Master Thesis

Entropy as an Organizing Principle for Selection in Evolutionary Robotics

Julius Fritz Faber
Matr.-Nr.: 352341
julfaber@gmail.com
Informatik

Date of submission: October 27, 2016

Examiners: Prof. Dr. Oliver Brock, Prof. Dr. Paulo Urbano
Advisors: Prof. Dr. Paulo Urbano, Dr. Joel Lehman, Prof. Dr. Oliver Brock

Technische Universität Berlin
School IV - Electrical Engineering and Computer Science
Department of Computer Engineering and Microelectronics
Robotics and Biology Laboratory

Declaration

I hereby declare in lieu of an oath that I have produced this work by myself. All used sources are listed in the bibliography. All content taken directly or indirectly from other sources is marked as such. This work has not been submitted to any other board of examiners and has not yet been published. I am fully aware of the legal consequences of making a false declaration.

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides Statt, dass ich die vorliegende Arbeit selbständig angefertigt habe. Andere als die angegebenen Quellen wurden nicht verwendet. Die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen wurden als solche kenntlich gemacht. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht. Ich bin mir bewusst, dass eine unwahre Erklärung rechtliche Folgen haben wird.

Ort/Datum

Julius F. Faber

Abstract

In evolutionary computation, a goal-based objective function is typically unable to include the local challenges on the way towards its fulfillment and tends to to cause the search to converge prematurely. Therefore, this work proposes to use objectives that are defined by different aspects of an individuals interaction with the environment and a selection procedure able to reallocate search efforts in order to avoid convergence. The objectives, curiosity, novelty and evolvability, differ in the time-scale they operate over and the amount of information they include about the problem structure. The common theme of these objectives is their tendency to increase the diversity of behaviors, which is assumed can act as general-purpose utility value. The goal is to combine the benefits of the objectives by optimizing subsets of them simultaneously with a multi-objective EA. Entropy is used as unifying framework for modelling the objectives and determining which of their values are considered desirable. Using entropy for the latter part relies on the fact that degenerate behaviors are more pervasive in search spaces than functional ones. Thus, selecting for high-entropy values implicitly treats the frequency with which a behavior occurs as a heuristic of its interestingness, reallocating search efforts towards diversity. The performance of the different objectives and selection methods are evaluated by solving deceptive navigation tasks. Verified on a more challenging biped locomotion experiment, the new finding of this work is that entropy selection is as good or better than optimization. Concerning the individual objectives, these work's results support previous findings that novelty is a very good indicator for selection and additionally show that it can be efficiently modelled with entropy. The method of modelling novelty with entropy is shown to be applicable to many, possibly higher dimensional and less informative behavioral characterizations simultaneously without a decrease in conceptual simplicity and computational efficiency, indicating how future research could explore more complex behavior spaces and problems.

Concerning the evolvability objective, which describes the capacity to produce diversity and generalization, this work investigates how it can be estimated from the many individuals discarded during search, in order to avoid the many extra evaluations necessary to calculate it precisely. Also, this work proposes how an elitist-multiobjective EAs could interpret evolvability as adaptive variation without referring to a specific task. Taken together, the negative results of both evolvability-estimations indicate that different behaviors might have different potentials for evolvability and should therefore not be compared on it globally.

Kurzfassung

Die primäre Abstraktion in Evolutionären Algorithmen ist die sogenannte Fitness-Funktion, welche darüber entscheidet welche Individuen für die nächste Generation ausgewählt werden sollen. Typischerweise misst die Fitness-Funktion die hnlichkeit zu jenem Verhalten, welches die Aufgabe löst die der Experimentator im Sinn hat. So kann mit Hilfe der Fitness-Funktion zwar erkannt werden, dass die Aufgabe gelöst wurde, jedoch können keine Information über die lokalen Hindernisse modelliert werden auf welche ein Individuum auf der dem Weg zur ihrer Erfüllung trifft. Dies führt gewöhnlicherweise dazu, dass die evolutionäre Suche frühzeitig konvergiert und in lokalen Optima stecken bleibt.

Deshalb schlägt diese Arbeit vor die Fitness-Funktion dahingegend zu verändern, dass Sie unterschiedliche Interaktionen eines Individuums mit seiner direkten Umgebung erfasst. Auerdem präsentiert diese Arbeit eine Anpassung des Selektionsmechanismus, sodass Konvergenz vermieden wird. Die alternativen Zielfunktion, welche an Stelle der Fitness-Funktion zur Bewertung der Individuen dienen, sind nach den biologischen Eigenschaften Neugier, Neuheit und Evolvierbarkeit modelliert. Sie unterscheiden sich bezüglich der Zeitspanne innerhalb der sie definiert sind und der Menge an Information die sie aus der Umgebung des Individuums verwenden. Ihre Gemeinsamkeit ist, dass Sie alle die Vielfältigkeit von Verhalten belohnen, welches als allgemeingültiges Nützlichkeitskriterium verstanden wird. Ziel dieser Arbeit ist es die Vorteile der einzelnen Eigenschaften miteinander zu vereinen, indem man sie gleichzeitig mit bereits bekannten und dafür geeigneten Algorithmus optimiert.

Als Rahmen um die drei Eigenschaften zu modellieren und auszuwählen welche ihrer Werte für den Fortschritt der Suche besonders wichtig sind, wird der informationstheoretische Term Entropie verwendet. Dieser ist besonders dafür geeignet, weil er die gesamte Suchhistorie in einer Verteilung darstellen kann. So kann genutzt werden, dass die Mehrheit aller Verhalten im Suchraum degenerierte Verhaltensweisen sind und nur wenige tatsächlich funktional. So reicht allein die Häufigkeit, mit der ein Verhalten gefunden wird als Indikator für seinen Wert bzgl. der Suche und kann somit genutzt werden um das Konvergieren der Suche zu vermeiden.

Um den Erfolg der vorgeschlagenen Methoden zu testen, werden die Eigenschaften benutzt um zwei Navigationsaufgaben in Labyrinthen zu lösen und eine dritte, deutlich schwierigeres Problem zu adressieren: Das Koordinieren eines zweibeinigen Roboters. Die Ergebnisse dieser Arbeit bestätigen bereits bekannte Erkenntnisse, dass Neuheit allein gut geeignet um zu selektieren und dass sie effizient mit Entropie modelliert werden kann. Es wird gezeigt das letzteres auch für höher dimensionale und weniger informative Verhaltensbeschreibungen ohne einen höheren rechnerischen Aufwand effizient gelöst werden kann, sodass diese Methods auch für komplexere Probleme und Verhalten anwendbar ist.

Da die Berechnung von Evolvierbarkeit normalerweise viele zusätzliche Evaluation impliziert, untersucht diese Arbeit weiterhin, ob es möglich ist diese zu vermeiden und entwickelt mehrere Heuristiken welche versuchen Evolvierbarkeit mit Hilfe des Stammbaum eines Individuums abzuschätzen. Ebenfalls wird eine neue Interpretation vorgeschlagen, wie Evolvierbarkeit unabhängig von einer bestimmten Aufgabe verstanden und gefördert werden kann. Das Ergebnis, das keine der beiden Methoden funktionieren, wird analysiert und die Vermutung aufgestellt, dass unterschiedliche Nischen unterschiedliche Potential für Evolvierbarkeit haben und dieses Kriterium deshalb weniger für globale Selektion geeignet ist.

Contents

List of Figures

List of Tables

1. Introduction

1.1. Introduction to Evolutionary Robotics

The classic approach in robotics is to build a robot incrementally: First the actual robot is designed, equipped with sensors as the roboticists think is appropriate, followed by modules for perception, planning and action. Evolutionary Robotics (ER) [3] takes a different approach by evolving the complete sensorimotor control loop and/or the robot morphology in one go. ER has successfully been applied to rather simple, well defined robotics problems such as navigation tasks [4], ball-collection [5], locomotion [6, 7, 8] or robot design [9, 10]. An example of ER's long term ambitions is given by the Golem Project [11], which evolved original, 3D-printable robot morphologies, including control structures that would execute efficient gaits. Such automated, autonomous design of solutions allows to avoid any preconceived notions and anthropocentric bias a human designer might have and can potentially find new innovative solutions, developed from the robot's embodied point of view of the world. The benefits of this more holistic approach become evident when ones tries for example to design energy efficient and compliant robots: By seeing design and control as one problem instead of two separate ones, one can discover and exploit synergies between morphology and control structure. The cockroach gives an impressive and well-studied example for that. It achieves extraordinary speed

and can traverse obstacles three times the height of its center of mass with a relatively simple motor control pattern. The coackroaches' secret lies in a dynamic interaction of its posture, a timed feed-forward controller and the passive-elastic properties of its structure, also known as 'preflexes' [12].
But yet, such goals are still far from being reached and as it is still a relatively young field, there is an uncertainty about the practical role ER can play in the field of artificial intelligence (AI) [13]. The traditional perspective casts evolution as a metaheuristic black-box optimizer, suited for finding the best solution to a narrowly defined task. An alternate, more recent perspective focuses more on evolutions capacity to produce a diversity of solutions [14, 15, 9]. ultimately aiming to provide the roboticists with a representative collection from the whole spectrum of solutions, instead of just one possibly optimal solution. The following paragraph will briefly introduce the basic principles shared by all types of evolutionary algorithms, whatever their desired purpose.

1.1.1. Principles of Evolutionary Computing

To capture the essence of natural evolution in an algorithm, methods from evolutionary computing, such as genetic programming [16] or evolutionary robotics, adopt the principle of blind variation and natural selection to a population of randomly assembled solutions. These solutions (also called individuals) are encoded on *genomes* - short parametric descriptions from which the solution can be reconstructed. Genomes can be modularly reassembled so that individuals can spawn offsprings either by simply copying their genomes or by mating with another individual. In genetic programming for example, where the solutions are programs, genomes consists of lines of code. In evolutionary robotics, a common choice is to evolve individuals as control structures for certain robot morphologies. The control structure most typically used are artificial neural network (ANNs), whose inputs are the robots sensor values and its outputs are the motor commands used to drive the robot. The morphology and weights of the ANN are then encoded on a genome, which, when loaded into the robot, can interact with the environment for a certain amount of time, thereby creating the individual's *behavior* (also called phenotype). Each generation the most successful controllers are chosen for reproduction and mutation, generating the next generation of controllers which, on average, is more successful than the generation before. Thus, the idea is that over generations behaviors evolve that are adapted to the environment and achieve high performance at a selected task.
In nature, the selection process is assumed to favor those mutations that are

beneficial and increase the chances of an individuals survival. In the case of ER, the selection is based upon a utility value assigned to each candidate solution by a specific function. This function is to favor those candidates that are more likely to be successful at the problem the experimenter set out to solve. Whereas most branches of AI refer to the function that is used to measure the progress towards a goal as the objective function, evolutionary roboticists call it the *fitness function*, metaphorically referring to Darwin's "survival of the fittest".

1.1.2. Problems of Fitness-Guided Search

Whereas fitness in nature is associated with high reproductive success, it can be freely defined by the experimenter when evolution is simulated on the computer. Usually it is done so with respect to a specific task the experimenter wants to solve. Akin to a horse breeder who acts as a selective force by only allowing the fastest animals to mate, evolutionary roboticists have hoped to evolve solutions to problems simply by describing how success on their problem would look like. If, for example, the experimenter wanted to evolve a controller for a bipedal walking robot, the fitness function would measure how far the robot walked before falling over. Thus, it would not pose any constraints on how the walking motion should look like. Possibly because the experimenter is not able to or because he does not want to specify how a solution to that problem shall be found. Consequently, the fitness function is just measuring how far a candidate solution is away from fulfilling the goal and is, next to the representation of the genomes, the only aspect specified by the experimenter. But as the fitness function must judge both a candidates proximity to the goal and whether it has good prospects of fulfilling it in the future, its design is particularly hard. Especially the latter part is problematic, as there is no guarantee that there exists a smooth path of increasing fitness towards the goal.

This causes problems at different stages of evolution: At the very beginning of evolution, when all solutions in the population are equally bad, the process has trouble to get traction and take off, which is referred to as the bootstrap problem [17]. The underlying assumption when using a fitness function modeled on the desired behavior is that a series of small improvements towards the goal will eventually reach it. This leads to the most severe problem: the fitness function causes the population to prematurely converge into a single or few types of mediocre solutions. This is the problem of local optima, which is ubiquitous in AI [18].

For more complex problems, the ones on which ER ultimately aspires to be of help to roboticists, the shape of the fitness landscape can be very rugged, neutral or deceptive [19]. In those cases it provides no hints, many contradicting hints or deceptive ones that lead the search into a dead-end. Complex problems, such as controlling a bipedal robot, often require the discovery of many intermediate solutions, some of them bearing no resemblance to the solution itself. In the example of the bipedal robot, intermediate solutions would be to balance the robot and to discover oscillatory motion patterns. But if fitness is the sole criterion used for selection, than individuals that discover those stepping stones will be discarded by the search process unless they immediately contribute to propelling the robot forwards. Therefore, the constant selective pressure towards greater performance can act as a constraint on how solutions can be discovered, inhibiting the discovery of innovative and more complex behaviors, which might at first traverse a path of decreasing performance but have a higher fitness in the long term.

This kind of convergence contrasts starkly with the prolific creativity and open-endedness of natural evolution, which produces a wide variety of solutions to the problem of surviving and reproducing in nature. Also, the organisms evolved by natural evolution are quick to adapt to changing environments. Similar adaptation skills are also desirable for robots, which, apart from a few industrial applications, are build with the intention to be usable for a wide range of tasks, some of them maybe unforeseen. If, for example a controller to for a multi-legged robot is evolved, it should be able to adapt its gait if one or several of its legs suffer damage or turn completely nonfunctional. Solutions evolved with fitness-based search paradigms make no assumptions about that, nor do they guarantee that the evolved solutions will be able to adapt to a slight variation of the environment, the goal, or to a completely new problem.

Although unintended, the fitness function thus restricts how solutions can be found. Its use to guide the search can prevent innovative solutions and hinder creativity, as it implies that all solutions must be build by a series of beneficial mutations. It is ill-suited for complex or deceptive problems, because it assumes that these problems can shed light on path towards their fulfillment. As this is not true, it causes the search to converge prematurely into local optima and its narrow focus on one task can potentially harm the generalization abilities of the learned solutions. The next section will present the approach taken by this thesis to alleviate these problems.

1.2. Objectives of this Thesis

To address the above mentioned shortcomings, the common approach is to supplement fitness-guided search with diversity maintenance techniques [20, 21, 22, 23, 24, 25] that ensure a variety of candidate solutions and encourage exploration. Yet in very big search spaces or very deceptive problem domains random exploration is not enough and the drive towards diversity needs to be directed towards useful or interesting structures. The direction in which to allocate search efforts in such cases would ideally be one of both local and global utility. One that it avoids getting stuck locally and benefits the search in the long run. With the problem of local optima being ubiquitous in AI, Different fields have proposed different ways to address this problem and this work picks up on an assumption shared across many of them: A drive towards a diversity of options does not only help to escape local optima, but can be beneficial on the global search scale as well, effectively acting as a general utility value to guide the search towards more complex behaviors. This suggests that instead of their resemblance to the objective, one could evaluate search directions by how much they contribute to increasing or preserving diversity.

While it is conceptually clear and widely accepted that increasing diversity is helpful in preventing convergence [22, 26], it is a rather new perspective that diversity can be worthwhile pursuing for its own sake, irrespective of any task. In relatively simple physical systems, Wissner-Gross & Freer [27] have shown that increasing an agent's future options leads to a range of adaptive behaviors that coarsely align with our intuitive concept of intelligence, without such being specified as a goal. Conceptually very related to this are works from reinforcement learning (RL) on *empowerment* [28, 29] that frame the diversity of options as the amount of influence an agent can exert over its environment such that it can perceive it via its sensors. Agents that maximized their empowerment were able to solve classic RL-tasks as for example pole-balancing [29], but also more sophisticated ones, such as avoiding a stream of spreading lava in a blockworld by building tunnels or drainage systems [30]. In the field of evolutionary robotics, Lehman's work on novelty search [1] first demonstrated the value of diversity as sole objective. By using a measure of behavioral sparsity, novelty search completely abandons the fitness function, instead selecting those individuals for reproduction that are most different from their peers. Without any notion of an objective, novelty search discovers behaviors that would have previously been specified as goals, such as bipedal walking or maze-navigation. This led to the counterintuitive insight that in

order to reach a goal one must not necessarily try to do so. The main reason why novelty search is successful is that collects information about the environment in a principled manner. After the simple behaviors have been explored, doing something new implies doing something more complex, often by adapting favorably to the affordances of the environment. Similarly, empowerment neglects the reward function and is defined by the agent's capacity to manipulate the environment. Logically, in absence of any task specification to inform the search, the environment is the only source to leverage information from. This suggests that the more interesting aspects of an individuals interaction with the environment are captured in the definition of the diversity objective, the more complex problems can be addressed.
In order to support this claim, I model three different exploratory search drives, which increasingly make use of the environment to define their measure of diversity: curiosity, novelty and evolvability. While all are concerned with increasing diversity, these objectives differ in the time-scale over which an individuals interaction with the environment is measured: Curiosity, defined as the diversity of the path an individual traverses during its lifetime, is limited to the present. Novelty is taking into account all behaviors found during the entire search, thus referring to past and present. The third objective, evolvability, refers to an individual's capacity to produce diverse offspring, thus increasing its chances of being able to adapt to future challenges. By biologists and EC practitioners alike, evolvability is seen as crucial in ensuring the potential for further evolution and has thus received increased attention over the last years [31]. By preventing over-specialization and ensuring future adaptability [32] it is acting in a role akin to regularization techniques in machine learning and has thus the potential to greatly enhance the reach of EAs. But as it requires many additional simulations to compute it exactly, evolvability does not lend itself well to be cast as objective. Therefore, this work investigates different ways it can be approximated without additional costs, trying to make use of the relatives in an individuals lineage and its offsprings that go extinct, both of which are usually discarded without further use.

Another commonality across the general-utility approaches from the above mentioned fields is that all of them feature the information theoretic quantity entropy in a prominent role. Mathematically describing the dispersion of a distribution, entropy can readily be adapted to measure different kinds of diversity. I show entropy's aptness for diversity based approaches by applying it on two different levels. On the level of the individual behaviors, I use it to model the above mentioned properties curiosity, novelty and evolvability with it, which I then apply to solve a deceptive maze-navigation tasks. Also, I then show that entropy can be applied at the population level, by using it to balance exploration vs. exploitation, or as it less commonly referred to diversification vs. intensification. Therefore, an archive is used to store the objective values produced during search. The entropy of the distribution captured by that archive can then be used to estimate which values are most likely to indicate search directions that preserve diversity. If, for example, the search treads on the spot because it is trapped in a local optimum, the respective objective values will accumulate and in turn be assigned a lower entropy value. As the frequency of those objectives values in the archive increases, the local optimum gets 'filled up' and the search progresses to other areas with different objective values. Like that, entropy can be used to interpret the frequency of an objective value as indicator of its interestingness. In order for this to work the degenerated behaviors, which are the vast majority because most mutations are deleterious, must be mapped to a small number of objective values. The practicality of that use of entropy has been demonstrated by Weise et al for fitness values [33] on a genetic programming problem and is here for the first time applied in ER and adapted to task-agnostic objective functions.

The main findings of this work are therefore (i) to reaffirm that diversity based objective functions outperform task-specific ones on deceptive problems, (ii) to illustrate that the degree with which a diversity objective captures the interaction with the environment contributes to its success and (iii) to show that entropy is well suited to model diversity-based objectives and select which of their values are favorable for the search progress.

These findings are then verified in a considerably harder problem domain: Evolving gaits for a bipedal robot. Albeit not significantly outperforming the objective based methods in that domain, the gaits evolved with entropy based methods are more natural and show a greater potential for further improvement.

A secondary contribution of this work comes from the implication of the failure to approximate evolvability from the discarded individuals. A possible conclusion of this is that different behaviorial niches have very different potentials for evolvability, i.e. an individuals lineage can spread out more or less depending on where in the environment it is. These results might be instructive for other researchers wanting to estimate evolvability, i.e. they could try to interpret the lineages in the context of the affordances of the local environment of an individual, and not compare them globally as I did.

1.3. Outline

This work is structured as follows: The next section will review the related work and highlight diversity's change of role from secondary to primary objective. Section 3 will describe the methodology of how the different diversity-objectives are modeled with entropy, explain the assumption behind the different possible heuristics for evolvability and how the maze-navigation task is set up. Section 4 describes the experiments, their results and their attempted verification in the biped domain. Section 6 is a conclusion that sums up the results of the different methods, discusses their limitations and possible directions for future research.

2. Related Work

This section will present related work concerning four different topics. Firstly, approaches from the fitness paradigm that aim to address more complex problems by making procedural adjustments or by shaping the fitness function to include a measure of diversity are analyzed. Secondly, more recent works presented that argue in favor of neglecting the notion of fitness for a diversity objective, which can lead the search to the same outcomes that fitness inspired approaches would have specified as goals. Works in the third section are concerned with evolvability and how it should be understood in order to be most beneficial in terms of future diversity. Lastly, I will present works that show the role of entropy in evolutionary computing and highlight that it is especially appropriate to measure diversity at different levels.

2.1. Modifying the Fitness Function

Usually, EAs use a fitness function modeled on the objective both to measure its fulfillment and guide the search towards it. This often results in deceptive fitness functions that make the search converge prematurely and get stuck in local optima. Approaches to remedy these shortcomings are numerous and take different angles. In [26], the authors suggest to classify methods in the literature into 'process helpers' and 'goal refiners'. Process helper methods address more complex problems by modifying the selection process, typically by breaking down the fitness function. In order to avoid this involvement by the experimenter, goal refiner approaches redefine the fitness function by including a measure of diversity, thus introducing a drive towards exploration that prevents the search from converging.

2.1.1. Process helpers

In an effort to avoid local optima and guide the search through the fitness landscape, staged evolutionary methods use increasingly complex evaluation conditions [26], for instance by gradually increasing the gravity of a simulation to make balancing harder. Similarly, incremental evolution techniques break [34] the goal down into sub-tasks or define a series of increasingly complex fitness functions. For example, [35] first evolve a neural oscillator, then use it to learn to walk. Instead of splitting up the fitness function, fitness aggregation methods, as for example [36], reward intermediate behaviors by aggregating subtasks into a single fitness function. That way, the intervention of the experimenter is reduced to tuning the weights of the aggregated fitness function, but the method still implies that all subtasks are obligatory in order to reach the best performance. The drawback of all these approaches is that they rely on prior knowledge of the experimenter how high fitness shall be reached, which is opposite of the spirit of evolutionary search. Consequently, no really innovative solutions can be found and the search will not conclude if the hints given by the experimenter are misleading.

