Behavior generation for a mobile robot based on the adaptive fitness function. *Robotics and Autonomous Systems*, 40:69–77(9), 2002.

[20] J. Urzelai, D. Floreano, M. Dorigo, and M. Colombetti. Incremental robot shaping. *Connection Science*, 10:341–360(20), 1998.

[21] D. L. Whitley. Fundamental principles of deception in genetic search. In *Foundations of Genetic Algorithms*, pages 221–241. Morgan Kaufmann, 1991.

[22] A. P. Wieland. Evolving neural network controllers for unstable systems. In *IJCNN-91*, pages 667–673, 1991.