2.1.2. Goal Refiners

Both local optima and premature convergence are effectively consequences of a lack of exploration. Therefore, introducing diversity maintenance methods into the search process is the typical methodology utilized taken to address them without relying on task-specific knowledge. A general methodology to affect diversity is to increase the population size, yet this choice is not trivial and increasing the population size is not always helpful [37, 38]. When crossover and recombination are used as search operators to spawn new individuals, there can be a loss of lineage diversity, i.e. all individuals soon descend from only one or few ancestors [37]. Burke et al [39] address this by modifying the selection process to sample from different lineages and then compare their fitness values. This increases diversity, but does so at the cost of reintroducing potentially outdated individuals in the population. Being still closer to how evolution happens in the real world, where there is no overall objective but individuals instead try to outdo one another, co-evolutionary methods [40] define fitness with respect to other individuals in the population. Then, the fitness function is relative to an individuals peers instead of being static. In competitive co-evolution the fitness of the individuals in one species is defined with respect to individuals of another species, for example in predator-prey scenarios [41]. Though such methods can potentially result in a never-ending 'evolutionary arms race', the progress of such methods is hard to track and they often turn into cycles of behaviors that are discovered, forgotten and re-discovered.

As already Darwin suggested, the competition in nature is fiercest between those individuals most closely related [42]. In evolutionary computing (EC), this notion is taken up by fitness sharing methods. Fitness sharing requires to partition the population into different species wherein the fitness of the individuals is shared. This prevents overcrowding of one niche, so that no one species can take over the population and cause the search process to converge. Different distance measures have been proposed to split the individuals into species or niches: Fitness uniform optimization [24], groups the population according to the fitness values, thus keeping a way out of a local maximum. One approach of particular interest to this work, whose principle I later adjust to other objectives, is fitness frequency assignment [33]. It leaves the computation of the fitness objective unchanged, but then uses it to assign an utility value to a solution according to how often other solutions with similar fitness values have already been discovered during the entire search. Hence, it groups individuals into species based on their fitness and 'fills' local optima up when the search is stalled, naturally swinging back and forth between exploration and exploitation.

Task-agnostic approaches have divided the population based on distances of their encoding (genotypes) [23] or their age [25]. But as genomes are only translated very indirectly into behavior, such approaches do not guarantee functional diversity in the population. Measuring the distance of the decoded genomes alleviates this problem but can be impractical in the case of ER, where those are often graphs (e.g. ANNs) for which computing distances is an NP-hard problem[43]. Recent approaches have therefore taken to measure diversity in the space where it ultimately matters: the space of behaviors. Such behavior based speciation [21] and diversity measures [44, 17] require special metrics to the measure distance between behaviors. In [1], Lehman has shown that relatively simple, intuitive measures work well and other works have shown that generic distances, such as information theoretic quantities [45, 46] or hamming distances [22] work as well. Comparative studies [22, 47, 5] have shown that these behavioral diversity measures outperform genetic ones.

The two previous sections highlighted that avoiding convergence by including prior knowledge into the fitness function limits the scope of what evolutionary search methods can achieve, diversity maintenance methods are best to avoid convergence and that diversity is best encouraged in the space of behaviors. The next section presents works that depart further from the fitness paradigm by promoting diversity not as an objective secondary to fitness, but as an objective of its own right.

2.2. Diversity as Objective in ER

Works in the previous section have recognized the importance of diversity in order to escape local optima, but at their core are still guided by the fitness function. This section will present works from ER that pursue diversity for its own sake, often leading to similar or better results than their task-specific counterparts.
Therefore I first present works that are solely driven by diversity, increasingly using more information from the environment, and then progress to their adaptations made possible by the introduction of multi-objective EAs. The latter allows parallel optimization of multiple objectives and as such are a means of increasing diversity by themselves. Their introduction to ER alleviated the problem of having to choose between fitness and diversity and enabled to investigate how different, possibly contradicting features of a problem interact (for instance the size of the robot vs. energy consumption) and vary in their potential for fitness. With those works the role of diversity shifts from ensuring that one high fitness can be reached, to helping increase the experimenter's understanding of the problem.

2.2.1. The Curiosity Instinct

A price paid for the convenience of simulating many generations on the computer is that solutions derived in simulation sometimes drastically decrease in performance when they are ported onto a real robot. In order to avoid that

and ground the robot in the real-world, a sub-branch of ER called *embedded ER* aims to develop evolutionary methods that can be applied on board of an actual robot, possibly without any communication to an external source.
One such approach that disregards the objective and instead leverages information from the environment is the 'curiosity instinct' [45], which refers to the diversity the robot experiences during its lifetime. Therefore the experience of the robot, as captured by its sensorimotor stream, is partitioned into states with a clustering algorithm. Curiosity is then measured as the entropy of the trajectory of the robot, discretized by those states. The assumption is that interesting controllers lead to high-diversity sensorimotor experiences. A second drive proposed in this work is the 'discovery instinct', which is similarly defined as the entropy of the robot's path, but with respect to the entire sensorimotor stream of the robot's lineage. This assumes that although there is no communication between individuals within the same population, parents can pass on their experience when they are spawning children. Both drives have been used to solve deceptive mazes, which are a common benchmark for diversity objectives [48, 1, 44, 49].
The next section presents novelty search, which uses more information by defining its measure of diversity by means of the behavior of the robot and the entire history of robotkind.

2.2.2. Deceptive Problems and Novelty Search

Works on novelty search have been motivated by the realization that most complex problems are deceptive. Whereas many notions of problem hardness exist, often based on properties of fitness landscape, such as its ruggedness [50] or the extend of neutral plateaus in it [51], the initial work on novelty [1] adopts a more intuitive description. There, a problem is said to be *deceptive* if an EA that follows the gradient of an objective will be lead into a local optimum More generally speaking, deceptive problems are problems where the steps towards the solution do not resemble it. Especially very ambitious objectives can not necessarily be expected to illuminate a path to themselves. Then the heuristic of fitness becomes uninformative, or even harmful[52]. In perversely deceptive problems, the fitness will lead into the wrong direction at most steps during the way. If, for example, the task is to find a path through a circuitive maze, a fitness function that measures the bird's eye distance to the exit, as it is typical for maze navigation tasks, will effectively oppose the right path at least 50% of the time, probably much more. In such cases, it becomes a liability rather than a useful heuristic for finding the goal [48].

Novelty search [1] is immune to the problem of deceptive fitness functions as it discards the notion of fitness completely. Instead it uses a measure for phenotypic diversity that favors those individuals that behave differently from their peers. A selection of previously encountered behaviors is kept in an archive and used to represent where the search has already been. An individual's novelty score is then computed as its distance to its k nearest neighbors within the current population and the archive. Thus novelty measures the sparsity of the neighborhood of an individual's behavior, using results from both the past and present of the search. As novelty is a dynamic objective, defined with respect to other individuals, novelty search is a divergent search method and does not suffer from convergence. Also, as the objective of what is novel is constantly changing, it never ceases to be informative and is thus suited for open-ended search. Interestingly, novelty search often outperforms fitness-guided search [48]. On its first introduction [1], it solved tasks where fitness-guided search fails and its solutions were found faster. Also, when the evolved controllers were ANNs, they were more elegant, i.e. they were more compact, with less connections than those of their counterparts evolved by fitness-guided search. This unnecessary complexity of the ANNs evolved by fitness-guided search can be seen as evidence that the requirement of having every step along way the being an improvement does actually hinder the discovery of simple and elegant solutions.

Novelty search has been successfully applied to different tasks, such as maze-navigation [1], evolving gaits for a bipedal walking robot [53], evolving different topologies and controllers for walking robots [9] and evolving controllers for homogeneous [54] and heterogeneous [55] robot swarms. On the maze-navigation task, it outperforms the curiosity and the discovery instinct [45], supposedly due to its use of ground-truth information and its more extensive use of past search results. Moreover, it has been found that novelty search can evolve plastic ANNs that are capable of learning to adapt to a reward signal [56]. This task, 'learning to learn', is highly deceptive as it is almost always easier in the short term to learn a policy that exploits statistical regularities or certain specifics inherent to the domain than to really learn to adapt to the actual dynamics of the environment.

Novelty search is not a random exploration, because it collapses different genomes and different ANNs onto the same point in the search space, unless they lead to a different behavior. Rather it presents a systematic way to explore the structure of the environment: Although it might be easy to discover a new way of behaving in the beginning, once all simple behaviors

are discovered, doing something new requires doing something more complex, often by adapting favorably to the affordances and dynamics of the environment.
Whereas the fitness landscape of complex problems is often rugged and irregular, the one searched by novelty is a lattice of connected nodes with increasing complexity. Nevertheless, such a lattice can include novelty plateaus, areas when the intermediate mutations a new behavior would require to not produce novel behaviors itself. In [49] Li, Storie & Clune suggest to also reward new fire patterns in the neurons of networks. Their approach, dubbed 'creative thinking', improves over simply searching novelty in a deceptive maze domain and a sequential ball-collection task.

While being able to solve a given task, novelty search does not finetune a found solution, as there is a constant pressure to innovate and the discovered behaviors effectively act as repellents for further search. This drawback has been addressed by the introduction of multi-objective search algorithms, which will be presented in the next section.

2.2.3. Multi Objective Search

Multi-objective evolutionary algorithms (MOEAs) [57] allow to rank solutions by more than one objective, thus keeping a population of diverse trade-offs between the objectives. Relying on the concept of pareto-dominance (see 3.5.1), some individuals may maximize one objective at the expanse of all others and others might find the best trade-off between a subset of objectives. This allows to simultaneously optimize competing objectives, such as performance and cost or diversity and fitness. Other than aggregate fitness functions, which require to finetune the weights of the individual objectives, there is no need to rank the objectives by their importance. MOEAs can also be used for single-objective optimization, additionally supplemented with helper objectives that provide a natural way out of local optima if the main objective seizes to be informative. They permit task decomposition without implying an order in which the tasks have to be solved or making any of them mandatory, as for example demonstrated on a light switching task in [17]. Disadvantages are that they can still suffer from deception [52], adding objectives is not always helpful [58] and there can only be added about three extra objectives [59] so that the typically used pareto-dominance principle that compares solutions stays meaningful. Algorithms that deal with more than four objectives are referred to as many-objective optimization methods [59]. Another issue from

which MOEAs can suffer is the so-called domino-convergence [26], when the optimization of one objective happens at a much faster rate than another, so that they effectively happen sequentially. If that is the case, then the values that slower converging objective reaches are determined by the value of the other objective, which converged faster.
MOEAs have been applied in ER with great success. [44] use a MOEA to combine fitness with a measure of behavioral diversity, which they measure as an individual's average behavioral distance to the rest of the population. Searches with this additional diversity objective outperformed classical fitness based approaches. Therefore this method will also be used as control method to compare the criteria developed in this work. Especially for novelty search, the possibility of using a second objective has led to many new variants. In order to allow finetuning discovered behaviors to their full potential, various works combine novelty with fitness [60, 61, 47, 48]. Their result is that this combination is more effective than pure novelty or fitness based search. Another approach that makes use of a MOEA with the novelty criterion is based on the idea that natural evolution finds many ways to solve the same problem of reproduction and survival. In minimal criteria novelty search [62], all individuals need to meet the same minimal fitness criteria, which effectively prunes the behavior search space so that solutions are found more consistently. Extending this idea further, progressive minimal criteria novelty search [63] also freely explores the behavior space as long as the individuals meet a fitness criterion that becomes stricter as the search progresses.

While the algorithms mentioned up to now do not include a notion of a task to be solved, they are still compared by their maximum performance on it. The outcome of one evolutionary run is thus measured by the success of its best individual. This is still very different from natural evolution that produces many species in one run, each of which successful in a different way. This is not attempted in this thesis, but for a more complete picture of diversity in ER, it is instructive to review algorithms that aim to do that. Especially, as those algorithms shed light on how different ways to characterize behaviors have different potentials for reaching high fitness solutions and how a problem can be solved by exploring how different aspects of it are connected.

2.2.4. Quality Diversity

The above mentioned novelty-hybrid approaches add some biological plausibility to EAs but are still a far shot off natural evolution. Although novelty is computed locally by comparing an individual to its nearest neighbors, the solutions are compared globally on the fitness objective. Yet, as it is put in [9], it does not make sense to compare grasshoppers with bears on speed for they live in completely different environments. Such global fitness competition will drive the algorithm towards those morphologies that are easiest to exploit. Addressing this, the *novelty and local competition* approach [9] explores a feature space with novelty search and computes an individual's fitness only relative to those most similar to it. Consequently, the algorithm rewards individuals that discover new niches to escape competition, mimicking the limited interaction between animals of different niches in nature. The approach is used to evolve different morphologies for walking robots and evolves a variety of functional walkers with different combination of features (size and weight) in one single run. At the cost of finding one potentially global optimum, the algorithm finds many different locally optimal structures, increasing the experimenter's understanding of how the explored features are related.

This inspired the authors of [64] to coin the term 'illumination algorithm'. Other than multi-objective optimization algorithms, which aim to find just the high-performing individuals for a combination of objectives, illumination algorithms [65, 64, 66] aim to illuminate the fitness potential of the entire search space. Another, recent work [15, 14] dubbed this type of algorithms *Quality Diversity* algorithms (QD), as they aim to discover a diverse range of high-quality solutions.

First introduced by Mouret, Clune and Cully [64, 67], *MAP-elites* (multidimensional archive of phenotypic elites) is the most widely applied QD-algorithm. It is parallelizable and conceptually very simple so that it lends itself to various adaptations. It discretizes a feature space into different bins, for each of which it tries to find the highest performing individual in terms of fitness. The feature space can be chosen such that it describes the behavior of the robot [15] or any other aspect of the problem that is of interest to the experimenter. Their choice can thus be motivated by their importance for solving the problem or simply by the interest of the experimenter. An example of the former approach is given in [64], where the features are an ANN's total connection costs and its modularity. Here the underlying assumption was that networks with lower connection costs tend to be more modular, which makes them more performant and also favors evolvability [65]. In that way, a two-dimensional feature space (connection costs and modularity) was used to explore the much higher dimensional solution space (morphology and parameters of the ANN). An example without any preconceived ideas about how the features relate to performance is when the experimenter wants to explore the interplay of different morphology properties, such as the number of legs, the body-size or the percentage of bone and muscle tissue [64]. Other adaptations of features allowed solving very complex mazes [15], finding diverse walking strategies for the same robot morphology so that it could adapt to damages [67], generating 2D-images to fool deep neural networks [68], or leveraging their internal representations as feedback for a creative process that produces new 3D printable objects [69].

Interestingly, for many tasks the best individuals found by QD-algorithms were better than those produced by the algorithms that just searched for one optimal solution. This indicates that the heuristic 'finding better solutions by making random changes to good solutions', which all evolutionary algorithms generally adopt, must not be true for complex problems. The authors of [64] confirmed that idea by mapping the lineage of the highest performing solutions and found that those individuals often traverse a long path through very different areas of the search space, suggesting that the stepping stones necessary to perform high in one area, must be developed partly elsewhere.

Another insight relevant for this work is given by Pugh et al in [15], where they show that QD-algorithms only lead to good solutions if the behavioral characterization (BC) that constitutes the feature space is aligned with the notion of quality, i.e. "finding novelty tends also to lead to higher fitness". Extending this insight in [14], they identify two factors that are indicative of a high degree of alignment: (i) the maximum fitness values in neighboring regions of the feature space must correlate and (ii) each behavior must only describe a narrow range of fitness values. If, for example, the task were to navigate a maze, an example for a BC with high alignment is the individuals position at the end of the simulation, as it indicates the exact distance towards the exit of a maze. If instead the used position were sampled earlier, for example already three seconds before the end, it would be less aligned with fitness, as the robot can still change its distance to the exit. Consequently, BCs with high alignments are needed to be successful in deceptive domains [15]. The latest adaptation of MAP-elites [14] allows to optimize diversity with multiple BCs, an aligned one in order to overcome deception and a second one that just represents a BC of interest to the experimenter. [1]

[1] As the major works on quality diversity and behavioral characterizations were published between July 2015 and July 2016, parallel to the making of this thesis, their insights were not considered in the design of the experiments. But as their methodology and perspective on the behavioral characterizations are helpful in understanding the role of diversity and the outcome of my experiments, they are stated here.

As they were primarily designed to inform the experimenter about the fitness potential of the entire feature space, QD-algorithms assign less resources to the most promising areas of the search space. The fact that their best solutions are still better than those found by optimization algorithms shows that searching simultaneously for many solutions is more effective than limiting the search to one objective. The solution to one problem (for example finding the fastest robot with a very small weight-height quotient) may depend on stepping stones from related regions (controlling a robot with a medium quotient). This highlights the importance of maintaining diversity and gives an intuition on what the BC used to drive diversity must include to be helpful.

But to maintain a diversity, it first has to be produced. If one-step mutations do not produce sufficient behavioral changes, then judging progress on any objective, fitness or diversity, becomes much harder. Therefore the next section will present works from ER about how the propensity of EAs to produce diversity can be enhanced. Furthermore, works from related fields that show that increasing future diversity can lead to even more complex behaviors than those currently in the reach or ER.

2.3. Evolvability & Future Diversity

A trademark feature of natural evolution is its prolific creativity and ongoing innovation. One key feature to achieve that is evolvability - an individual's or a system's capacity to evolve. The interest in it, both of researchers from biology and EC, has greatly increased over the last years [31] and there is a disagreement in the communities what evolvability actually is. The first part of this section will therefore present the different perspectives on evolvability and discuss some theories about how high evolvability in nature comes about. This latter issue will only be discussed superficially as it touches upon the fundamental question of why there is an increase of complexity in natural evolution in the first place - a topic that is hotly debated with very little consensus [70, 71].

The second part presents how evolvability and its two main interpretations are measured in EC and how the evolutionary search procedure has been adapted in order to passively or actively promote it. The third part will then present work from related fields that use the diversity of future options without the constraining framework of EC. Their conceptual relatedness and their impressive results show the value of evolvability and motivate the attempt to find a heuristic for it.

2.3.1. Interpretations and Effects of Evolvability in Biology

In the current literature, there is no one way evolvability is defined, [72, 73] but two common interpretations :

1. Evolvability refers to the genetic or phenotypic variety the population can produce that can be affected by natural selection, irrespective of the usefulness of that diversity for further adaptation [74].

2. In its second sense, evolvability refers to the ability to generate adaptive variation. Other than (1) that is variation that helps to survive and reproduce over the mid and long term. [75]. This second interpretation ties evolvability to the ability to acquire novel functions [73], or even breakthrough innovations [76], which is more powerful and includes the first interpretation.

But whether or not one assumes the variation to be adaptive, evolvability can be beneficial for evolutionary progress in different ways: A lineage that is evolvable might need less mutations to produce novel traits, adapt faster to changing environments or environmental pressures and, as Kirschner shows in [74], reduce the potential lethality of mutations. Other works [72, 65] suggest a connection between evolvability and modularity, which also is favorable for quick adaptations. Another latent contributor to high evolvability might be robustness. In [77], Wagner suggests that an individual's robustness to neutral mutations might also contribute to its evolvability, as mutations neutral in one environment, might be adaptive in another one.
Although these properties are desirable for the process of evolution, it is questionable whether natural evolution actually selects for high evolvability, or as Pigliucci asks: ”Is evolvability evolvable?” [72].
The path of evolution is governed by mutation, selection, and genetic drift. Mutations are random and new variants are only stabilized into a population by selection. Selection has no foresight and consequently no reason to stabilize traits that provide future benefits, so that it might in fact sometimes be harmful to evolvability [31]. In [78], Lehman shows that evolvability can be a byproduct of genetic drift and nature's tendency to accumulate niches. Evolvable lineages are able to found and populate more niches. An effect that can be enhanced by extinction events that act as evolutionary bottlenecks [79].

Evolvable lineages have better chances to survive those extinction events, as they are inhabit more niches. By randomly eradicating the majority of niches, extinction events then free them from all established competition that might previously have prevented an evolvable species to migrate. Afterwards, the more evolvable survivors are faster at repopulating niches that were emptied out. Day [80] supports this idea showing that genetic diversity does indeed increase the colonization success of populations. Hence, although there might be no immediate selection for high evolvability at an individual level, it can occur at the level of species.

The next subsection will show how the different connections between evolvability and evolutionary progress have been adapted and favored in EC.

2.3.2. Measuring and Encouraging Evolvability in EC

Given that EC has yet fallen short of its aspirations of solving really complex problems or producing open-ended innovation that rivals that of natural evolution or human designers, measuring and ensuring the potential for further evolution is a concern for researchers throughout all branches in EC. The works in this paragraph show that evolvability can be a concern at almost every decision throughout the design of an evolutionary search.

Other than in nature, the way individuals are encoded on a genome is not fixed, but depends on the experimenter's choice of encoding. Therefore, representations have different potentials for evolvability [75], according to how much they facilitate the discovery of new viable phenotypes. Reisinger and Miikulainen [81] propose a test to measure the evolvability of a particular representation, separating it from the selection operators and the task to be solved. Similarly, [82] gauge different encoding by their mutation operator's potential for evolvability with evolvability signatures. Their evolvability signatures are a combination of the average change in fitness after a mutation and the quantity of behavioral variation they generate, thus encompassing both interpretations of evolvability. In [80] Day also takes into consideration the strength of selection. If higher selection pressure is needed to produce variation, the evolvability is lower. Consequently, evolvability is seen as change in genetic diversity per unit strength of selection. He further derives that when no assumptions about selective strength are made, Shannon's information entropy of the genetic variation is the appropriate choice to measure genetic diversity.

The other choice of the experimenter is the fitness function. Here, evolvability can be favored indirectly or directly. Indirect approaches rely on assumptions as how evolvability comes about, such as by using changing environments [83] or fitness functions that provide many objectives [84, 85], emulating the high number of niches in nature. Based on the connection between modularity and evolvability, [86] encourages the discovery of modular solutions, by using modularly varying goals. Similarly, [65] encourages modularity by punishing connection costs, which lead to faster adaptation in new environments. Another indirect way to favor evolvability is shown by divergent search processes [1, 44, 46, 45], such as novelty search. By defining a moving target, such processes favor lineages that are able to constantly produce novel phenotypes - and are more likely to be evolvable [87, 1]. Noteworthy, the finding that novelty favors evolvability adopts it in its first sense, measuring it as an individual's capacity to produce many phenotypically distinct offspring.

If, as intended in this work, the evolvability measurements are to be used to influence the selection procedure, measuring it becomes more challenging. In order to be useful for differentiating between the evolvability of individuals within the same population the measurements can not be done in retrospect, but have to be available after each generation. In the related field of genetic programming, where the individuals are computer code, approaches [88, 89] often opt to define evolvability in its second sense, as the propensity to produce fitter offspring, which must be inferred by doing extra simulations. Notably, [88] show that one can do some of the computational work before starting the evolution. They use machine learning techniques to build an a-priori model to predict an individuals chance to produce a fitter offspring than itself. Later during evolution they use this model to influence the selection process, leading to an overall improved fitness. Disadvantages of their approach are that the model can not be adjusted throughout the search, is still computationally expensive and that their approach depends largely on the definition of fitness, which can be misleading.

In [90], Wilder and Stanley make another useful distinction that helps to classify the different approaches aiming favor evolvability. They argue that adaptive methods, such as using varying fitness functions or environments are helpful in increasing the evolvability of individuals, whereas non-adaptive processes (e.g. novelty search) increase the evolvability of the population. While the former are beneficial for the offspring of an individual lineage, the latter is beneficial for the search process in its entirety. They observe that processes that are good for one kind of evolvability can hurt the other and vice versa. Furthermore, they argue that non-adaptive methods, such as divergent processes, act at the level on which evolution ultimately is defined, the population level, and consequently are more important. Thus, this perspective is useful to reconcile some misunderstandings due to conflicting views on how to favor evolvability and motivate research to find more non-adaptive mechanisms. Yet, a more recent work gives a reason why this might not be enough and why promoting individual evolvability should be pursued as well.

In [91], Mengistu et al show that novelty and objective based search are capped in their potential for evolvability. In a maze navigation domain, the maximum evolvability reached by novelty search and objective based search stagnated well below what was reached when evolvability was directly selected for. When selecting for novelty or fitness, even numerous extra simulations did not raise evolvability any further, indicating that some other incentive is missing to produce complex and very adaptable individuals. Their algorithm, called *evolvability search* replaces the novelty objective with a measure of how many distinct offspring an individual can produce. Due to the extensive cost of simulating many extra individuals to compute evolvability, the algorithm was not designed to be of practical use, but rather as a tool for facilitating the study and understanding of evolvability. One insight from that algorithm is that evolvability by itself can be used to maintain diversity and to solve deceptive problems, the latter being demonstrated with deceptive mazes a bipedal robot. Moreover, individuals selected for evolvability generalized better to other mazes than those produced by novelty. This was true independent of whether or not the individuals were used as starting points for further evolution or applied immediately. This result dovetails with a previous work by Kouvaris [32], that casts evolvability as evolutions generalization skill - an analogy they support by showing that regularization techniques known to alleviate overfitting in machine learning, such as penalizing the complexity of the model and introducing random noise, can be used to enhance the evolvability of the learned solutions.

Conclusively, evolvability will play a crucial role when one wants to extend the reach of ER to more complex problems. It can maintain and increase phenotypic diversity, speed up the evolutionary process and ensure that the learned solutions are not overly specialized so that they can still adapt to a wider range of future problems.

To illustrate the value of the currently unattained goal of efficiently maximizing evolvability during selection, the next section presents works from related fields that similarly quantify the diversity of future options, but outside of the constraining framework of EC.

2.3.3. Future Options as General Utility Value

This section presents different approaches from reinforcement learning, physics and swarm intelligence that try to define locally useful, task-independent criteria based on future diversity.

A related AI-domain, which also faces the problem of a gauging the contribution of a local step to achieving a distant goal is reinforcement learning [92]. In reinforcement learning (RL) an agent needs to learn a behavior solely based on a qualitative reward signal, telling it whether an action was rather good or rather bad. As this reward signal is only given upon completion of the task, the robot's learning algorithm must decide which of the actions it took were in fact critical for achieving the goal - a dilemma referred to as the delayed-reward and credit-assignment problem. Similar to ER, RL-problems are of a composite nature so that achieving the final goal (for instance opening a door) requires many previous actions that are not apparently related (grabbing the handle of the door, pushing it down, etc.). Many of those intermediate actions merit by themselves to be learned, irrespective of whether the experimenter wants the robot to go through the door or not. Therefore, an alternate approach to using a distant objective is to define an internal reward mechanism that drives the robot's exploration. Within the paradigm of external reward functions, exploration is usually defined as not trying to solve the goal but doing something else, possibly at random. Yet exploring a space at random quickly becomes impracticable as the number of degrees of freedom increases. To avoid the curse of dimensionality inherent to such random exploration, the homeokinesis approach [93] suggests to guide the exploration by learning an internal model of the world. With the sole obligation to keep moving and interpret the world in terms of its model, the robot then explores the environment in a principled manner. In order to improve its model, the robots must limit its exploration to parts of the environment where it can more or less predict the outcome of its actions from past experiences. In the initial work [93] interesting and seemingly goal directed behaviors such as obstacle avoidance, ball collecting and maze navigation emerged as a byproduct of that goal-less internal drive. Later works [94, 95] extend that approach to more complex robotic structures, such as an armband robot consisting of linked elements which develops spontaneous cooperation with decentralized control.

Another internal motivation, whose authors also hypothesizes the existence of a simultaneously local and global optimality criterion, is empowerment [96].

Based on the idea that humans perceive the objects in the world based on the affordances they provide to them, empowerment describes the degree of control an individual can exert over its environment in a way that he becomes aware of it via its sensors. Therefore, empowerment refers to the maximum information an individual can imprint onto his environment such that it can be perceived by its sensors. Maximizing empowerment has been used to evolve efficient sensors [97] but also to drive an agent's behavior in pole-balancing tasks, maze domains [96], collision-avoidance arenas [98] or when interacting with block worlds [30]. In the latter example, the agent had perfect knowledge about the effects of its actions and discovered different ways to avoid a stream of spreading lava, including tunnels, bridges or drainage systems. Noteworthy, all those behaviors were discovered without being set out as goals, but only from the agents drive to stay in control and to preserve its options.

In board games, such as othello or checkers, a general heuristic for the quality of a situation on the board is *mobility*: an estimate of how many other states can be reached from it. Game players that maximize their mobility seek out states where they are in control of the game. Players who are losing typically do not have many options left (e.g. check-mate).

In a physically motivated approach, Wissner-Gross and Freer [27] have shown that an agent whose sole goal is to maintain a diverse range of future options surprisingly shows a range of adaptive behaviors that align coarsely with the intuitive concept of intelligence. These behaviors included something alike to tool-use and social cooperation between two agents, two behaviors that require cognitive functions that are often considered to be trademark features of intelligence. However, the approach used required perfect knowledge of the system's dynamics and simulating all possible futures, which is unfeasible in real world applications and even in simulation quickly becomes computationally intractable with increasing degrees of freedom. But the idea that preserving a diverse range of future options leads to self-organized spontaneous actions has proved to be of value not only for individuals such as in [27, 96], but also for a group of individuals as a whole: Garnett & Mann [99] show that assuming that a swarm's future is maximally uncertain (thus preserving as many options as possible) explained the emergence of cohesive decision making and could be used to derive predictions about social interaction rules for individuals in swarms that are consistent with other studies.

Although conceptually not applicable to ER, these approaches show that ensuring diversity, especially with respect to future options, can act as a general-utility value. For both individual and populations, the sole objective of maximizing future options can lead to many different sophisticated behaviors. Moreover, the complexity of the discovered behaviors increases with the amount of knowledge about the interaction of the agent with the environment. Especially the works on empowerment [28] and causal entropic forces [27] show that if the agent understands the limits and affordances the environment imposes upon him, a course towards maximum future options can lead to impressive behaviors.

2.4. Entropy as Organizing Principle

This section will briefly introduce the origins of entropy, point out its general properties and then show how it is used in ER and related fields.

2.4.1. Background

Initially, entropy was introduced by Rudolf Clausius as a measure of a thermodynamical systems' change of state when energy was added to it by a reversible process. Later Ludwig Boltzmann reformulated this in a probabilistic way, using entropy to describe a systems' internal disorder. For physical systems, maximum entropy is associated with their equilibrium state, towards which a natural system will spontaneously shift according to the second law of thermodynamics. In 1948, it was reinterpreted in new way by Claude Shannon to found the field of information theory, which it served to measure the expected information contained in a message and the minimum length of a lossless encoding. Its basic form, as used in probability theory statistics or information theory, is depicted in equation 2.1.

$$H = \sum_{i} p_i * log(p_i) \tag{2.1}$$

Without any meaning attached to the individual events, this definition describes the dispersion of a discrete probability distribution. The total sum is bigger the more equiprobable the individual events are that make up the distribution - the distribution with maximum entropy being the uniform distribution. For the individual events of the distribution a low p_i leads to a high entropy, and the respective events are said to be surprising or information rich. The generality of 2.1 allows it to be used in many different contexts and the next section will show that it is prominently featured in many approaches that are concerned with maintaining and promoting diversity.

2.4.2. Entropy in EC

Several works presented in the previous section rely on entropy to measure diversity. Both the curiosity- and the discovery instinct [45] measure the diversity of the states visited by the individuals with entropy. The other internal drive from RL, empowerment [28], uses entropy to measure the channel capacity between the agent's actuators to its sensors. Also, entropy is used to measure the diversity of the future paths, both for an individual agent [27] or a swarm [99]. Except for the last work, the use of entropy is restricted to the actions of the individual.

When defined with respect to the entire population, entropy can be used to assess the overall diversity present in the population as well as an individuals contribution to that diversity. Generally a high population entropy signifies that there is lot of diversity upon which selection can act. In cases of low population entropy, the individuals are clustered closely together, so that evolution degrades to random search. Day [80] states that measured upon the distribution of genomes, it refers to the genetic diversity in the population and can be used as an approximation of evolvability. Rosca [100] used the distribution of fitness values as a feedback to adjust the selection pressure. Liu et al [101] do this for a genetic algorithm approximating a set of commonly used benchmark functions, effectively balancing exploration vs. exploitation and leading to higher fitness and better convergence rates. In [102], population entropy is used to adjust the crossover rate and the mutation strengths for a GA that addressed the problem of frequency reuse of mobile bandwidth providers.

In [33], Weise et al define the scope of entropy even wider by using an archive to accumulate the distribution of fitness values over time, thus using not only the present but also past values to assess the state of the search. Importantly, they do not use the fitness value for selection, but the frequency (measured as entropy) with which it has occurred during search. Their *fitness frequency assignment* (FFA) approach is able to autonomously oscillate back and forth between exploration and exploitation. As searches in local optima will increasingly produce similar fitness values, their entropy score will quickly degenerate and drive the search away from that area, increasing exploration. In order to do exploitation, it relies on the idea that bad solutions are found more readily than the good ones, thereby using the frequency with which an objective value is found as indicator of its interestingness. As the authors note, this might not be the case for some problems, such as open mazes. Yet it makes use of one otherwise rather negative characteristic of evolutionary search methods: The search operations of EAs are random mutations and therefore most of the time destructive and result in degenerate behaviors, Instead of following a fitness trend, irrespective of the range of improvement, FFA's archive allows to follow a trend in the frequency landscape. This approach does not assume that a better objective value lies closer to the global optimum, which makes it suitable to address deceptive problems.

Summarizing, entropy can capture diversity with respect to an individual's interaction with the environment, the diversity in the present and past population and the progress of the search as a whole. The next section will present how I use it to model different search drives and the experimental setup I use to evaluate their effectiveness.

3. Methodology

To evaluate the success of the different objectives I use a maze-navigation domain and a bipedal walking robot. Mazes make it easy to design problems with varying degrees of deception, which is why they are a widely used benchmark task for behavioral diversity methods [1, 91, 45, 49, 5]. Evolving controllers for walking robots is a popular problem within EC [35, 87, 9, 91] and is considered to be much more difficult than the maze navigation problem due to the need for balance and the nonlinear dynamics of walking. Because of its higher computational needs, I will not be able to test all combinations of objectives in both domains. Instead, I use the maze domain to investigate the interactions of the proposed objectives and try to reproduce the most interesting findings in the biped domain.

3.1. The Mazeworld Setup

Mazes can be seen as metaphors for real-world problems where the fitness function is deceptive. In this analogy, the endpoint of the maze represents the solution to the problem and the walls around and within the maze are constraints. As these are not known a-priori, fitness in maze domains is typi-

cally measured as the bird's eye distance from the final position of the robot to the goal. Thus, the fitness function returns the euclidean distance between the final position of the robot and the goal. As this disregards the walls of the maze, which in the metaphor used above represent the structure of the problem, this often leads the search into a dead-end, the abstractions of local optima in the search space. The more distance the robot needs to travel in directions away from the goal, the more deceptive the problem is. The mazes used in this work (see figure 3.1) were initially proposed in [103], adopted for novelty search by Lehman & Stanley [1] and subsequently used in other works on behavioral diversity [45, 49, 91]. The code for the maze-experiments was taken from Joel Lehman's publicly available GitHub-profile [104]. My adaptations can be found at [105].
In order to demonstrate the objectives' capacity to escape local optima, the mazes increase in their degree of deception: The medium one requires a weaving motion and the harder of the two mazes has a very pronounced local optima in the lower left corner right near the start. With the sides of the mazes being 300 units long, the mazes are considered solved if the robot is within 5 units of the goal.

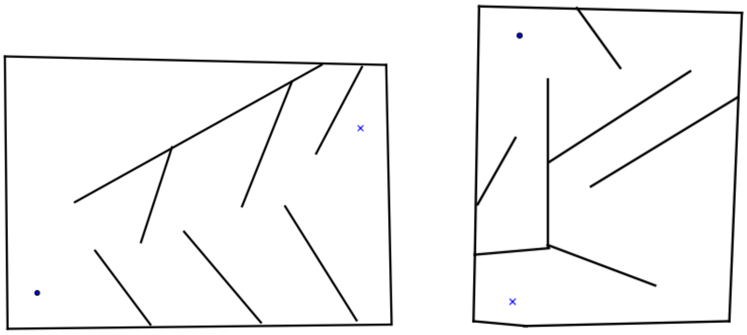


Figure 3.1.: Two of the mazes used in this work. The circle marks the starting point of the robot and x marks the goal. The medium maze (left) requires a weaving motion. The hard maze (right) has a very deceptive dead-end close to the goal in the lower left corner.

The robot is equipped with six sensors that detect the distance to the walls and four pie-slice radar sensors that indicate in which quadrant of the robot is the direction to the goal. Figure 3.2 shows the arrangement of the sensors. The ANN that controls the robot receives those sensor values as inputs and maps them to two output units, controlling the robots angular and linear speed.

In the initial ANNs all sensor neurons are fully connected to the output units. Over the course of generations the topology of the hidden layers is evolved with the NEAT method [106], tuning the weights and adding hidden

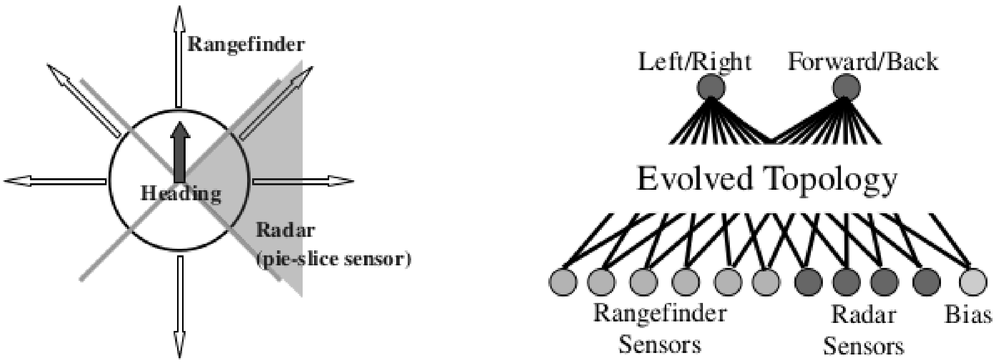


Figure 3.2.: Left: Each robot is equipped with 6 rangefinder and 4 pie-slice sensors. The readings of those sensors are the input to the neural network. Right: The inner topology of the neural network is evolved throughout the generations. The output of the ANN are two values controlling the linear speed and angular velocity of the robot.(Images with courtesy of Joel Lehman, taken from [1])

nodes and connections as the result of mutations. Each individual is simulated for 400 timesteps.

The task-agnostic objectives, novelty, curiosity and evolvability, are defined upon the behavior of the robot. Although different domain-agnostic ways to characterize behavior have been proposed and successfully applied to measure novelty [22] and curiosity [45], I will stick to the more domain-specific methodology used on novelty search's introduction [1]. Therefore, the behavior is characterized in a straightforward manner by the x,y-location of the robot at the end of the simulation.

The following section will describe how the behaviors are used by the selection criteria derived in this work and the control objectives which are used to measure the success of the controllers.

3.2. Entropy Objectives

This section presents the application of entropy to measure novelty, curiosity and evolvability. Therefore I will first introduce the original measures which inspired their use in this work, followed by their adaptations with entropy.

3.2.1. Novelty as Rarity

In the original work [1], an individual r's novelty is its sparsity in behavior space, computed as the average distance to its k nearest neighbors μ:

$$Nov_r = \frac{1}{k}\sum_{i}^{k} dist(r, \mu_i) \tag{3.1}$$

according to a distance metric $dist(\cdot\,,\cdot\,)$. With the behavior characterized by the final location x_r in the maze, distance is measured with the euclidean metric. The nearest neighbors are taken with respect to the current population and an archive of past behaviors. The archive is maintained to remember where the search has already been and discourage backtracking. In the initial work, the archive consisted of individuals whose novelty was above a certain threshold when discovered. The threshold would be adjusted dynamically throughout the trial in order to consistently add individuals to the archive. With that approach, the newly added behaviors would effectively act as repellent on the direction of the search, even though they actually indicate a region worthy of further exploration. In later work, [107] Lehman suggested to randomly sample λ individuals instead of taking the most novel ones. That way the archive would be more representative of where the search had been.

Novelty with Entropy

In order to use entropy, the search history must be represented as a distribution. This is done by discretizing all the N_{his} behaviors that have occurred during the search into a histogram. Novel behaviors, being in sparse regions of the behavior space, fall into bins of the histogram that are little or not populated. This histogram, which I will refer to as the *archive-grid*, will serve as the equivalent of the archive in novelty search and represent where the search has already been. The diversity of the search history can then be computed as the entropy H_{his} of that grid:

$$H_{his} = \sum_{i}^{N_{cells}} p * log(p) \tag{3.2}$$

with $p = \frac{|g_i|}{N_{his}}$ being the fraction of individuals in grid cell g_i of the archive grid and N_{cells} being the number of non-empty cells of the histogram.

An individual r's contribution H_{his_r} to that sum can be attained by splitting the entropy value of one grid cell among all the individuals in that cell:

$$Rar_r = \mathrm{H}_{his_r} = \frac{1}{|g_i|} * p * log(p) \tag{3.3}$$

$$= \frac{1}{N_{his}} * log(p) \tag{3.4}$$

Noteworthy, the rarity value is updated each generation, possibly changing as the cell of the individual becomes more populated. If the cell becomes more populated, its rarity decreases. Yet if it does not, the growing number of total individuals in the archive will make it even rarer, indicating that the underlying behavior might be worth of further exploitation.

This adaptation has two important ramifications:

1. Other than in the original adaptation, every individual's behavior is represented in the archive. This can be done because the complexity of comparing an individual to the archive stays the same, irrespective of how many individuals it holds.

2. The grid method is unable to compute exact distances between individuals from different cells. Consequently, it can not make a difference between an individual that is in a sparse neighborhood far away from the others and one that is is in a sparse neighborhood close to the rest.

Given these two differences to the original algorithm, I deem it more appropriate to name that quantity *rarity.* The implicit trade-off made by using entropy is that rarity can incorporate the entire search, but only with a certain resolution, whereas novelty search needs to sample the history of behaviors, but can compute exact distances. This allows novelty to differentiate between the aforementioned two individuals: As the first behavior is more farther away from the rest it will be assigned a higher reward by novelty. For rarity, the sparsity of the region is all that counts and the two behaviors will be equally good.

For the maze-navigation task, where the behavioral characterization is the end-position of the robot, the archive grid will thus be two-dimensional. Because the maze is bounded, the maximum and minimum coordinates are known in advance. The total number of cells in the archive grid is the squared number of bins used to discretize the x- and y-coordinate of the robot. When that number is N_{bin}, then equation 3.2 becomes:

$$H_{his} = \sum_{i}^{N_{bin}} \sum_{j}^{N_{bin}} p * log(p) \tag{3.5}$$

and $p = \frac{|g_{i,j}|}{N_{his}}$ being the fraction of individuals in grid cell $g_{i,j}$ of the archive grid.

3.2.2. Curiosity as Path Entropy

Curiosity refers to the diversity of the path an individual traverses during its lifetime. For the curiosity instinct as implemented in [45], this path is described by the sensorimotor-stream of the robot. It is then discretized into states with a clustering algorithm and each state is assigned a probability according to how often the robot has been in that state during its trajectory. The entropy of that distribution of probabilities is then interpreted as the individual's curiosity.

The sensorimotor stream was used to have a domain-independent drive that could be computed on board of the robot. As this work is interested more in the effect of defining entropy with respect to different frames of reference the behavioral characterization only needs to be adjusted to include temporal information, but should otherwise be defined similar to rarity. Therefore, the robot's x,y-location is mapped into a histogram at each timestep t of the simulation, capturing the robot's trajectory in a grid. At the end of the run, after N_{sim} timesteps, curiosity is measured as the sum over the entropy of the grid cells:

$$Cur = \sum_{i}^{N_{bin}} \sum_{j}^{N_{bin}} p * log(p) \tag{3.6}$$

with $p = \frac{|n_{i,j}|}{N_{sim}}$ and $|n_{i,j}|$ being the number of times the individual has been in the respective grid cell.

Therefore curiosity encourages the robot to travel long paths, thus visiting as many different states as possible and sharing its time equally among the states visited.

3.2.3. Evolvability as Grid Entropy

As described in section 2.3.1, evolvability can be interpreted in two different ways: As the amount of variation an individual can produce or as an individual's capacity to produce adaptive variation. The former interpretation is more lax as it only requires the offspring to be phenotypically distinct, whereas the latter requires the offspring to be fitter than their parents. Although very practical, the second definition suffers from the obvious drawback that the fitness function can be misleading. In that case it will render that approach useless for the more complex problems on which evolvability is needed. As the first interpretation does not suffer from that and is in its spirit closer to the idea of this thesis to find a goal-independent utility value I first tried to find a heuristic for this kind of evolvability. Also, the work on evolvability search [91] has already shown that this kind of evolvability is already favorable for generalization and diversity. Therefore, this work and other that measured evolvability in the maze domain [1, 48] approximate evolvability by the number of distinctly different offspring an individual can produce. Therefore a great number of offspring is spawned and the final positions of the candidate and its mutated children in the maze are mapped into a histogram. The evolvability of the individual then corresponded to the number of grid cells that held at least one individual.

Consequently, the adaptation with entropy is straightforward; evolvability is measured as the entropy of that histogram:

$$Evo = \sum_{i}^{N_{bin}} \sum_{j}^{N_{bin}} p * log(p) \tag{3.7}$$

with $p = \frac{|g_{i,j}|}{k_{off}+1}$ being the fraction of individuals in grid cell $g_{i,j}$ and N_{grid} being the total number of children generated.

The difference between using the number of non-empty cells and using the entropy is that entropy also considers how equally the individuals are spread out. Although this could favor those individuals that are more constantly producing diversity, it could also be irrelevant given that most mutations are deleterious and thus the resultant behavior is a product of mere chance.

The two interpretations will only be compared by their degree of correlation, as using them to drive the search is computationally too costly.

3.3. Control-Objectives

This section presents the control objectives used to test the effectiveness of the objectives. Next to the typical fitness function, the control objectives are behavioral diversity[17], which is another state-of-the-art method, and novelty, which is used to see the effect of using entropy.

3.3.1. Novelty

As explained above, the original work by Lehman [1] computes novelty as an individuals r's average distance to its k nearest neighbors μ among the current population and an archive (see equation 3.1).

Similar to this work, Lehman also uses the robot's end-position as characterization of its behavior and the euclidean metric for $dist(\cdot\,,\cdot\,)$. Other parameters for novelty are the number of neighbors k and the method according to which new individuals are added to the archive. The initial work and following adaptations [9, 63, 62] have used $k = 15$ and a comprehensive study about how to devise effective novelty search [61] confirmed it to be most performant across a variety of mazes. Gomes et al [61] also found that random sampling was indeed superior and that $\lambda = 6$ was a good value over a range of different mazes.

3.3.2. Behavioral Diversity

Another diversity measure in the behavioral domain has been proposed by Mouriet and Doncieux [44, 5]. Akin to novelty, it measures an individual's distance to other members of the population. It differs from novelty search in that it computes the mean distance to all members, instead of just its nearest neighbors and in that it does not use an archive. Instead it is used together with a fitness objective that is optimized in parallel by a MOEA, often NSGA-II [108].

$$Maximize \begin{cases} Fitness \\ Div_r = \sum_i^{N_{pop}} dist(x_r, x_i) \end{cases} \tag{3.8}$$

Similar to novelty and rarity, the BC will be the end-position of the robot and distance will be measured with the euclidean metric. Fitness is measured as distance between the end-position and the goal.

3.4. Heuristics for Evolvability

This section presents the approaches taken to approximate evolvability without doing extra simulations, instead using the lineage and the offspring of the individuals that are discarded during search. As all attempts at doing so were unsuccessful, I will instead describe the different ways it was tried and the underlying intuitions upon which they were based.

3.4.1. Estimating Phenotypic Variation from Past Behavior

As the capacity to produce a diversity of differently behaving offspring had already been proven to be valuable in the maze domain [1, 91], I first try to find a heuristic that would predict evolvability according to that first of its two interpretations. In order to avoid the many extra individuals that usually are created to approximate this exactly, I instead use the ancestors of the individual. Here, the assumption is that the lineage of an evolvable individual is very diverse. Individuals that are evolvable and have covered a wide range of behaviors in the past, are assumed to also be evolvable in the future. Consequently, the diversity of an individual's lineage could be used to approximate the diversity of its offspring. In order to test this, the individuals save their behaviors in a *lineage-grid*, which they pass on to their children. By that, each individual knows about the behaviors of their parents and grandparents. If the assumption is true, it should be possible to recognize evolvable lineages by the high diversity of their lineage-grids. Individuals that stem from a lineage of diverse behaviors should be able to spread their offspring over a wide range of grid cells, thus scoring high on evolvability. A more detailed account of how the transmitted behavior is characterized and the diversity of the lineage-grid is measured is given in the experiment section 4.6.1.

Noteworthy, this reasoning requires that the selection procedure, which effectively fills the lineage-grid, rewards novelty. If there were only a fitness objective, diverse offspring would not register in the lineage-grid unless they also outperformed all other individuals in terms of fitness. Therefore the evolution will start by only selecting for novelty, so that the lineages consist of individuals that, at their lifetime, were contributing to the search. After a certain number of generations, when the lineages are meaningful, the heuristics will be optimized parallely to novelty as a second objective.

To gauge their effect on the search, the heuristics will then be turned on and off at regular intervals, at the beginning and end of which evolvability will be measured.

Support for this approach is given by previous findings [1, 87] that indicate that selecting individuals for novel behavior also promotes evolvability. If the stepping stones discovered by novelty search confer evolvability to the search process, as is suspected by Doncieux & Mouriet in [26], one can expect evolvable individuals to be frequently selected for discovering new behaviors or niches [78] and in turn have a diverse lineage.

As this attempt to approximate the diversity of an individuals offspring failed, I then tried to approximate evolvability as an individual's success of producing children that are likely to be selected.

3.4.2. Estimating Adaptive Variation with Feedback from Children

In its second interpretation [75] evolvability refers to an individuals capacity to produce adaptive variation. Works from EC [77] that adopt that view require individuals to produce children that are fitter than themselves, so that they contribute to the progress of reaching the objective. Although very practical, that suffers from the obvious drawback that the fitness function can be misleading, which renders that approach useless for the more complex problems on which evolvability is needed.

Therefore, I suggest to replace the fitness objective with a diversity objective. Instead of fitness, children are measured on whether or not they perform high on a diversity objective. To avoid having to produce many extra offspring that are not used for the search, it was implemented that the actual children provide feedback to their parents whether or not they have been selected to be part of the next generation. The amount of positive and negative feedback an individual got was then used to approximate its contribution to the search progress.

As this feedback mechanism only gives credit in retrospect, another diversity objective needs to be used to select between the newly created children. Furthermore, the algorithm must keep a gene pool, which is updated after each generation to hold the most successful individuals produced by the entire search. This approach, also referred to as elitism, is common in different MOEAs [108, 109, 110]. Keeping the most successful solutions as parents instead of discarding them significantly increases convergence [111], as only a small fraction of each generation actually represents an improvement.

So whenever a child is newly adopted into the gene pool, its parents, if also still present, are rewarded. Analogously, parents are punished when they are disproportionately bad at producing successful children. For a MOEA, 'successful' typically means the individual represents a new trade-off between the target objectives, so that it gets taken up into the archive (see section 3.5.1 for a more detailed explanation of how this trade-off is measured with pareto-dominance). Therefore, each generation the average rate of newly adopted individuals into the gene pool is computed and parents that have spawned

children are compared to that number. For the experiments to come, this ability to produce good children will be referred to as *viability* and abbreviated with (VIAB).
An individual r's viability at iteration t is computed as:

$$VIAB_{r,t} = 1 * children_{r,elite_t} - progress_t * children_{r,spawned_{t-1}} \tag{3.9}$$

with $progress_t$ being the percentage of children adopted into the gene pool at iteration t, $children_{spawned_{t-1}}$ the number of children individual r spawned in the iteration before and $children_{elite_t}$ referring to the number of its children that have been taken into the gene pool. Thus, the first part of the sum is the actual number of r's children that have been adopted and the second part is the expected number, calculated by multiplying the number of its offspring with the average adoption rate (progress).
Consequently, individuals whose children can not contribute anymore to the progress of the population, will have their viability decreased, whereas those whose children are adopted is increased. Thus, viability is a feedback on the efficiency of the exploitation mechanism: In the case of the maze, an example for the former would be an individual that by lucky circumstances very quickly evolved a structure that walks to a unusually rare point in it, but is so fragile that all its children are useless. Without the viability objective, it would be selected repeatedly for reproduction until the population has caught up to it. Until then, it will be selected again and again, as its behavior is very rare. Yet with viability, there is a criteria that reflects its missing contribution to the overall progress of the search.
The obvious drawback of this method is that only few individuals are used and the feedback works only in retrospect. Yet as progress on difficult objectives is typically slow (in the maze-domains of this work, only about 2% of each generation get adopted into the gene pool), the individuals might stay long enough in the gene pool for this heuristic to become accurate enough to support the search.

3.5. Evolutionary Setup

This section present the choice of the MOEA used to optimize two or more objectives in parallel and the experimental parameters which are assumed to have no influence on the functionality of the diversity-objectives, such as the choice of encoding and the method of assembling the child generation from the gene pool.

3.5.1. Ranking with NSGA-II

The MOEA used in this work is NSGA-II [108]. It is computationally fast and widely used in ER, for example to combine novelty with other objectives [61, 60, 9], to optimize behavioral diversity together with fitness [17] and in a study that compared novelty and behavioral diversity with different behavior characterizations and metrics [5].

Like most MOEAs, it relies on the concept of pareto-dominance to compare the quality of different solutions. A candidate $\tilde{x}$ is considered to dominate a candidate x if

1. $\tilde{x}$ is not worse than x in all objectives.

2. $\tilde{x}$ is strictly better than x in at least one objective.

Individuals that are not dominated by others in the population thus represent a unique trade-off between the objectives to be optimized.

It is important to note that pareto-dominance does not imply an ordering on all candidate solutions. Candidates can have very differing values for the single objectives, yet no one dominates the other. When the number of objectives is very high, then the proportion of non-dominated solutions increases. That is also why MOEAs with more than four objectives are considered many-objective optimization methods [59] and rely on different concepts. But also with few objectives the number of non-dominated candidates can increase quickly and that is why MOEAs typically use a crowding mechanism to determine how densely the region a non-dominated solution resides in is populated. Figure 3.3 illustrates the optimal line along which the solutions can spread, the pareto-front, and two populations that approximate it in a desirable and undesirable way.

To preserve a diverse set of trade-offs between the different objectives and to make sure that the good solutions are not lost, NSGA-II maintains an external archive of a fixed size. This archive, also referred to as gene pool or elite, holds the best non-dominated solutions found by the entire search. Every iteration of the search a new generation of children is generated from it. After the children have been evaluated, the archive is assembled anew by selecting from the combined population of children and parents, preserving a diversity of trade-offs. To assert that diversity, all solutions are ordered according to

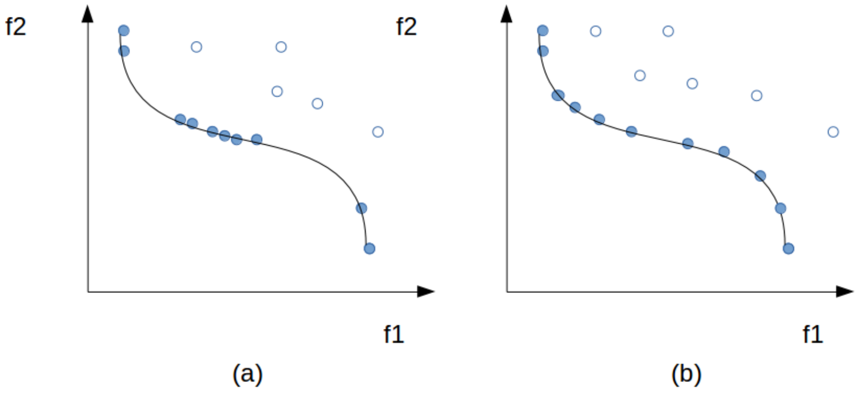


Figure 3.3.: Two objectives f1 and f2 are optimized simultaneously. The points represent possible solutions along the pareto-front. The blue points that lie on the front are non-dominated, whereas the white ones farther to the upper right are dominated. (a) presents a group of solutions with little diversity - the points lie closely together around the *knee* of the front. The solutions in (b) maintain a higher diversity and are a better representation of the trade-offs between the two objectives.

a specialized ranking system. To compare the solutions, each is assigned two values: (i) a non-domination rank and (ii) a crowding distance assignment. The first is determined by how many solutions need to be removed from the population in order for it to be non-dominated. The second measures the average distance for each objective to its nearest neighbor in both directions. Figure 3.4 shows that this is the average side length of the cuboid spanned by the nearest neighbors in the front. These two properties then allow to define a strict order on all solutions: When comparing two solutions one is considered superior if it has a lower non-domination-rank or, in case those are equal, if it has a higher crowding distance. The archive of the next generation then consists of the top half of the combined parent and child generation, sorted according to that order.

NSGA-II can also be run with just one objective. Then it acts as an normal generational elitist EA. As NSGA-II only dictates an ordering on the solutions, the choice of how to create the child generation from the parent generation is in the hand of the experimenter. This includes choices about the selection and mutation mechanisms that will be described in the next section

3.5.2. Exploitation by Tournament Selection

Whereas the objectives so far have been mainly considering how to effectively increase the amount of exploration, selection and mutation will be used to

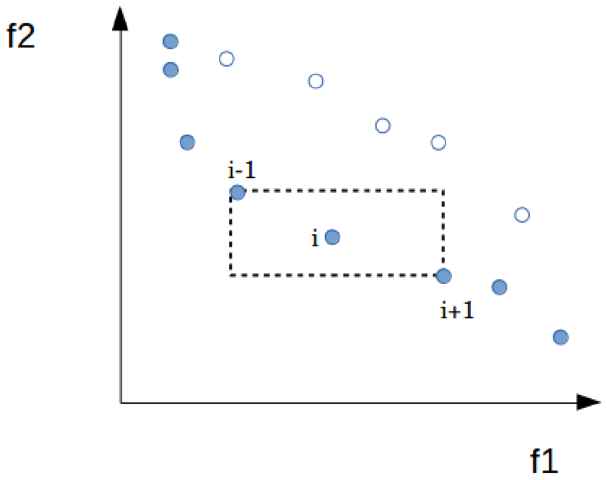


Figure 3.4.: The crowding distance of the candidate solution i is computed as the average side length of the cuboid.

ensure exploitation.

Exploitation by selection is done by creating more offspring from the high scoring individuals. By representing them more in the next generation, the potential benefits that can be gained from finetuning their behaviors are more likely to be discovered. A typical selection approach for generational EAs, such as NSGA-II, is to do x-size tournament selection, as described in [112]. Therefore x individuals are randomly sampled from the population and the fittest, or in the case of NSGA-II the one ranked highest, is added to the next population. This is repeated until the next generation is filled. The size of the population and the size of the tournament both influence the degree of exploitation. Small populations and bigger values of x increase exploitation: The bigger the tournament size, the less likely it is for unfitter individuals to end up in a tournament where they come out as winners. And the smaller the population the less tournaments provide that opportunity. In this work I use a tournament size of 4 and a population size of 100. The assumption is that the degree of exploitation will not distort the results as long as it is not too low or too high.

For mutation to contribute to exploitation, it is important that the mutated genome changes the behavior only by a small margin. These variations still occur randomly, but the more slightly different variations of that behavior are tested, the more likely it is fine-tuned. I don't use crossover operations, as those often contribute more to exploration when they pair two genetically very different individuals.

NEAT-parameter	Value
Probability of adding a new node	0.03
Probability of adding a new connection	0.05
Trait mutation power	1

Table 3.1.: Important parameters for the NEAT algorithm. Weight mutation power affects how strongly the connection-weights are changed.

The next section will describe the NEAT method, which is used to encode the ANNs on a genome and mutate them when creating offspring.

3.5.3. Encoding with NEAT

Encoding methods in ER can either be direct [113] or generative [66, 114, 25]. The first directly maps genes to phenotypic traits, whereas the latter encompass a one-to-many mapping. Thus generative encodings facilitate the discovery of regular or modular phenotypic patterns, for example by also evolving the type of activation function used in the neurons (linear, gaussian or sine-functions). For the maze-domain, the ANNs are encoded using the generative NEAT method [106], which is widely applied [1, 54, 91] and well understood. NEAT and its adaptations [66] are popular neuroevolution methods, as they provide a good way to encode ANNs, align genomes to do crossover and iteratively complexify the topology of the evolved networks. The last point is especially desirable as it favors the discovery of simple solutions before more complex ones. This effectively limits the size of the search space throughout the entire search. Comparative studies [61] have shown that novelty search and behavioral diversity work about equally well when using the NEAT-encoding or other alternatives. As those objectives and the ones newly proposed in this work are defined in the behavioral space, I suppose that they are largely independent from the neuroevolutionary algorithm that is used. The decisive parameters for NEAT are shown in table 3.1. The other parameters are the default parameters taken from the NEAT user page [2] and are listed in the appendix A.1.

4. Experiments & Results

In order to show the usefulness of the entropic objectives I use them to solve two maze-navigation tasks. The evaluation criteria for the maze performance are the convergence rate, showing how consistent the objectives find a solution, and the average number of generations until the solution was found. The consistency is the primary measure for success because solving speed is only important when the maze is solved consistently. After analyzing the performance of curiosity and rarity, the following sections will address potential concerns and questions raised by the results. The following section shows the experiments undertaken to investigate the heuristics for evolvability. Finally, the most interesting results are verified in the biped domain.

4.1. Maze Performances

All experiments in this and the following section are run with a population of 100 for a maximum of 1000 generations. Unless specified differently, the resolution of the grid used to discretize the robot's end positions was set to 15, a choice that will be justified by the experiments in the section 4.3.
First, curiosity and rarity are used to drive the search in two maze navigation tasks. The mazes (see Figure 3.1) increase in their degree of deception to illustrate the objectives capacity to escape local optima. Both objectives and their combination are compared to the traditional fitness objective, another state-of-the-art diversity method that combines fitness and behavioral diversity (FIT+DIV) [17] and novelty search. The results for both mazes are shown in the tables 4.1 and 4.2.

The main result is that the methods that increase diversity with respect to past and present search results, rarity and novelty, are most successful. For both mazes, they are most consistent as they reach the goal area in 30 out of 30 trials. Moreover, both objectives are significantly faster than the rest ($p < 0.05$ according to a two-tailed Mann-Whitney-U-test). Averaged over both mazes, rarity needs about 132 generations ($SD = 95$) to find a solution and novelty 154 generations ($SD = 235$), whereas the next method, fitness with behavioral diversity (FIT/DIV), needs about twice as long (467, $SD = 187$). The significance values for the other speed differences can be seen in the appendix B.2 and B.1.

Curiosity solves both mazes in about half of the trials. In the medium maze, that is the worst convergence rate of all objectives. In the hard maze, curiosity is only better than fitness, which does not converge at all. On average, curiosity needs 397 generations to find a solution ($SD = 292$). When combined with one another, rarity and curiosity show no unexpected interaction. The performance of it is worse than rarity and better than curiosity.

Fitness performed well in the medium maze (70% convergence), but never found a solution in the hard maze because the population got stuck in the local optimum close to the goal (see Figure 4.2). When supplemented with a diversity measure, the performance improved but the hard maze was still solved less frequently (57% vs 70%) and slower (448 vs. 325 generations). Remarkably, this dynamic is reversed for the diversity objectives, i.e. the harder maze is solved faster than the medium maze for curiosity, novelty and rarity, showing that diversity as sole objective is insusceptible to deceptive local optima.

Objectives	Convergence Rate	Solving Time	SD	N
RAR	1	157.77	140.65	30
NOV	1	193.8	155.7	30
RAR/CUR	0.83	444.68	266.97	30
FIT	0.73	333.09	284.67	30
FIT/DIV	0.7	325.05	173.85	30
CUR	0.47	471.07	322	30

Table 4.1.: Medium maze: Convergence rate and solving speed for rarity, curiosity and the control objectives. The maximum number of generations was 1000 and the population size 100.

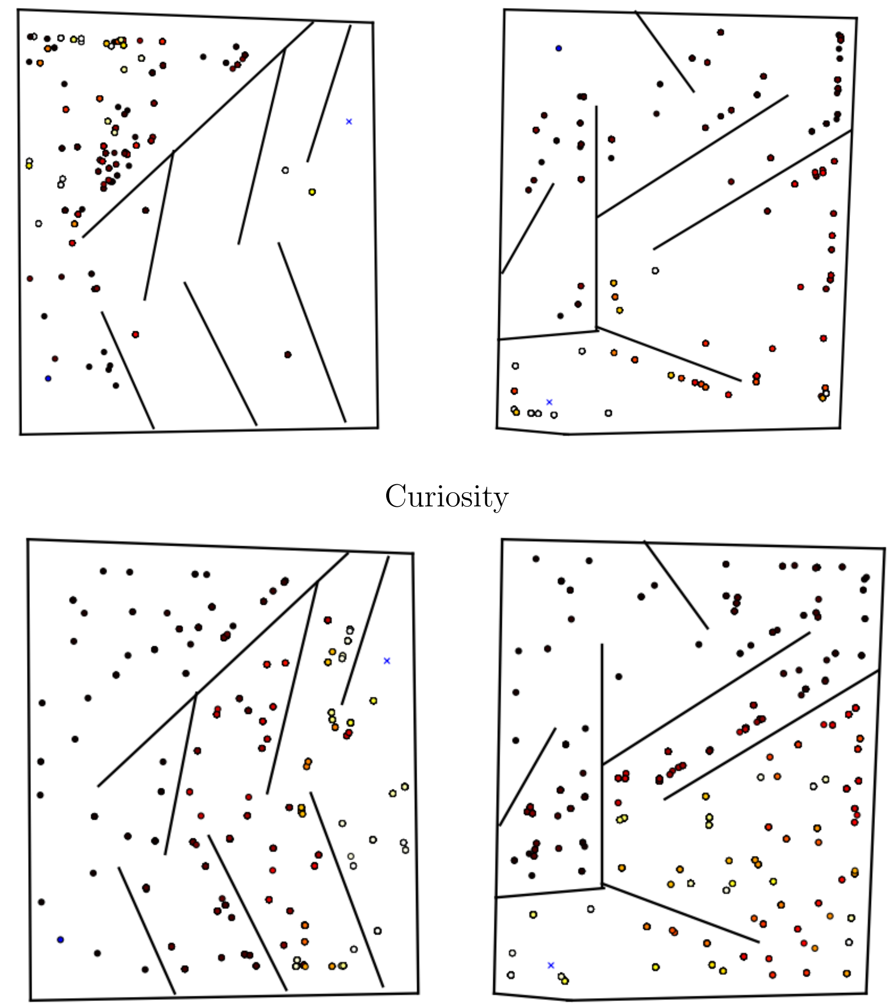
Objectives	Convergence Rate	Solving Time	SD	N
RAR	1	106.87	48.54	30
NOV	1	114.8	48.67	30
RAR/CUR	0.97	210.1	202.89	30
FIT/DIV	0.57	447.24	284.41	30
CUR	0.53	324	262.05	30
FIT	0	0	0	30

Table 4.2.: Hard maze: Convergence rate and solving speed for rarity, curiosity and the control objectives. The maximum number of generations was 1000 and the population size 100.

4.2. Typical Behavior

With fitness guided search, the majority of the population ends up in the local optima closest to the goal (see Figure 4.2). When fitness is supplemented with the behavioral diversity objective, the population splits in two: One half converges in the local optimum and the other in the corner of the maze opposite to it (Figure 4.3). This is because the behavioral diversity is measured as average distance to the entire population and is thus affected by the part of the population stuck in the local optimum. Individuals on right path to the goal need to traverse an area in the middle of the maze, which passes close to the local optimum. Consequently, the average distance to the population on that path is very low so that neither fitness nor diversity will encourage exploring it. By including the entire population in the diversity score, this search method is still handicapped by deceptive local optima.

Typical runs for the entropy-based objectives rarity and curiosity are shown in Figure 4.1, Rarity exhibits an even distribution across the maze, progressing through the maze as the search goes on. When maximizing curiosity the robots tend to stick close to the walls and in wide areas, possibly because those might facilitate traveling long paths.



Curiosity

Rarity

Figure 4.1.: Typical runs for the diversity-based objectives curiosity (top) and rarity (bottom). For each generation, the ten highest scoring individuals on the respective objective are plotted. The color of the dot indicates the generation, from dark red (first generations) until white (generation 200).

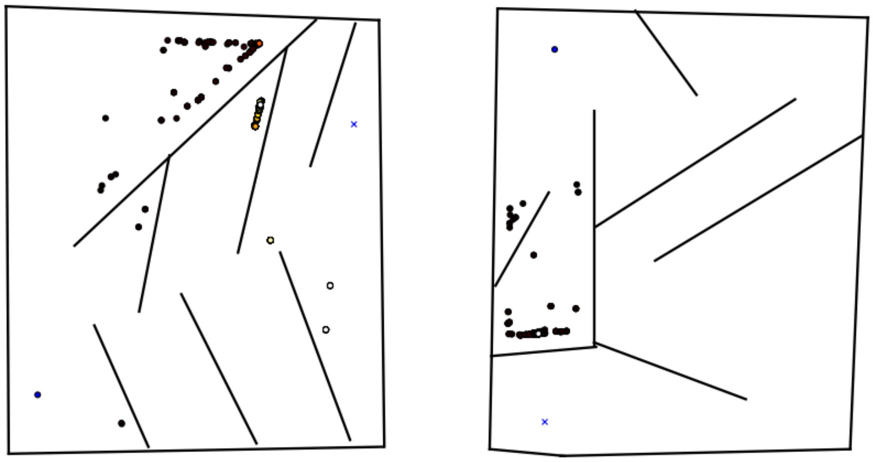


Figure 4.2.: Fitness

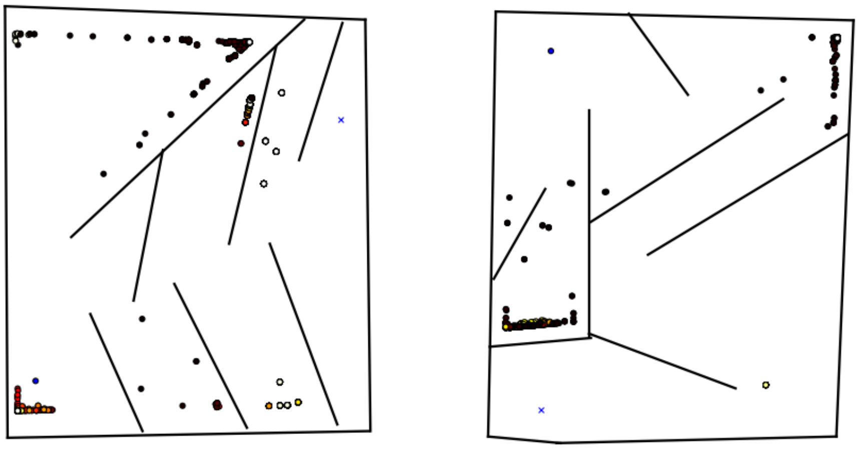


Figure 4.3.: Fitness + behavioral diversity

Figure 4.4.: Typical runs for the fitness- (top) and fitness with behavioral diversity (bottom) guided search. For each generation, the ten highest scoring individuals on the respective objective are plotted. The color of the dot indicates the generation, from dark red (first generation) until white (generation 200). Fitness was measured as the euclidean distance between the robot's end location and the goal.

All experiments in this and the previous section were run with a population of 100 for a maximum of 1000 generations. The resolution of the grid used to discretize the robot's end positions was set to 15, a choice that will be justified by the experiments in the next section.

Overall, these experiments affirm that (i) ensuring diversity with respect to an individuals immediate experience (curiosity) or with respect to past and present experiences (novelty and rarity) can efficiently solve deceptive mazes. And (ii), modeling novelty with entropy was possible without a decrease in performance.

The next two sections will present experiments that address possible concerns and questions raised by the formalization of rarity and curiosity.

4.3. Varying the Discretization of the Behavior Space

Both curiosity and rarity rely on a grid to discretize the robot's behavior. As the resolution of the archive grid decides which locations are considered to be equivalent and which can be differentiated, it has the potential to strongly affect the success of the objectives rarity and curiosity. Big (or small) grid sizes are equivalent of a grid with many (few) cells, leading to a many (few) behaviors being collapsed into the same point in the search space. Results for using various grid sizes with curiosity, rarity and their combination are shown in Figure 4.5.

Curiosity is sensitive to size of the grid. It performs best at low grid-sizes, which require individuals to travel longer distances in order to score high on curiosity. As the goal is at the far end of the maze, this aligns curiosity and solving the maze, as both require long distances. At higher grid sizes, it is easier to traverse different states without leaving the starting area of the maze, so that finding the solution takes more time and occurs less often. But as too low resolutions allow only a very crude distinction between the paths taken through the maze, it remains questionable whether curiosity is useful for more sophisticated problem domains. With that possible shortcoming and too high resolutions decreasing convergence and speed, the choice of the grid-parameter for curiosity is not trivial.

Contrary to that, rarity's performance is robust to high or low grid sizes, also performing slightly faster for the lower sizes. For very low discretizations, the convergence rate decreases a bit, probably because a smaller grid size drives the population towards new areas at the risk of overlooking a solution if it lies in a already densely populated grid. If, for example, the maze contained a bottleneck that would by accident be located in the corner of a large and densely sampled grid cell, then the appearance of an individual able to pass through the bottleneck becomes less likely, as the variants that nearly pass through would be punished for ending up in an already populated grid. A higher resolution collapses less behaviors into the same cell and thus decreases the chance of confounding behaviors which should better be differentiated. Hence, the grid cells become very small and individuals are more likely to register as rare. In that case, it takes longer to progress to new areas of the maze, but ultimately those areas will be discovered too. As a general rule, rarity is very robust to the grid size, which when in doubt can be used to trade of speed for convergence rate.

Looking beyond the application of rarity to this domain, discretizing the behavior space has an important drawback when the number of behavioral dimensions increases: When discretizing a behavior space with an equal amount of bins per dimension, the total number of distinct behaviors grows exponentially with the number of dimensions used. Consequently, the distinction between behaviors becomes so fine-grained, that almost all behaviors become rare. The next section considers this effect in more detail and presents two ways to address it.

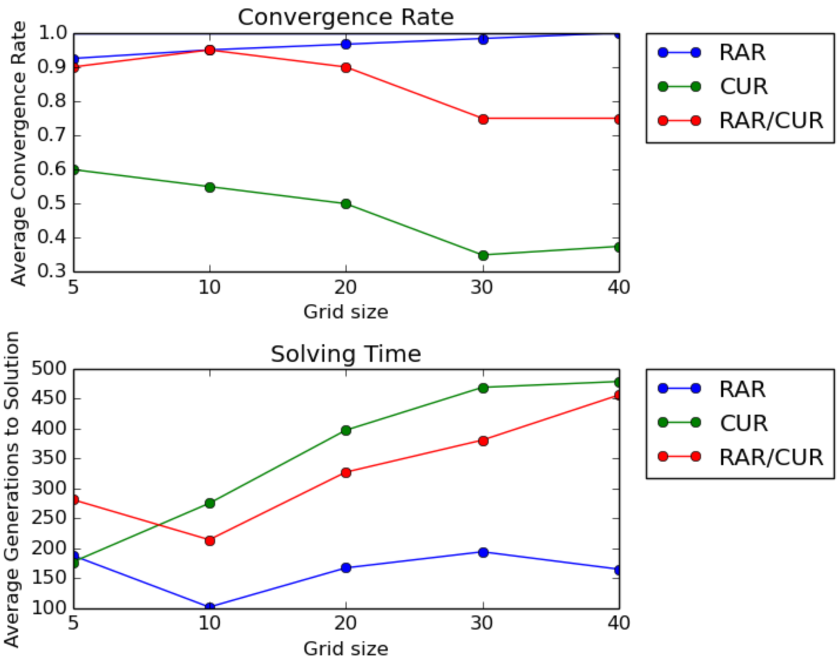


Figure 4.5.: Performance of the rarity and curiosity objective, affected by grid discretization: The upper graph shows the convergence rate, indicating how regularly a solution is found. The lower graph shows how fast the first solution was found. All combinations of objectives are relatively robust to the choice of the grid size parameter. The results in these graphs are averaged over the two mazes, with each grid discretization being run 30 times per maze. The maximum number of generations was set to 1000.

4.4. Lengthening the Behavioral Characterization

As rarity requires the degenerate behaviors to be sampled more often, the total number of behaviors it discriminates is a possible concern. With the robot's end-position as behavioral characterization, the number of all possibly existing behaviors is only the squared number of bins used to discretize one dimension, and therefore the archive can be filled quickly. Yet it is easy to imagine situations where a two-dimensional description of behavior is not sufficient or convenient to the experimenter. An example illustrates the implications of having the archive's size depend on the dimensionality of the behavior: Depending on the number of dimensions d used to characterize an individual's behavior and the number of bins k used to discretize them, the archive needs to keep track of the frequencies of k^d different behaviors. If, for example, the behavioral characterization had $d = 10$ dimensions, each discretized with $k = 10$, there would be 10 billion different behaviors. In order for rarity to recognize the behaviors in one cell as being degenerate, it requires the cell to be more populated than the others. But the more possibly behaviors exist in the archive, the longer this can take. With evaluations typically being the computationally most demanding parts of EAs, it is therefore interesting to consider ways in which the size of the archive can be limited.

As the combinatorial explosion occurs as all these d behavioral dimensions are discretized with respect to each other, a simple approach would be to discretize each dimension by itself, thus keeping d archives with k cells instead of one archive with k^d cells.

A behavior is then discretized into d cells, each of which with a different rarity-score. For each dimension d_i, the individual r is credited with the *dimensional rarity* Rar_{d_i}. Analogously to equation 3.3, Rar_{d_i} is computed as the individual's contribution to the rarity of its cell $g_{r,i}$:

$$Rar_{d_i} = -\frac{1}{N_{his}} * log(p) \tag{4.1}$$

with $p = \frac{|g_{r,i}|}{N_{his}}$ being the fraction of individuals throughout the search whose i-th behavioral dimension got discretized into that grid cell $g_{r,i}$. The total rarity of an robot is the sum over all d dimensional rarities:

$$Rar_{naive} = \sum_{i=0}^{d} Rar_{d_i} \tag{4.2}$$

As the approach naively neglects the possible relations between subsequent dimensions (the x- and y-position of the same location), it will be referred to as *naive rarity* (naiveRAR).

To simulate a higher dimensional behavior without the overhead of switching the domain, the position of the robot is sampled at regular intervals during its run through the maze. Therefore $d/2$ samples of a robot r's location are taken at timesteps $t_0, t_1, ..., t_d$ and concatenated into a vector $b_r = \{x_{t_0}, y_{t_0}, x_{t_1}, y_{t_1}, ...x_{t_{d/2}}, y_{t_{d/2}}\}$, resulting in a behavioral characterization of length d.

This also allows to investigate a more principled approach to assemble the different archives. Using the relation inherent to a time-series of samples, the second approach would be to only relate dimensions that are sampled at the same timestep, thus using $d/2$ archives with k^2 cells. The corresponding total score would then be

$$Rar_t = \sum_{i=0}^{d} Rar_{t_i} \tag{4.3}$$

$$= \sum_{i=0}^{d} -\frac{1}{N_{his}} * log(p) \tag{4.4}$$

with $p = \frac{|g_{r,i}|}{N_{his}}$ being the fraction of individuals throughout all the history of the search whose location at timestep i also got discretized into the grid cell $g_{r,i}$ of archive i.

These two approaches differ in the amount of prior knowledge about the dimension the experimenter chose to include. As knowing beforehand which dimensions benefit from being related is not trivial, the first approach simply relates none of them. The other approach follows the simple intuition of using the position on the robot's path. As this makes uses of the temporal ordering of the dimensions, it will be referred to as *temporal rarity* (tRAR).

Both approaches and the initial rarity (RAR), which does relate all dimensions were run 10 times in each maze with behavioral characterizations up to 400 dimensions. For the latter, the robot's location was sampled every second step of the simulation. Convergence rate and solving speed are shown in Figure 4.6).

tRAR and naiveRAR solve the mazes consistently (> 95% of the trials). Surprisingly, naiveRAR converged as good as tRAR. The only advantage of tRAR is that it is a bit faster, yet not by a large margin. The unadjusted RAR-objective, which relates all dimensions, quickly degrades in performance.

Moreover, the space requirements for RAR exceeded the memory capacities of the computer when tracking behaviors with more than 40 dimensions. Yet with the relaxed dependence assumptions, the dimension of behavior does not seem to have an influence on the performance of rarity.

To confirm that the performance of the adjusted rarities was not specific to one particular grid size, tRAR was run with various grid discretizations k. Figure 4.7 shows that at all grid and sample sizes, tRAR successfully finds a solution, only with a sleight increase in solution time for higher dimensional BCs.

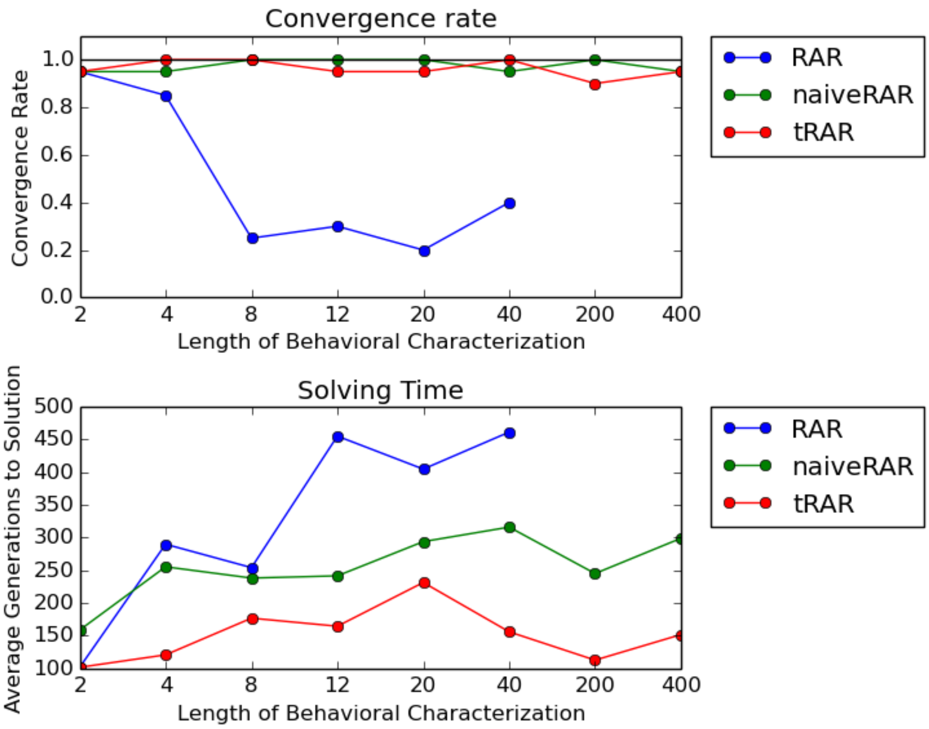


Figure 4.6.: Performance of the different rarity adaptations, affected by the length of the behavioral description: The upper graph shows that both tRAR and naiveRAR consistently find solutions, even for very high dimensional behavioral characterizations. The lower graph shows that tRAR, which other than naiveRAR relates the x- and y-locations of the same timestep, finds solutions the fastest.

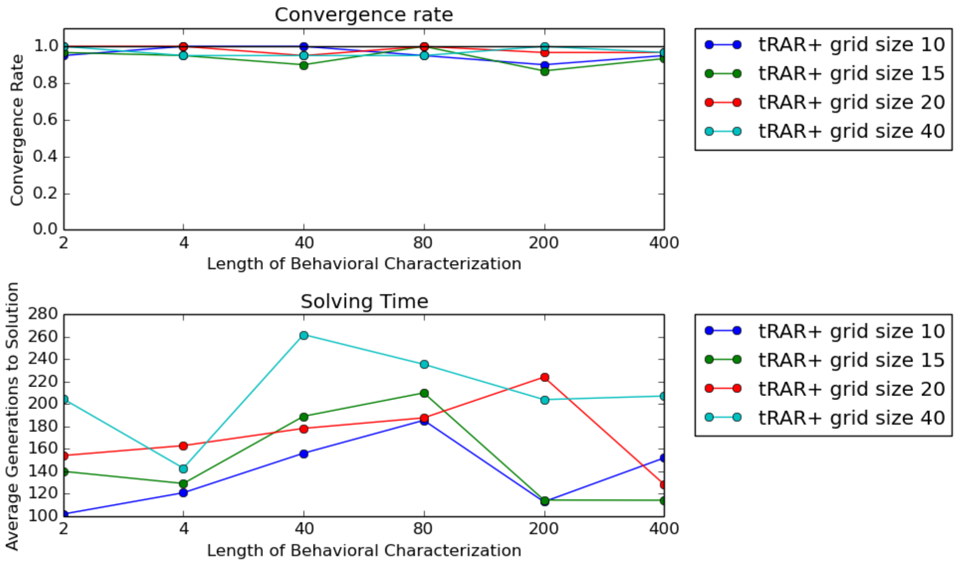


Figure 4.7.: Performance of differently discretized tRarity-objectives, affected by the length of the behavioral description: The upper graph shows that for all resolutions of the individual sample positions (caused by different grid sizes used to discretize the position in the maze) the maze is consistently solved, independent of the length of the behavioral description. The lower graph shows that higher grid-resolutions increase the time until the first solution is found.

The previous two sections have shown that rarity is robust to its hyperparameter of the grid discretization and can be adapted to behavioral descriptions of greater length by using multiple archives.

4.5. Entropy as Selection Principle

The experiments in this section show that the success of the rarity-objective is not only due to the expressiveness of the robot's position as the behavioral characterization but also to the use of entropy as selection principle, selecting those individuals that are most likely to contribute to preserving diversity.

Therefore I adapt the entropy-selection to curiosity and fitness, by discretizing them and keeping track of the frequencies with which their values have appeared throughout the search with an archive. Instead of selecting the individuals with the maximum values, rarity then selects those which have occurred least often. For curiosity, this will be referred to as frequency-curiosity (frCUR). For fitness, this alternative selection scheme will be referred to as fitness frequency assignment (FFA), as suggested by Weise et al in [33], where they apply it to a genetic programming problem. Both fitness and curiosity have maximum and minimum values, so that the number of bins used to dis-

Objectives	Convergence Rate	Solving Time	SD	N
FFA	0.68	425.47	212.81	20
frCUR	0.6	346.75	191.02	20
CUR	0.55	253.04	191.53	20
FIT	0.38	0	239.17	20

Table 4.3.: Results for replacing the maximum selection mechanism for a frequency selection mechanism for fitness and curiosity. Results are averaged over two mazes. For all discretizations, both to identify frequencies and to differentiate paths for curiosity, the grid size was set to 10.

cretize the objective value was set to 10. The grid-size parameter for curiosity was also set to 10. Each method was run 20 times in both mazes.

Table 4.3, shows the results averaged over both mazes. Results for the individual mazes are shown in the appendix B.4 and B.3. Fitness improves remarkably when entropy is used as selection principle. In the medium maze, performance increases from 75% to near perfect (90%). Importantly, fitness frequency is less susceptible to local optima, shown by the improved convergence in the hard maze (0% vs. 45%). For curiosity the performance stays about the same. frCUR is slightly better in the medium maze, CUR in the hard maze.

4.6. Searching a Heuristic for Evolvability

Two approaches to estimate evolvability are tried, resting on different ideas how to avoid having to do extra simulations.

4.6.1. Estimating Evolvability from Past Behavior

The first interprets evolvability as the phenotypic variation of an individuals offspring and tries to approximate it from the diversity of the lineage of the individual. The underlying hypothesis of this approach is that in the presence of selection for novelty, an evolvable individual will have a diverse lineage-grid. Thus promoting diverse lineages should favor individuals who can spawn diverse offspring.

To be able to validate that hypothesis, it is important to consider the different ways in which its key-elements are realized. These are (a) the way the lineage-grid is composed (all ancestors or only the most recent ones) (b) the way the diversity of it is measured (the spread of the lineage or just the number of distinct behaviors in it) (c) the kind of behavior that is registered in the grid (only the end-position or the entire path)

Therefore, the following enumeration lists the different heuristics that are used, each representing a different combination of the above factors.

1. Lineage-Grid-Entropy (LGE) is measuring the entropy over the lineage-grid, assuming that evolvable individuals have shown a wide range of different behaviors as in 3.5.

2. short-term LGE (shLGE) is measuring the entropy, but only over the most recent parents, assuming that older ancestors are less important.

3. LG-Diversity (LGD) is measuring only the number of different behaviors in the lineage-grid. Contrary to 1.) and 2.) this neglects how evenly the ancestors are spread out. This should encourage development towards unseen regions, rather than making sure the lineage is nicely distributed.

4. LGDr is measuring LGD over a reduced, coarser grid (lower resolution), so that behaviors have to be farther apart to register as different. Therefore the number of grid cells was reduced by a factor of 0.8, 0.6 and 0.4.

5. short term LGD (shLGD) is measuring the number of different behaviors, but only among the most recent parents. Both shLGD and shLGE were evaluated with lineage-lengths 100, 50 and 25 ancestors.

6. LGDnd is measuring the amount of different behaviors, normalized by the length of the lineage.

7. The discovery instinct: Here the complete path of the individuals is mapped into the lineage-grid (instead of using only their end-positions). Then diversity is measured as the entropy of that grid.

Criteria LGD and LGE have a bias towards older lineages. That bias can be neutralized by taking only the most recent history into account, assuming that once an individual has an evolvable structure it can continuously produce diverse behavior. (6) reverses that bias by dividing through the length of the lineage, effectively favoring short lineages. Thus, lineages of length one are assumed to be evolvable, whereas the longer the lineages need to maintain that status by not tracking back to already exhibited parts of the grid. Criterion (7), inspired by the discovery instinct from [45] saves the complete path an individual takes through the maze, thus mapping the whole area covered by that lineage. The intuition behind this is that evolvable lineages should be able to cover larger ground.

4.6.2. Results for Using Past Behavior

To see whether the criteria are useful proxies for evolvability, I measured evolvability and its correlation with the proxy-criteria at regular intervals. As there are no meaningful lineages at the beginning of evolution, the first iterations are run only with the rarity objective. Later the heuristics are run together with the rarity, as they require selection to favor the discovery of new behaviors. To single out the effect of the proxy, it was turned on and off in regular intervals, at the beginning and end of which evolvability was measured. Therefore 200 individuals were generated per individual and simulated in the maze. Evolvability was then measured as the entropy of the end-locations of all offspring, as described in equation 3.7. In both mazes, this measure correlated with $r = 0.9$ with the way evolvability was measured previously in works on novelty search [1] and evolvability search [91], which only counted the number of distinct behaviors in that grid instead of using its entropy.

The result is that all of the criteria are harmful for evolvability. This can be seen in Figure 4.8, which shows how the evolvability of the individuals develops 200 generations in the medium maze. It shows a zigzag-curve, going down when the objective is selected for and going up when its selection is disabled. Figure 4.8 also is representative for other interval-lengths and runs in the hard maze, i.e. they always result in a zigzag-curve. On the upward motion, the individuals are only selected for their rarity, which has a positive effect on evolvability (correlation coefficient around $r = 0.5$). For comparison, the sole effect of rarity on evolvability is also shown in the graph as a black line.

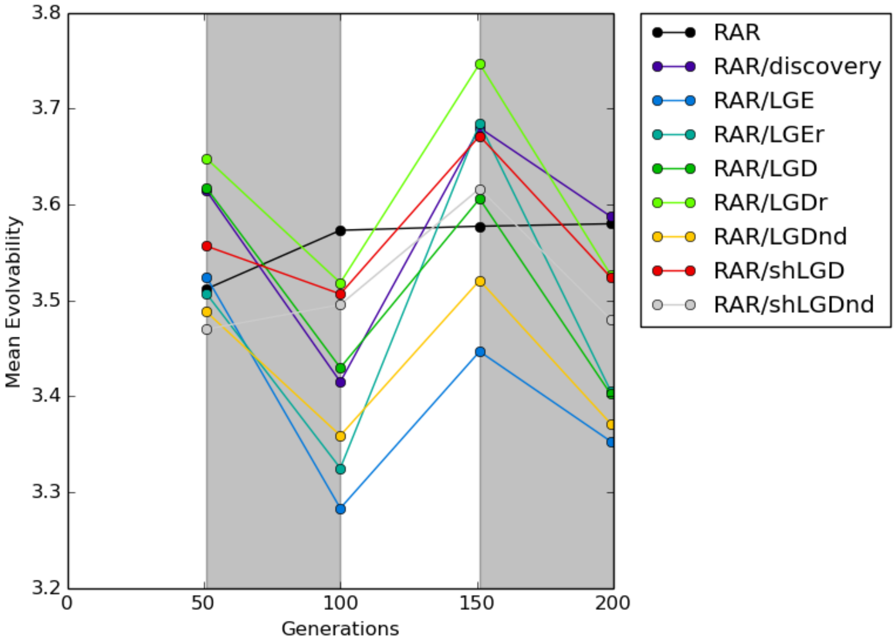


Figure 4.8.: Evolvability over time: The shaded areas mark generations when the selection for the proxies was enabled. White background signifies that rarity (RAR was the sole objective. There results are obtained by averaging over 2 trials in the medium maze. Results in the hard maze were similar. The zig-zag pattern was present also for different interval-lengths, for larger numbers of maximum generations (up to 1000) and for all values of grid-resolutions (LGDr) and lineage-lengths (shLGD and shLGE).

Table 4.4 shows the correlation (pearson correlation coefficient) with evolvability at the last generation. All r-values are negative, showing that it is indeed harmful. A more detailed picture of how the objectives relate can be seen in Figure 4.9,which shows a scatterplot of some of the objectives against evolvability. Apart from the negative correlation, the scatterplot also reveals that the individuals with high lineage-grid entropies score very low on fitness, indicating that most of them have stayed in the initial area of the maze. In an attempt to make the lineage more meaningful, I included even more ancestors by also feeding back the position of the children into the lineage-grid of the parents. The lineage-grid then represented the area covered by the behavior of the ancestors plus all one step mutations of them that were spawned throughout evolution. Yet even then, none of the introduced measures correlated with evolvability or had a positive effect on it.

Target-objective	r_{EVO}	r_{RAR}	r_{FIT}
LGE	-0.15	-0.5	-0.46
LGEr	-0.2	-0.34	-0.34
LGD	-0.22	-0.4	-0.47
LGDr	-0.27	-0.3	-0.36
LGDnd	-0.23	-0.34	-0.46
shLGDnd	-0.38	-0.47	-0.55
shLGD	-0.38	-0.5	-0.57
discovery	0.01	-0.25	-0.25
RAR	0.44		0.8

Table 4.4.: The pearson correlation coefficient for the last generation of two trials in the medium maze. All objectives were selected for together with the rarity-objective.

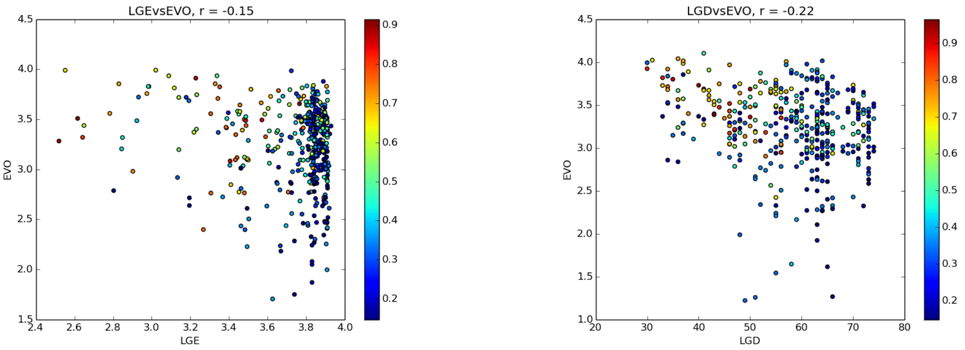


Figure 4.9.: Evolvability plotted against the evolvability-proxies LGE (left) and LGD (right), which select individuals which evenly distributed lineages and lineages that cover as many cells of the lineage-grid, respectively. Individuals from both the gene pool and the population of the last (200th) generation are shown. The individuals are colored by their fitness value. This reveals that most individuals that score high on the evolvability-proxy stayed in the area close to the beginning, where there is low fitness. Results are from two trials in the medium maze. Plots for the other heuristics and the other maze were similar.

4.6.3. Results for Using Viability to Estimate Evolvability

A possible reason for the failure of the above heuristics could be that the diversity of the lineage information is simply to old to be useful in predicting future success. Ancestors in the lineage were novel at their time, but maybe are not relevant in whether or not the individual will be able to produce novel behaviors itself. Therefore the second approach was to use the fate of an individual's children to determine whether it was evolvable or not. Therefore, viability (abbr. VIAB and explained in more detail in 3.9), measures a parent's success in contributing children to the gene pool. As such it is only defined for individuals that also already are in the gene pool. Individuals that contribute at an above-average rate to the gene pool receive high viability-scores. The viability of parents whose children never score high on the selection criteria is decreased. Although parents receive only credit in retrospect, the hope was that this becomes useful in situations when the overall progress is slow. Again, viability's effect was measured by switching it on and off and by measuring correlation with evolvability.

Noteworthy, the ability to produce children which are adopted into the gene pool is not necessarily related to producing children that are spread out over the maze. Therefore, the second heuristic was not expected to correlate or influence the evolvability as described above. Nevertheless, its effect on evolvability is rather positive. Figure 4.10 shows how evolvability develops over time, once when VIAB is selected at regular intervals and when it is selected for permanently (VIABP). When selecting for VIAB and RAR, the increase in average evolvability was steeper than on selection for RAR only. Fittingly, when VIAB was selected for continuously, the development of evolvability mirrored that of RAR, but at a higher level. As VIAB and RAR do not correlate ($r = -.29$, Table 4.5), they probably contribute to evolvability independently of each other. Figure 4.10 shows how evolvability and VIAB relate in the gene pool of the last generation. For this, the population was excluded as newly created children always have VIAB zero.

Target-objective	r_{EVO}	r_{RAR}	r_{FIT}
VIAB	0.14	-0.29	0.01
VIABP	0.14	-0.43	-0.1
RAR	0.44		0.8

Table 4.5.: The pearson correlation coefficient for the gene pool of the last generation of two trials in the medium maze. All objectives were selected for together with the rarity-objective. The current population is excluded as newly created children always have VIAB zero.

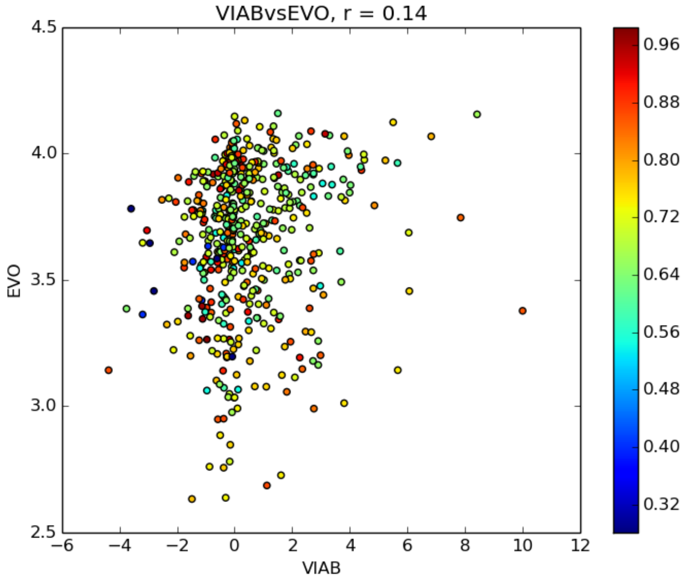


Figure 4.10.: Evolvability plotted against VIAB (feedback from children): Individuals from the gene pool of the last (200th) generation are shown, taken from two trials in the medium maze. The individuals are colored by their fitness value as indicated by the color bar on the right.

Experiments in the next section test whether that small increase in evolvability has a positive effect on the maze performance.

4.6.4. Effect of Viability on Performance

To test its effect on maze performance, viability was run in combination with rarity, curiosity and fitness. Results averaged over 30 runs in both mazes are shown in Figure 4.6. The results for the individual mazes are in the appendix B.5 and B.6. For rarity, the effect of viability is positive, but not significant. As rarity was already converging perfectly, there was not much room for improvement. Viability has a slightly positive effect on speed, which is not significant (see appendix B.7 and B.8). For the other two objectives the effect is negative, i.e. that it decreases their convergence rate, but also by a margin too small to be really significant.

To get an idea, how viability compares to the actual evolvability, I ran several selected objectives in combination with the evolvability objective, generating 200 extra offspring per individual at each timestep and mapping them into a grid as described in 3.7. For all objectives, the performance with evolvability or viability is not really different from the runs without them, so that it is not possible to say whether those objectives have a large impact in the first place. Due to the high computational effort needed the number of trials run is

Objectives	Convergence Rate	Solving Time	N
RAR/VIAB	1	126.35	30
RAR	1	132.32	30
CUR	0.5	397.54	30
CUR/VIAB	0.42	397.35	30
FIT	0.37	333.1	30
FIT/VIAB	0.33	255.85	25

Table 4.6.: Convergence rate and average generations until a solution was found when supplementing different objectives with viability. Maximum number of generations was 1000, population size 100 and the grid size for rarity and curiosity was set to 15. Results are averaged over both mazes.

Objectives	Convergence Rate	Solving Time	SD	N
RAR/VIAB	1	126.35	94.59	30
RAR	1	132.32	234.93	30
RAR/CUR/VIAB	0.92	270.69	214.1	30
RAR/CUR	0.9	327.39	254.47	30
RAR/EVO	0.89	266.71	83.29	13
RAR/CUR/EVO	0.82	322.33	218.31	20

Table 4.7.: A direct comparison of evolvability and its heuristic viability. Results were averaged over two mazes.

too few to draw any definite conclusions. Yet the impression is that, averaged over both mazes, evolvability does not help in solving the objectives, whereas viability does so slightly, if it is used parallel to rarity.

4.7. Biped Locomotion Experiment

To give more substance to the results of the maze-domain, it is attempted to reproduce them in the considerably more challenging biped domain. Developing controllers for bipedal robots is a popular problem in ML [7, 6] and has been used in previous ER works on novelty- and evolvability search [1, 91]. Due to the need for balance, coordination and the discovery of oscillatory patterns it is much harder than the maze domain. Whereas all behaviors in the maze domain lead the robot somewhere, the biped domain is much more brittle as controllers that do not manage to maintain the robot balanced will just fall over. For fitness functions, which typically measure the distance between the starting point and the location where the robot fell, the deceptiveness of the domain lies in the first steps. At the beginning of evolution, flinging the robot forwards such that it falls over with a lot of momentum leads to better results than trying to balance the robot and walking slowly. Therefore, even if a controller discovers a behavior that could contribute to balancing the robot, it might be discarded as it does not immediately contribute to propelling the robot forward. One approach to address this is to induce prior knowledge, such as making the fitness function enforce symmetry [115] or oscillatory patterns. But restricting the search space based on the experimenters expectations can hinder the evolutionary process and prevent innovation. For example, imposing oscillation can be helpful once the robot has started walking, but might make taking the first steps more difficult.

4.7.1. The Biped Robot Setup

To enable the nonlinear-dynamics of walking gaits, the ANN that controls the robot in this experiment is a continuous time recurrent neural network (CTRNN). CTRNNs can use different kinds of activation functions and encode network structures in a way that facilitates the discovery of parallel and oscillatory firing patterns. The ANN receives two inputs, one from each foot, which are one or zero depending on whether the foot touches the ground. Aside this sparse input, the net receives no further information about the robot's joints or its orientation relative for the ground, thus making the problem more difficult. The biped robot, depicted in Figure 4.11, has six degrees of freedom (DOF), each of which is controlled by one output neuron of the ANN. Two DOFs in each hip-joint (pitch and roll) and one in each of the knee-joints DOF (pitch). The range of the knee allows to bend the knee backwards. This is necessary because the robot walks on sphere-shaped feet with no ankles and lacks an upper body which could provide a counterweight for balancing. The ANNs outputs are interpreted as positions for the joints' angles, scaled to match the respective range. The difference between the desired position produced by the ANN and the actual position of the robot is given to a proportional controller which applies torque to reduce that disparity.

To encode the CTRNNs, the NEAT-algorithm is adjusted with self-adapting mutation rates, which influence how strong the individual connections are affected by mutations. These self-adapting mutation rates encode how likely a connection is subject to mutation and how strong these mutations are on average. Dependent on which mutation setting a node is following its connections could be exempted from mutations or subject to many more mutations. The mutation rates and the association between them and the nodes are themselves subject to mutation and can change over time. This is done because self-adaptation has been shown to favor evolvability in the biped domain [87]. The mutation settings and the other NEAT parameters are listed in the appendix. For a more detailed description of the self-adaptation feature, I refer to [87]. For more more information on the biped robot and the simulation parameters, see [1] where this experiment is adopted from.

Each controller is simulated for a fixed amount of time (eight seconds) in a physically realistic simulation. The population size was 100. For each run the robot that walked farthest is considered the solution. Therefore, the position of its center of mass is sampled once every 0.25 seconds and the distance is measured between the starting point and the last sample before ground contact.

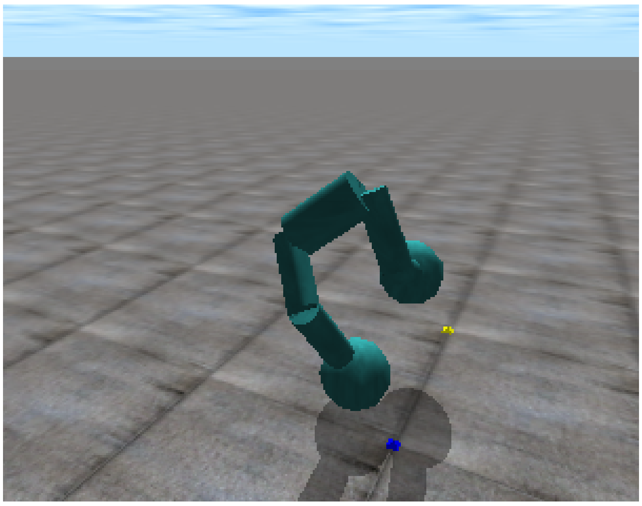


Figure 4.11.: The biped robot has 6 degrees of freedom: One in each knee (pitch) and two in each hip (pitch and roll). It is controlled by an ANN that receives two inputs, one from each foot, which are one or zero depending on whether or not the foot touches the ground. The ANN has one output unit for each DOF, whose values are translated to a torque value for a proportional controller, reducing the disparity between the position desired by the ANN and the actual position of the robot's joint. Its height is 0.33m. For a more detailed description of the experimental setup, see [1].

4.7.2. Objectives

The baseline objective will be fitness, which is measured as the straight-line distance between the robot's starting and end-point. Similar to the maze-domain, the robot's end-position is used as behavioral characterization for novelty and rarity. Rarity will be computed similar to 3.3, as the entropy of the grid cell the robot's end-position is discretized into. But as this domain does not have any constraints on where to move, another way to discretize the robots location can be considered. Taking into account the symmetry of the domain, an alternate way is to discretize the robots location with concentric circles instead of rectangular cells. As these circles measure the distance from the starting point, this adaptation of rarity is then equivalent to fitness frequency assignment and will therefore be referred to as FFA. For both rarity and FFA, preliminary tests showed that $0.5m$ was a good choice for the side-lengths of the bins. Thus only end-positions more than half a meter apart were considered different.

For curiosity, the position of the robot's center of mass was sampled every 0.25 seconds and recorded into a grid. To use entropy selection on the curiosity objective (frCUR), curiosity values were discretized with a bin side-length of 0.1. As the maximum curiosity for an eight-second walk is approximately 3.5, this results in 30 different values.

To see whether the rarity objective can make use of temporal information, temporal sampling of the robot's location is also used as behavioral characterization. Therefore the N_{samp} x,y-positions of the biped's center of mass are concatenated into a vector $x_1, y_1, x_2, y_2, ..., x_{N_{samp}}, y_{N_{samp}}$. If the robot falls over, all subsequent time samples are set to the starting location. Similar to the experiments in 4.4, the temporal-rarity (tRAR) will be computed by keeping track of the frequencies of the robot's location in different archives, one for each timestep. Viability and novelty can be adapted without any changes.

In order to keep the computational effort of determining an individual's novelty tractable, the size of the archive used to compute the distance to the 15 nearest neighbors was limited. Other approaches that limit the archive are to simply exclude the oldest members [1, 14]. As this allows the search to backtrack to behaviors already explored, the approach chosen here was slightly changed. Similarly, the archive's size was reduced to 1500 whenever it exceeded 2000 individuals. The 500 behaviors excluded were the ones in the regions least sparse of the archive. To determine that, the novelty computation only needs to be limited to include only those neighbors that already are in the archive. Thus, the backtracking problem was mitigated by excluding those members that contribute the least to representing where the search has already been.

4.7.3. Results

At first the objectives were run for 6000 generations. The only result visible at that stage was that curiosity, frequency-curiosity and rarity with viability were producing results much worse than fitness (see Figure B.1 in the appendix). Typical behaviors were doing one, at most two steps and falling over. As the differences between the other objectives were not interpretable the maximum number of generations was increased to 30.000. The deceptiveness of the domain lies in evolving a gait that can be used beyond the first few steps. Controllers that did not balance the robot could lunge it about 2m, just by applying a lot of force to the joints.

For the increased number of generations, the average maximum distance traveled in the last generation is shown in Table 4.8 and the progress made over the generations in Figure 4.12. The controllers evolved by fitness traveled an average distance of 2.9m ($SD = .68$) before falling over. An video analysis showed that this distance could be reached when the controller maintained the robot upright for long enough to execute about 2 or 3 steps. The only objectives that are at least 2 or more steps beyond that, are FFA (3.36m, $SD = .85$) and novelty (3.53m, $SD = .8$). Rarity and temporal rarity are at about the same distance as fitness with 2.88m ($SD = .63$) and 3.18 ($SD = .76$), respectively. All differences are not significant (see Table B.9 in the appendix), but due to the high number of generations, the number of trials per objective was relatively low (N=10), which makes it harder to obtain significant results. The maximum distance achieved over all trials were 4.2m for fitness, 4.6m for rarity and temporal rarity, 5.3m for FFA and 5.4m for novelty.

Although the differences in distance were small, a video-analysis of all the gaits evolved by an objective showed that the objectives tend to evolve different walking strategies. With fitness as objective, most of the controllers followed a rather short-sighted strategy: They created angular momentum by spinning on the spot and then propelled themselves in one direction, trying to avoid falling over as long as possible. Often this resulted in the robot spinning about its middle axes, while walking - a strategy hard to maintain for more than a few steps. Some would, prior to losing their balance, flex one leg very strongly to lunge themselves forwards in a cartwheel-like manner. 7 out of 10 of the best individuals would use such a tactic, only two were attempting a 'normal' bipedal gait that kept the robot oriented in the same direction. Similarly, when rarity was computed based only on the end position, 8 out of 10 evolved gaits implemented the spinning-strategy. When the objectives where such that it was possible to score high without moving far (tRAR, FFA and novelty), the majority of the gaits evolved followed the more natural strategy of walking straight ahead (6, 7 and 9 out of 10). Videos of the most successful gaits and the two strategies ('spinning' and 'natural') can be seen at https://www.youtube.com/watch?v=qGlM98PJ3wQ.

Objectives	Maximum Distance	SD	N
NOV	3.53	0.8	10
FFA	3.36	0.85	10
tRAR	3.18	0.76	10
FIT	2.93	0.68	10
RAR	2.88	0.63	10

Table 4.8.: Average maximum distance at each generation for the objectives in the biped domain. Distance is measured in meters. The maximum number of generations was 30.000.

5. Discussion

This work shows the usefulness of entropy to model diversity-based objectives, suggests a way to identify useful search directions in order to avoid convergence and highlights the difficulty of recognizing evolvability without explicitly testing it. This section discusses each of these implications in more depth by pointing out their limitations and if relevant, implications for future research. Finally, it concludes by reviewing the most important results in the light of very recent findings that reinterpret ER's role for the field of AI.

5.1. Entropy for Diversity-Objectives

Task-based objective functions are severely limited by local optima and the unpredictable shape of their fitness landscapes. The fitness paradigm constraints solutions to be built as a series of fitness-increasing mutations, which makes addressing complex problems even harder than they already are. An alternative route to guide the search is to define the search progress with respect to the immediate challenges posed by the environment of the population or agent. Diversity based objectives do this in the most general form, by simply trying to do as many different things possible. As entropy naturally measures the dispersion of a distribution, it is well-suited to measure such diversity.

In this work, I show this by using entropy to model two such diversity drives, curiosity and novelty, that have been shown previously to be able to solve deceptive problems. Curiosity measures the diversity of the robot's experience, which in the case of the tasks used in this work is equivalent to a diverse trajectory. Thus, to score high on curiosity the robots have to visit many different states during their lifetime. Because that is easier when the robot adapts to the environment, selection for curiosity often evolves robots that perform well.

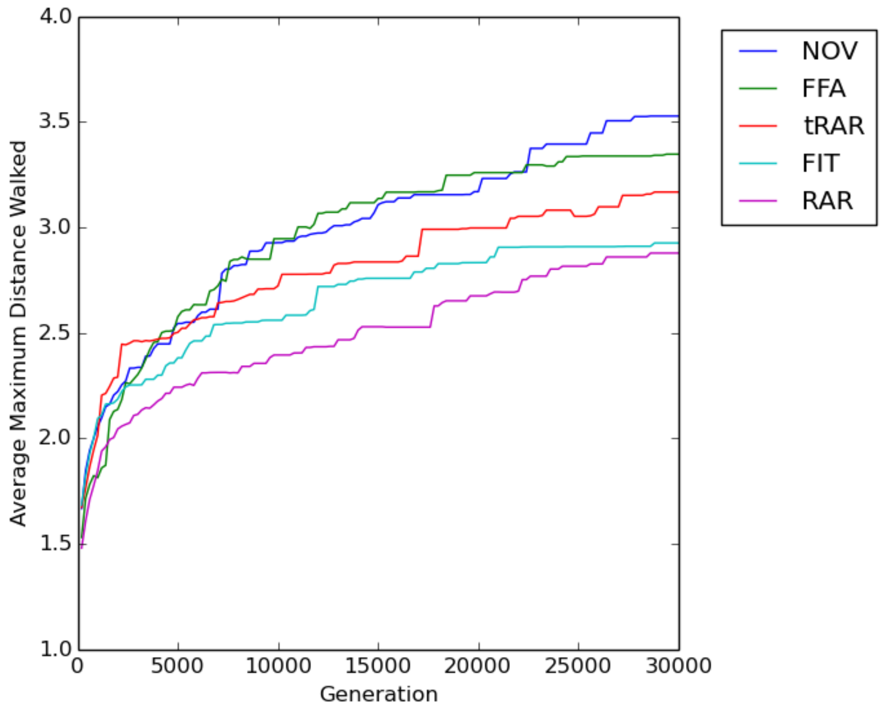


Figure 4.12.: Average maximum distance walked for the different objectives for each generation. Results were averaged over 10 trials.

In the previous work [45], curiosity was also modeled with entropy and applied to maze-navigation tasks, which it solved irrespective of their degree of deceptiveness. Aside from reproducing these findings, the contribution of this work was to test it on the more challenging biped domain. Here, curiosity selection was not able to overcome the deceptiveness of the domain and evolve a controller that maintains balance. As an individual's curiosity is not influenced by what its peers are doing, the population probably converged into one or few local curiosity-optima it was not able to escape from. Future work on curiosity might therefore try to account for that by including a way to measure whether the states a robot is visiting have been explored previously, or are also being explored by other members of the population. The latter would be a population-level curiosity. When the history of the population is also accounted for, curiosity then is equivalent to novelty. Therefore, the greatest value of curiosity lies in the fact that is conceptionally so simple that it can be computed on board, as shown in [45] even without global knowledge of the robot's position. Yet it's applicability for challenging domains remains questionable.

The other measure that was modeled was novelty. Novelty search [1] replaces the fitness objective with a measure of behavioral sparsity, so that it favors robots that behave different than their peers. The interesting and somewhat counterintuitive result is that although novelty search has no notion of a goal-behavior, it often finds it faster and more efficiently than search methods that explicitly try to do so. The reason for this is that novelty search is a divergent objective, effectuating an open-ended search that never converges. With an archive that represents where the search has already been, the objective of what is novel keeps changing over time. Thus, diversity is defined with respect to the past and present of the entire population. After all simple behaviors have been explored, doing something new requires doing something more complex, often by adapting favorable to the environment. In that process, the population discovers behaviors that would previously have been specified as goals. In order to use entropy for modeling novelty, an important trade-off is made: Whereas novelty typically uses a distance metric that returns qualitative distances (such as behavior A is x different from B) between behaviors, its equivalent modeled with entropy can only make nominal distinctions (behavior A is different from B or not). This is a consequence of having to partition the behavior space into a number of distinct regions in order to be able to interpret the search history as a distribution. The advantage of this is that the archive can represent the entire search and the computational effort remains the same, no matter how many individuals are in the archive. The disadvantage is that there is no difference between very novel and slightly novel behaviors, which is why its entropic adaptation was dubbed *rarity*. Also, when using higher dimensional behavioral characterizations (BCs), rarity's archive had to be split up into many archives, whereas novelty's distance computation remains the same. For the maze-domain, this trade-off did not have a negative effect and rarity actually found solutions a bit faster than novelty. In the more challenging biped domain, this trade-off was more significant. Novelty-search performed much better than fitness as it evolved controllers that walked much further and were of a higher quality than fitness-guided search. Rarity was as bad as fitness, both in terms of distance walked and the quality of the evolved gaits. Possibly, this was because rarities measure of distinguishing between behaviors was too coarse. With only the sparsity of the end-position relevant, the way for a controller to score high on rarity was to walk further than the other, leading the search into the same local optima as fitness. This could partially be redeemed by including the intermediate positions of the robot in extra archives. With this additional temporal information, rarity also evolved controllers that maintained balanced and walked with a natural gait.

Another observation on the novelty-rarity comparison in the biped domain is that novelty was computationally extremely demanding. Although the size of its archive was limited, it still required much longer than rarity. If n were the size of the archive, computing an individual's novelty score required n^2 distance computations to determine its nearest neighbours. This could be reduced to O(log n) by implementing the archive in a favorable way, such as using KD-Trees, but will not be as fast as rarity, for which the computation is a simple look-up in a table whose cost is in O(1).

Therefore, it is interesting to consider ways in which rarity could be improved by adding more archives. Rarities robustness to different resolutions of behavior discretizations (see section 4.3), suggests that rarity could also be used with more than one grid size simultaneously, each capturing the search at a different resolution. Behaviors at the edge of the population would thus receive different rarity-scores as behaviors in a completely uncharted region of the behavior space. This could mitigate rarity's problem of implementing an imprecise distance metric, without affecting it's capacity to represent the entire search. The experiments with naiveRAR (see section 4.4) show that using additional archives does not impair rarity, even if some of them are ill-designed: NaiveRAR used overly simplistic archives which failed to capture an important relation between the behavioral dimensions and which mostly described features of the behaviors that were irrelevant for the individuals performance, such as the robots x-location during its first steps. Yet, this only affected the time it took to evolve solutions. Thus, adding archives to rarity is conceptually simple, computationally cheap and adding irrelevant archives seems to not harm performance. Consequently, supplementing rarity with additional archives could be an interesting direction for future research, for instance by varying them in their discretization of the behavior to improve rarity's ability to differentiate them or by capturing other interesting aspects of it, hoping to evolve more complex behaviors.

5.2. Using Entropy to Inform Selection

At their core, rarity and novelty rely on evolutionary search to produce the undesirable behaviors more often than the interesting ones. For domains where this is not the case, such as mazes without walls, novelty can be supplemented with a minimum performance criteria, that all individuals have to fulfill. In nature this criterion is reproduction and survival. In EC this can be defined with by the experimenter and would be fitness. This has been shown to be successful in [62, 63]. With the search space pruned by such a criteria, the core assumption of the degenerate behaviors being found more often than the desirable ones should be of no further concern, as the majority of mutations usually is deleterious. Consequently, the frequency with which a behavior occurs can be used as a proxy for its interestingness. This approach includes the typical optimization methods which aim for a maximum objective value because high values are usually rare. Yet it implies no constraints on the way these values are to be reached. Instead, it dynamically changes what objective values are to be allocated more search resources. By keeping track of the search with an archive, unfruitful search directions can be identified and avoided.

Particularly interesting is that this concept can be applied to any objective. In previous work [33], it has been used to guide fitness-based search on a genetic programming problem. In this work, it is for the first time applied to ER-problems and to other objectives. In the maze domain, the fitness vs. rarity comparison (see 4.2 nicely illustrates the benefits of not defining by hand which behavior is desirable. Whereas fitness picks one particular point in the behavior space (the goal-area of the maze) and measures the distance to it, rarity defines as desirable simply what seldom has been found before. As long as the maze in not completely explored, there must always be a direction in which the population can move in order to do something new. Consequently, rarity evenly progresses through the maze and eventually finds not only the behavior specified by fitness, but also all the others.

The comparison of fitness vs. fitness frequency (see 4.5 illustrates that it is also better not to specify how a high objective value should be reached. Fitness implies a series of increasingly beneficial mutations and becomes trapped in the very deceptive dead-end of the harder maze. As fitness frequency (FFA) only aims for rare values, it has no disposition for high or low values and also no constraints on how these values should be reached. It thus can escape the local optima and solve the hard maze at a rate even higher than another state of the art method that explicitly favors diversity next to fitness (see Table 4.2). In the more challenging biped domain, FFA is conceptually equivalent to rarity with circular states and produces the best results after novelty. Also, the solutions evolved by FFA are of a high quality, i.e. 7 out of 10 walkers evolve a natural, forward-oriented gait. By oscillating back and forth between what fitness values are desirable, FFA overcomes the deceptiveness of the first steps. When selecting for controllers that walk far, it drives the search towards long distances. When periodically switching to selection for controllers that walk shorter distances it uses the same controllers, possibly discovering ways to maintain balance.

These findings are supported further by applying selection for rare values to curiosity. Although a diversity objective, curiosity is maximized and therefore susceptible to local optima in the curiosity-landscape. In both the maze- and biped domain, curiosity and its adaptation frequency curiosity (frCUR) perform equally well or bad, respectively.

The computational efficiency and ease of implementation make the use of selection for rarity a valuable method to inform the search and automatically balance exploration vs. exploitation. It can thus make convergent objectives into periodically divergent ones so that they can escape local optima in their respective landscape and are more suited for open-ended search. A main limitation of this approach is that the objective value has to be discretized. This is especially difficult if one does not know a priori in what range the objective values are likely to be. Similar to the grid-discretization of rarity, it could be possible to alleviate this by using grids with different resolutions. A direction for future research would thus be to investigate the effect of the bin size on the search progress and potentially use it to affect the degree of exploration that is executed upon the different parts of an objective range. If, for example, the high values of an objective value are deemed more interesting by the experimenter, they could be discretized with a higher resolution. Another possible direction for future research would be to empty the archive at certain points during the search. This would provide a chance for the more evolvable individuals to rediscover previously well-explored values, akin to the random extinction approach proposed by Lehman in [79].

5.3. Approximating Evolvability

All of the previous methods, rarity, novelty and curiosity, are concerned with maintaining diversity by recognizing it once it is produced. Thus, they rely on the efficiency of the encoding method and the mutation operator to produce phenotypic variation. Yet some individuals, although behaving in a way that scores high on a diversity objective, will not be able to produce children that contribute to the search. For other individuals it might be the opposite: They behave rather usual but their children tend to do very diverse and new things. For the former this might be because the behavior of the solution is very fragile, i.e. small disturbances of the parameters completely mess it up. For the latter, the behavior might be very reusable. If, for example, the robot discovered oscillatory patterns, the children might do very different things depending on what they use the pattern for.

The key feature that distinguishes the two individuals is their evolvability, which describes their potential for future evolution. Typical approaches that measure it exactly [1, 91, 82, 88] do many additional evaluations, which is why evolvability is ill-suited to be used directly as objective. Yet, doing so would benefit the search by increasing diversity and evolving solutions that are better able to generalize, as has been shown in a computationally extensive effort by Mengistu et al [91]. Thus, selection for evolvability could have the same effect as regularization-methods in ML [32] and therefore has the potential to increase the complexity of problems ER can address.

In this work, I tried to avoid doing extra simulations by instead using the many individuals that are discarded during search. Especially, I tried to infer an individual's evolvability from its lineage. The hypothesis tested was that individuals with a diverse lineage are likely to be evolvable. Approaches taken are presented in detail in section 4.6.1 and included varying the ancestors represented in the lineage (ancestors or ancestors + one step mutations), the length of the lineage (all or only the most recent), the description of behavior that was passed on from parent to child (end-position or complete path) and the measure used to differentiate between behaviors (entropy or range of the distribution). Yet, none of the developed heuristics was successful.

One possible reason for that failure might be that different locations in the maze have inherently different potential for evolvability, i.e. they are particularly suited or ill-suited to produce diverse lineages. Especially in the large starting area of the medium maze, mutations have a high chance to change the robot's end location to a different grid cell, leading to a diverse lineage in the process. These individuals then register as highly evolvable, although they could be the product of a series of completely deleterious mutations. Contrasting to that, the individuals that actually are evolvable might have progressed to the narrower areas of the mazes. As those are more restricted, their lineages will occupy less grid cells and their progress through that area would register in the grid as a more compact area.

Future work on heuristics for evolvability should therefore account for the environment the individual is in. In nature, the living conditions in different niches impose differently strict or forgiving conditions on the genetic or phenotypic variability that one population can maintain without going extinct. Therefore it does not make sense to directly compare these among species of different niches.

5.4. Conclusion

Until recently, the prevalent abstraction of evolution's algorithm was that of an optimizer, finding the best solution to one problem. Following this view, it is in direct concurrence to many methods from machine learning, most of them having a longer tradition and a more solid, elaborated theoretical framework. The more recent view, represented in works on quality-diversity algorithms [14, 64, 15, 9] is that evolutionary algorithms should aim to reproduce evolution's potential for finding a diverse range of possible solutions to one or many related problems. According to this view, EAs main purpose is to help the experimenter understand the problem by showing how some of its features relate to each other and how they influence the overall performance. An argument for this, or maybe rather against the first perspective, is given by the fact that the best solutions found when searching for diversity are often better than the ones found by optimization algorithms. First shown by novelty search, which found solutions only using a measure of behavioral novelty, this is reaffirmed by the results of this thesis, which abstracts novelty by partitioning the behavior space in a principled way, and using behavioral characterizations of different lengths and composition. Recent work with the MAP-elites algorithm [64] abstracted that further, showing that exploring the behavior space can be explored by means of other measures as well, for instance the percentage of a robot's bone or muscle tissue. The reason these approaches work is that high-fitness solutions often traverse a path through many areas of that space, partly with very low fitness potentials - a path that can not be found by fitness-guided search. Hence, searching for many solutions simultaneously is better than just searching for one, which sooner or later leads into a local optimum.

This thesis supports that point by pitting classical optimization against a new entropy-based selection method for fitness and behavioral characterizations of varying degrees of usefulness and dimensionality. Instead of defining a priori what range of an objective value is desirable, this method simply aims for what range or values so far been achieved least often. Thus, it naturally includes the past search results in order to allocate to find as many behaviors possible and avoid becoming trapped by local optima. This superiority of the new method is shown to work on a deceptive navigation task and a challenging biped locomotion task. In both domains, it exploits the fact that degenerate behaviors are pervasive in search spaces and will be found more often than the desirable ones, effectively using the frequency (measured with entropy) of an objective value as a heuristic for its interestingness.

Although not done in this work, this method could easily be adapted to be a quality diversity method, simply by saving the fittest individual of each cell of the archive-grid. A very recent adaptation [14] of the most successful quality diversity algorithm MAP-elites [64] uses two different behavioral characterization simultaneously, each of which captured by a separate archive,and achieves good results on very challenging navigation tasks. This work shows that it is possible to also use a much higher number of archives, possibly enabling the experimenter to capture higher dimensional descriptions of behavior or understand the relation between more features of the robot's interaction. Ultimately, both of these enhance the reach of ER by leading to a better understanding of ER's problems and possibly to more complex end-products of evolutionary search.

Bibliography

[1] J. Lehman and K. O. Stanley, "Abandoning objectives: Evolution through the search for novelty alone," *Evolutionary computation*, vol. 19, no. 2, pp. 189–223, 2011.

[2] "Neuroevolution of augmenting topologies, available at https://www.cs.ucf.edu/k̃stanley/neat.html, accessed 2016-08-25."

[3] S. Nolfi and D. Floreano, *Evolutionary robotics.* Mit Press Cambridge, 2000.

[4] D. Floreano and F. Mondada, "Evolution of homing navigation in a real mobile robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 3, pp. 396–407, 1996.

[5] J.-B. Mouret and S. Doncieux, "Encouraging behavioral diversity in evolutionary robotics: An empirical study," *Evolutionary computation*, vol. 20, no. 1, pp. 91–133, 2012.

[6] G. McHale and P. Husbands, "Gasnets and other evolvable neural networks applied to bipedal locomotion," *From Animals to Animats*, vol. 8, pp. 163–172, 2004.

[7] D. Hein, M. Hild, and R. Berger, "Evolution of biped walking using neural oscillators and physical simulation," in *Robot Soccer World Cup*, pp. 433–440, Springer, 2007.

[8] J. Yosinski, J. Clune, D. Hidalgo, S. Nguyen, J. Zagal, and H. Lipson, "Evolving robot gaits in hardware: the HyperNEAT generative encoding vs. parameter optimization," in *Proceedings of the 20th European Conference on Artificial Life*, pp. 890–897, Citeseer, 2011.

[9] J. Lehman and K. O. Stanley, "Evolving a diversity of virtual creatures through novelty search and local competition," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pp. 211–218, ACM, 2011.

[10] R. Pfeifer, F. Iida, and J. Bongard, "New Robotics: Design Principles for Intelligent Systems," *Artificial Life*, vol. 11, pp. 99–120, Jan. 2005.

[11] J. B. Pollack and H. Lipson, "The GOLEM project: Evolving hardware bodies and brains," in *Evolvable Hardware, 2000. Proceedings. The Second NASA/DoD Workshop on*, pp. 37–42, IEEE, 2000.

[12] S. A. Bailey, J. G. Cham, M. R. Cutkosky, and R. J. Full, "Comparing the locomotion dynamics of the cockroach and a shape deposition manufactured biomimetic hexapod," in *Experimental Robotics VII*, pp. 239–248, Springer, 2001.

[13] S. Doncieux, J.-B. Mouret, N. Bredeche, and V. Padois, "Evolutionary Robotics: Exploring New Horizons," in *New Horizons in Evolutionary Robotics* (J. Kacprzyk, S. Doncieux, N. Bredche, and J.-B. Mouret, eds.), vol. 341, pp. 3–25, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[14] J. K. Pugh, L. B. Soros, and K. O. Stanley, "Quality Diversity: A New Frontier for Evolutionary Computation," *Frontiers in Robotics and AI*, vol. 3, 2016.

[15] J. K. Pugh, L. B. Soros, P. A. Szerlip, and K. O. Stanley, "Confronting the Challenge of Quality Diversity," pp. 967–974, ACM Press, 2015.

[16] J. R. Koza, *Genetic programming: on the programming of computers by means of natural selection*, vol. 1. MIT press, 1992.

[17] J.-B. Mouret and S. Doncieux, "Overcoming the bootstrap problem in evolutionary robotics using behavioral diversity," in *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*, pp. 1161–1168, IEEE, 2009.

[18] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, *Machine learning: An artificial intelligence approach.* Springer Science & Business Media, 2013.

[19] T. Weise, R. Chiong, and K. Tang, "Evolutionary Optimization: Pitfalls and Booby Traps," *Journal of Computer Science and Technology*, vol. 27, pp. 907–936, Sept. 2012.

[20] B. Sareni and L. Krhenbhl, "Fitness sharing and niching methods revisited," *Evolutionary Computation, IEEE Transactions on*, vol. 2, no. 3, pp. 97–106, 1998.

[21] L. Trujillo, G. Olague, E. Lutton, and F. F. de Vega, "Behavior-based speciation for evolutionary robotics," in *Proceedings of the 10th annual conference on Genetic and evolutionary computation*, pp. 297–298, ACM, 2008.

[22] S. Doncieux and J.-B. Mouret, "Behavioral diversity measures for evolutionary robotics," in *Evolutionary Computation (CEC), 2010 IEEE Congress on*, pp. 1–8, IEEE, 2010.

[23] D. E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," in *Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms*, pp. 41–49, Hillsdale, NJ: Lawrence Erlbaum, 1987.

[24] M. Hutter and S. Legg, "Fitness uniform optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 10, no. 5, pp. 568–589, 2006.

[25] G. S. Hornby, "ALPS: the age-layered population structure for reducing the problem of premature convergence," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp. 815–822, ACM, 2006.

[26] S. Doncieux and J.-B. Mouret, "Beyond black-box optimization: a review of selective pressures for evolutionary robotics," *Evolutionary Intelligence*, vol. 7, pp. 71–93, Aug. 2014.

[27] A. D. Wissner-Gross and C. E. Freer, "Causal Entropic Forces," *Physical Review Letters*, vol. 110, Apr. 2013.

[28] C. Salge, C. Glackin, and D. Polani, "EmpowermentAn Introduction," in *Guided Self-Organization: Inception* (M. Prokopenko, ed.), vol. 9, pp. 67–114, Berlin, Heidelberg: Springer Berlin Heidelberg, 2014.

[29] T. Jung, D. Polani, and P. Stone, "Empowerment for continuous agent–environment systems," *Adaptive Behavior*, vol. 19, pp. 16–39, Feb. 2011.

[30] C. Salge, C. Glackin, and D. Polani, "Changing the Environment Based on Empowerment as Intrinsic Motivation," *Entropy*, vol. 16, pp. 2789–2819, May 2014.

[31] J. F. Brookfield, "Evolution and evolvability: celebrating Darwin 200," *Biology Letters*, vol. 5, pp. 44–46, Feb. 2009.

[32] K. Kouvaris, J. Clune, L. Kounios, M. Brede, and R. A. Watson, "How Evolution Learns to Generalise: Principles of under-fitting, over-fitting and induction in the evolution of developmental organisation," *arXiv preprint arXiv:1508.06854*, 2015.

[33] T. Weise, M. Wan, P. Wang, K. Tang, A. Devert, and X. Yao, "Frequency Fitness Assignment," *IEEE Transactions on Evolutionary Computation*, vol. 18, pp. 226–243, Apr. 2014.

[34] F. Gomez and R. Miikkulainen, "Incremental Evolution of Complex General Behavior," *Adaptive Behavior*, vol. 5, pp. 317–342, Jan. 1997.

[35] M. A. Lewis, A. H. Fagg, and A. Solidum, "Genetic programming approach to the construction of a neural network for control of a walking robot," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, pp. 2618–2623, IEEE, 1992.

[36] S. Nolfi, "Evolving non-trivial behaviors on real robots: A garbage collecting robot," *Robotics and Autonomous Systems*, vol. 22, no. 3, pp. 187–198, 1997.

[37] N. F. McPhee and N. J. Hopper, "Analysis of genetic diversity through population history," in *Proceedings of the Genetic and Evolutionary Computation Conference*, vol. 2, pp. 1112–1120, Citeseer, 1999.

[38] T. Chena, K. Tangb, G. Chenb, and X. Yaob, "A Large Population Size Can Be Unhelpful in Evolutionary Algorithms6," 2011.

[39] E. K. Burke, S. Gustafson, G. Kendall, and N. Krasnogor, "Is increased diversity in genetic programming beneficial? An analysis of lineage selection," in *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*, vol. 2, pp. 1398–1405, IEEE, 2003.

[40] J. Paredis, "Coevolutionary Computation," *Artificial Life*, vol. 2, pp. 355–375, Jan. 1995.

[41] G. Bason and T. Ziemke, "Competitive Co-evolution of Predator and Prey Sensory-Motor Systems," in *Applications of Evolutionary Computing* (G. Goos, J. Hartmanis, J. van Leeuwen, S. Cagnoni, C. G. Johnson, J. J. R. Cardalda, E. Marchiori, D. W. Corne, J.-A. Meyer, J. Gottlieb, M. Middendorf, A. Guillot, G. R. Raidl, and E. Hart, eds.), vol. 2611, pp. 605–615, Berlin, Heidelberg: Springer Berlin Heidelberg, 2003.

[42] C. Darwin, *The Origin of Species.* John Murry London, 1929.

[43] H. Bunke and K. Shearer, "A graph distance metric based on the maximal common subgraph," *Pattern Recognition Letters*, vol. 19, pp. 255–259, Mar. 1998.

[44] J.-B. Mouret and S. Doncieux, "Using behavioral exploration objectives to solve deceptive problems in neuro-evolution," in *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pp. 627–634, ACM, 2009.

[45] P. Delarboulas, M. Schoenauer, and M. Sebag, "Open-Ended Evolutionary Robotics: An Information Theoretic Approach," pp. 334–343, Springer Berlin Heidelberg, Sept. 2010.

[46] F. J. Gomez, "Sustaining diversity using behavioral information distance," in *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pp. 113–120, ACM, 2009.

[47] B. Inden, Y. Jin, R. Haschke, H. Ritter, and B. Sendhoff, "An examination of different fitness and novelty based selection methods for the evolution of neural networks," *Soft Computing*, vol. 17, pp. 753–767, May 2013.

[48] J. Lehman, K. O. Stanley, and R. Miikkulainen, "Effective diversity maintenance in deceptive domains," in *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, pp. 215–222, ACM, 2013.

[49] J. Li, J. Storie, and J. Clune, "Encouraging creative thinking in robots improves their ability to solve challenging problems," pp. 193–200, ACM Press, 2014.

[50] K. M. Malan and A. P. Engelbrecht, "Quantifying ruggedness of continuous landscapes using entropy," pp. 1440–1447, IEEE, May 2009.

[51] C. M. Reidys and P. F. Stadler, "Neutrality in fitness landscapes," *Applied Mathematics and Computation*, vol. 117, pp. 321–350, Jan. 2001.

[52] J. Lehman and K. O. Stanley, "Novelty search and the problem with objectives," in *Genetic Programming Theory and Practice IX*, pp. 37–56, Springer, 2011.

[53] J. Lehman, *Evolution Through the Search for Novelty.* PhD thesis, Citeseer, 2012.

[54] J. Gomes, P. Urbano, and A. L. Christensen, "Evolution of swarm robotics systems with novelty search," *Swarm Intelligence*, vol. 7, pp. 115–144, Sept. 2013.

[55] J. Gomes, P. Mariano, and A. L. Christensen, "Novelty Search in Competitive Coevolution," *arXiv:1407.0576 [cs]*, July 2014. arXiv: 1407.0576.

[56] S. Risi, C. E. Hughes, and K. O. Stanley, "Evolving plastic neural networks with novelty search," *Adaptive Behavior*, vol. 18, pp. 470–491, Dec. 2010.

[57] D. A. V. Veldhuizen and G. B. Lamont, "Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art," *Evolutionary Computation*, vol. 8, pp. 125–147, June 2000.

[58] D. Brockhoff, T. Friedrich, N. Hebbinghaus, C. Klein, F. Neumann, and E. Zitzler, "Do Additional Objectives Make a Problem Harder?," in *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*, GECCO '07, (New York, NY, USA), pp. 765–772, ACM, 2007.

[59] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review.," in *IEEE congress on evolutionary computation*, pp. 2419–2426, Citeseer, 2008.

[60] J.-B. Mouret, "Novelty-based multiobjectivization," in *New horizons in evolutionary robotics*, pp. 139–154, Springer, 2011.

[61] J. Gomes, P. Mariano, and A. L. Christensen, "Devising Effective Novelty Search Algorithms: A Comprehensive Empirical Study," pp. 943–950, ACM Press, 2015.

[62] J. Lehman and K. O. Stanley, "Revising the evolutionary computation abstraction: minimal criteria novelty search," p. 103, ACM Press, 2010.

[63] J. Gomes, P. Urbano, and A. L. Christensen, "Progressive minimal criteria novelty search," in *Advances in Artificial IntelligenceIBERAMIA 2012*, pp. 281–290, Springer, 2012.

[64] J.-B. Mouret and J. Clune, "Illuminating search spaces by mapping elites," *arXiv preprint arXiv:1504.04909*, 2015.

[65] J. Clune, J.-B. Mouret, and H. Lipson, "The evolutionary origins of modularity," *Proceedings of the Royal Society B: Biological Sciences*, vol. 280, pp. 20122863–20122863, Jan. 2013.

[66] J. Clune, B. E. Beckmann, C. Ofria, and R. T. Pennock, "Evolving coordinated quadruped gaits with the HyperNEAT generative encoding," in *2009 IEEE Congress on Evolutionary Computation*, pp. 2764–2771, IEEE, 2009.

[67] A. Cully, J. Clune, and J. B. Mouret, "Robots that can adapt like natural animals. arXiv preprint," *arXiv preprint arXiv:1407.3501*, vol. 2, 2014.

[68] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 427–436, IEEE, 2015.

[69] J. Lehman, S. Risi, and J. Clune, "Creative Generation of 3d Objects with Deep Learning and Innovation Engines," in *Proceedings of the International Conference on Computational Creativity*, 2016.

[70] S. J. Gould, "Full house: the spread of excellence from Plato to Darwin," *Senior Managing Editor*, vol. 5, no. 2, p. 68, 1998.

[71] D. C. Dennett, "Darwin's dangerous idea," *The Sciences*, vol. 35, no. 3, pp. 34–40, 1995.

[72] M. Pigliucci, "Is evolvability evolvable?," *Nature Reviews Genetics*, vol. 9, no. 1, pp. 75–82, 2008.

[73] W. A, “Robustness, evolvability, and neutrality. - PubMed - NCBI.”

[74] M. Kirschner and J. Gerhart, “Evolvability,” *Proceedings of the National Academy of Sciences*, vol. 95, pp. 8420–8427, July 1998.

[75] G. P. Wagner and L. Altenberg, “Perspective: Complex Adaptations and the Evolution of Evolvability,” *Evolution*, vol. 50, no. 3, pp. 967–976, 1996.

[76] J. M. Smith and E. Szathmary, *The major transitions in evolution.* Oxford University Press, 1997.

[77] A. Wagner, “Robustness, evolvability, and neutrality,” *FEBS Letters*, vol. 579, pp. 1772–1778, Mar. 2005.

[78] J. Lehman and K. O. Stanley, “Evolvability is inevitable: increasing evolvability without the pressure to adapt,” *PloS one*, vol. 8, no. 4, p. e62186, 2013.

[79] J. Lehman and R. Miikkulainen, “Enhancing Divergent Search through Extinction Events,” pp. 951–958, ACM Press, 2015.

[80] T. Day, “Information entropy as a measure of genetic diversity and evolvability in colonization,” *Molecular Ecology*, vol. 24, pp. 2073–2083, May 2015.

[81] J. Reisinger, K. O. Stanley, and R. Miikkulainen, “Towards an empirical measure of evolvability,” in *Proceedings of the 7th annual workshop on Genetic and evolutionary computation*, pp. 257–264, ACM, 2005.

[82] D. Tarapore and J.-B. Mouret, “Evolvability signatures of generative encodings: beyond standard performance benchmarks,” *Information Sciences*, vol. 313, pp. 43–61, 2015.

[83] N. Kashtan, E. Noor, and U. Alon, “Varying environments can speed up evolution,” *Proceedings of the National Academy of Sciences*, vol. 104, no. 34, pp. 13711–13716, 2007.

[84] J. Reisinger and R. Miikkulainen, “Selecting for evolvable representations,” in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp. 1297–1304, ACM, 2006.

[85] J.-B. Mouret and S. Doncieux, “Incremental evolution of animats behaviors as a multi-objective optimization,” in *International Conference on Simulation of Adaptive Behavior*, pp. 210–219, Springer, 2008.

[86] N. Kashtan and U. Alon, “Spontaneous evolution of modularity and network motifs,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 39, pp. 13773–13778, 2005.

[87] J. Lehman and K. O. Stanley, “Improving evolvability through novelty search and self-adaptation,” in *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pp. 2693–2700, IEEE, 2011.

[88] B. Fowler and W. Banzhaf, “Modelling Evolvability in Genetic Programming,” in *Genetic Programming*, pp. 215–229, Springer, 2016.

[89] L. Altenberg and others, “The evolution of evolvability in genetic programming,” *Advances in genetic programming*, vol. 3, pp. 47–74, 1994.

[90] B. Wilder and K. Stanley, “Reconciling explanations for the evolution of evolvability,” *Adaptive Behavior*, vol. 23, no. 3, pp. 171–179, 2015.

[91] H. Mengistu, J. Lehman, and J. Clune, “Evolvability Search: Directly Selecting for Evolvability in order to Study and Produce It,”

[92] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*, vol. 1. MIT press Cambridge, 1998.

[93] R. Der, U. Steinmetz, and F. Pasemann, “Homeokinesis - A new principle to back up evolution with learning,” Feb. 1999.

[94] R. Der, “Selforganized robot behavior from the principle of homeokinesis,” *Proc. workhop soave*, pp. 39–46, 2000.

[95] G. Martius, R. Der, and N. Ay, “Information driven self-organization of complex robotic behaviors,” *PloS one*, vol. 8, no. 5, p. e63400, 2013.

[96] A. S. Klyubin, D. Polani, and C. L. Nehaniv, “All else being equal be empowered,” in *Advances in Artificial Life*, pp. 744–753, Springer, 2005.

[97] A. S. Klyubin, D. Polani, and C. L. Nehaniv, “Keep Your Options Open: An Information-Based Driving Principle for Sensorimotor Systems,” *PLoS ONE*, vol. 3, p. e4018, Dec. 2008.

[98] University of Hertfordshire, C. Salge, C. Glackin, and D. Polani, “Empowerment and State-dependent Noise - An Intrinsic Motivation for Avoiding Unpredictable Agents,” pp. 118–125, MIT Press, Sept. 2013.

[99] R. P. Mann and R. Garnett, “The entropic basis of collective behaviour,” *Journal of The Royal Society Interface*, vol. 12, pp. 20150037–20150037, Apr. 2015.

[100] J. P. Rosca, “Entropy-driven adaptive representation,” in *Proceedings of the workshop on genetic programming: From theory to real-world applications*, vol. 9, pp. 23–32, Citeseer, 1995.

[101] S.-H. Liu, M. Mernik, and B. R. Bryant, “To explore or to exploit: An entropy-driven approach for evolutionary algorithms,” *International journal of knowledge-based and intelligent engineering systems*, vol. 13, no. 3, p. 185, 2009.

[102] L. San Jos-Revuelta, “A new adaptive genetic algorithm for fixed channel assignment,” *Information Sciences*, vol. 177, pp. 2655–2678, July 2007.

[103] K. Stanley, B. Bryant, and R. Miikkulainen, “Real-Time Neuroevolution in the NERO Video Game,” *IEEE Transactions on Evolutionary Computation*, vol. 9, pp. 653–668, Dec. 2005.

[104] “Joel lehman on github, https://github.com/jal278/mazerobot-python, accessed 2016-10-25.”

[105] “my github, https://github.com/djudjuu/mazerobot-python, accessed 2016-10-25.”

[106] K. O. Stanley and R. Miikkulainen, “Evolving neural networks through augmenting topologies,” *Evolutionary computation*, vol. 10, no. 2, pp. 99–127, 2002.

[107] J. Lehman and K. O. Stanley, “Efficiently evolving programs through the search for novelty,” in *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pp. 837–844, ACM, 2010.

[108] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, “A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II,” *Lecture notes in computer science*, vol. 1917, pp. 849–858, 2000.

[109] J. Knowles and D. Corne, “The pareto archived evolution strategy: A new baseline algorithm for pareto multiobjective optimisation,” in *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, vol. 1, IEEE, 1999.

[110] G. Rudolph, "Evolutionary search under partially ordered sets," *Dept. Comput. Sci./LS11, Univ. Dortmund, Dortmund, Germany, Tech. Rep. CI-67/99*, 1999.

[111] E. Zitzler, "Evolutionary algorithms for multiobjective optimization: Methods and applications," 1999.

[112] C. K. Oei, D. E. Goldberg, and S.-J. Chang, "Tournament selection, niching, and the preservation of diversity," *Urbana*, vol. 51, p. 61801, 1991.

[113] D. Floreano and C. Mattiussi, *Bio-inspired artificial intelligence: theories, methods, and technologies.* MIT press, 2008.

[114] N. Cheney, R. MacCurdy, J. Clune, and H. Lipson, "Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding," in *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, pp. 167–174, ACM, 2013.

[115] B. Allen and P. Faloutsos, "Complex networks of simple neurons for bipedal locomotion," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4457–4462, IEEE, 2009.

[116] A. Baranes and P.-Y. Oudeyer, "R-IAC: Robust Intrinsically Motivated Exploration and Active Learning," *IEEE Transactions on Autonomous Mental Development*, vol. 1, pp. 155–169, Oct. 2009.

[117] A. Baranes and P.-Y. Oudeyer, "Robust intrinsically motivated exploration and active learning," in *IEEE 8th International Conference on Development and Learning, 2009. ICDL 2009*, pp. 1–6, June 2009.

[118] S. Mohamed and D. J. Rezende, "Variational Information Maximisation for Intrinsically Motivated Reinforcement Learning," pp. 2116–2124, 2015.

[119] M. Karl, J. Bayer, and P. van der Smagt, "Efficient Empowerment," Sept. 2015.

[120] T. Schnier and X. Yao, "Using multiple representations in evolutionary algorithms," in *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, vol. 1, pp. 479–486, IEEE, 2000.

[121] T. Hu, W. Banzhaf, T. Hu, and W. Banzhaf, "Evolvability and Speed of Evolutionary Algorithms in Light of Recent Developments in Biology, Evolvability and Speed of Evolutionary Algorithms in Light of Recent Developments in Biology," *Journal of Artificial Evolution and Applications, Journal of Artificial Evolution and Applications*, vol. 2010, 2010, June 2010.

[122] L. Altenberg, "The schema theorem and Prices theorem," *Foundations of genetic algorithms*, vol. 3, pp. 23–49, 1995.

[123] *The Neutral Theory of Molecular Evolution.*

[124] V. Sperati, V. Trianni, and S. Nolfi, "Mutual Information As a Task-Independent Utility Function for Evolutionary Robotics," in *Guided Self-Organization: Inception* (M. Prokopenko, ed.), vol. 9, pp. 389–414, Berlin, Heidelberg: Springer Berlin Heidelberg, 2014.

[125] C. Segura, C. A. Coello Coello, G. Miranda, and C. Len, "Using multi-objective evolutionary algorithms for single-objective optimization," *4OR*, vol. 11, pp. 201–228, Sept. 2013.

[126] B.-T. Zhang, "Information-theoretic objective functions for lifelong learning," tech. rep., DTIC Document, 2012.

[127] D. Shorten and G. Nitschke, "How evolvable is novelty search?," in *Evolvable Systems (ICES), 2014 IEEE International Conference on*, pp. 125–132, IEEE, 2014.

[128] F. Glover, "Tabu SearchPart I," *ORSA Journal on Computing*, vol. 1, pp. 190–206, Aug. 1989.

[129] R. Velez and J. Clune, "Novelty search creates robots with general skills for exploration," pp. 737–744, ACM Press, 2014.

[130] J. Gomes, P. Urbano, and A. L. Christensen, "Progressive Minimal Criteria Novelty Search," in *Advances in Artificial Intelligence IBERAMIA 2012* (D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, J. Pavn, N. D. Duque-Mndez, and R. Fuentes-Fernndez, eds.), vol. 7637, pp. 281–290, Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[131] Jun He and Xin Yao, "From an individual to a population: an analysis of the first hitting time of population-based evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 495–511, Oct. 2002.

[132] C. L. Nehaniv, "Measuring evolvability as the rate of complexity increase," in *Artificial Life VII Workshop Proceedings*, pp. 55–57, 2000.

[133] A. Wagner, "Robustness and evolvability: a paradox resolved," *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 275, no. 1630, pp. 91–100, 2008.

[134] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary computation*, vol. 3, no. 2, pp. 82–102, 1999.

[135] J. Clune, K. O. Stanley, R. T. Pennock, and C. Ofria, "On the performance of indirect encoding across the continuum of regularity," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 3, pp. 346–367, 2011.

[136] G. S. Hornby, H. Lipson, and J. B. 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.

[137] T. Reil and P. Husbands, "Evolution of central pattern generators for bipedal walking in a real-time physics environment," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 159–168, 2002.

A. Parameters

NEAT-parameter	Maze Experiment	Biped Experiment
trait_param_mut_prob	0.5	0.5
trait_mutation_power	1	0.1
linktrait_mut_sig	1	1
nodetrait_mut_sig	0.5	0.5
weigh_mut_power	2.5	2.5
recur_prob	0	0.5
disjoint_coeff	1	1
excess_coeff	1	1
mutdiff_coeff	0.4	3
mutate_only_prob	0.25	1
mutate_random_trait_prob	0.1	0.2
mutate_link_trait_prob	0.1	0.1
mutate_node_trait_prob	0.1	0.5
mutate_link_weights_prob	0.9	0.6
mutate_toggle_enable_prob	0	0.1
mutate_gene_reenable_prob	0	0.05
mutate_add_node_prob	0.03	0.005
mutate_add_link_prob	0.05	0.06
recur_only_prob	0	0.5
pop_size	100	100
newlink_tries	20	20
babies_stolen	0	0
time_const_mut_power	-	0.2
time_const_mut_prob	-	0.3
bias_mut_power	-	0.2
bias_mut_prob	-	0.3

Table A.1.: Parameters for the NEAT algorithm. Names are unaltered from the NEAT user page [2].

B. Experimental Results

	RAR	NOV	RAR/CUR	FIT	FIT/DIV	CUR
RAR	-	-	-	-	-	-
NOV	p=0.348	-	-	-	-	-
RAR/CUR	p=0.000*	p=0.000*	-	-	-	-
FIT	p=0.006*	p=0.055	-	-	-	-
FIT/DIV	p=0.000*	p=0.002*	-	-	-	-
CUR	p=0.004*	p=0.011*	p=0.872	p=0.224	p=0.321	-

Table B.1.: Significance p-values for the differences in solving speed in the medium maze (see 4.1). P-values are only shown if the column-objective outperformed the row-objective both in terms of convergence and speed. The statistical test used is a two-tailed Mann-Whitney-U test.

	RAR	NOV	RAR/CUR	CUR	FIT/DIV	FIT
RAR	-	-	-	-	-	-
NOV	p=0.473	-	-	-	-	-
RAR/CUR	p=0.006*	p=0.024*	-	-	-	-
CUR	p=0.001*	p=0.000*	p=0.061	-	-	-
FIT/DIV	p=0.001*	p=0.001*	p=0.009*	p=0.080	-	-
FIT	p=0.000*	p=0.000*	p=0.000*	p=0.000*	p=0.000*	-

Table B.2.: Significance p-values for the differences in solving speed in the hard maze (see 4.2). P-values are only shown if the column-objective outperformed the row-objective both in terms of convergence and speed. The statistical test used is a two-tailed Mann-Whitneyr-U test.

Objectives	Convergence Rate	Solving Time	STD	N
FFA	0.9	363.61	189.32	20
FIT	0.75	316	276.88	20^{u}
frCUR	0.55	456.27	256.73	20
CUR	0.3	281.33	243.39	20

Table B.3.: Medium-maze-results for replacing the maximum selection mechanism for a frequency selection mechanism for fitness and curiosity (see 4.5). Results are averaged over two mazes. For all discretizations, both to identify frequencies and to differentiate paths for curiosity, the grid size was set to 10.

Objectives	Convergence Rate	Solving Time	STD	N
CUR	0.8	224.75	138.66	20
frCUR	0.65	237.23	168.89	20
FFA	0.45	487.33	289.03	20
FIT	0	0	0	20

Table B.4.: Hard-maze-results for replacing the maximum selection mechanism for a frequency selection mechanism for fitness and curiosity (see 4.5). Results are averaged over two mazes. For all discretizations, both to identify frequencies and to differentiate paths for curiosity, the grid size was set to 10.

Objectives	Convergence Rate	Solving Time	STD	N
RAR/VIAB	1	152.37	108.01	30
RAR	1	157.77	140.65	30
FIT	0.73	333.09	284.67	30
FIT/VIAB	0.65	255.85	224.78	20
CUR/VIAB	0.5	441.4	285.39	30
CUR	0.47	471.07	322	30

Table B.5.: Convergence rate and average generations until a solution was found for the experiment in Section 4.6.4 on the effect of viability on maze performance in the medium maze.

Objectives	Convergence Rate	Solving Time	STD	N
RAR/VIAB	1	100.33	58.57	30
RAR	1	106.87	48.54	30
CUR	0.53	324	262.05	30
CUR/VIAB	0.33	353.3	264.79	30
FIT	0	0	0	30
FIT/VIAB	0	0	0	20

Table B.6.: Convergence rate and average generations and until a solution was found for the experiment in Section 4.6.4 on the effect of viability on maze performance for the hard maze.

	RAR/VIAB	RAR	FIT	FIT/VIAB	CUR/VIAB
RAR/VIAB	-	-	-	-	-
RAR	p=0.668	-	-	-	-
FIT	p=0.008*	p=0.006*	-	-	-
FIT/VIAB	p=0.062	p=0.059	-	-	-
CUR/VIAB	p=0.002*	p=0.002*	p=0.315	p=0.107	-
CUR	p=0.004*	p=0.004*	p=0.224	p=0.085	p=0.879

Table B.7.: Significance values for the differences in speed for the experiment in Section 4.6.4 on the effect of viability on performance in the medium maze. Importantly, none of the differences of between an objective and its combination with viability is significant.

	RAR/VIAB	RAR	CUR	CUR/VIAB	FIT
RAR/VIAB	-	-	-	-	-
RAR	p=0.535	-	-	-	-
CUR	p=0.000*	p=0.001*	-	-	-
CUR/VIAB	p=0.000*	p=0.000*	p=0.833	-	-
FIT	p=0.000*	p=0.000*	p=0.000*	p=0.000*	-
FIT/VIAB	p=0.000*	p=0.000*	p=0.000*	p=0.000*	p=-1.000*

Table B.8.: Significance values for the differences in speed for the experiment in Section 4.6.4 on the effect of viability on performance in the hard maze. Importantly, none of the differences of between an objective and its combination with viability is significant.

	NOV	FFA	tRAR	FIT
NOV	-	-	-	-
FFA	p=0.970	-	-	-
tRAR	p=0.307	p=0.791	-	-
FIT	p=0.186	p=0.307	p=0.427	-
RAR	p=0.017*	p=0.104	p=0.473	p=0.970

Table B.9.: Significance values for the difference in maximum distance walked in the biped experiment in Section 4.7.3.

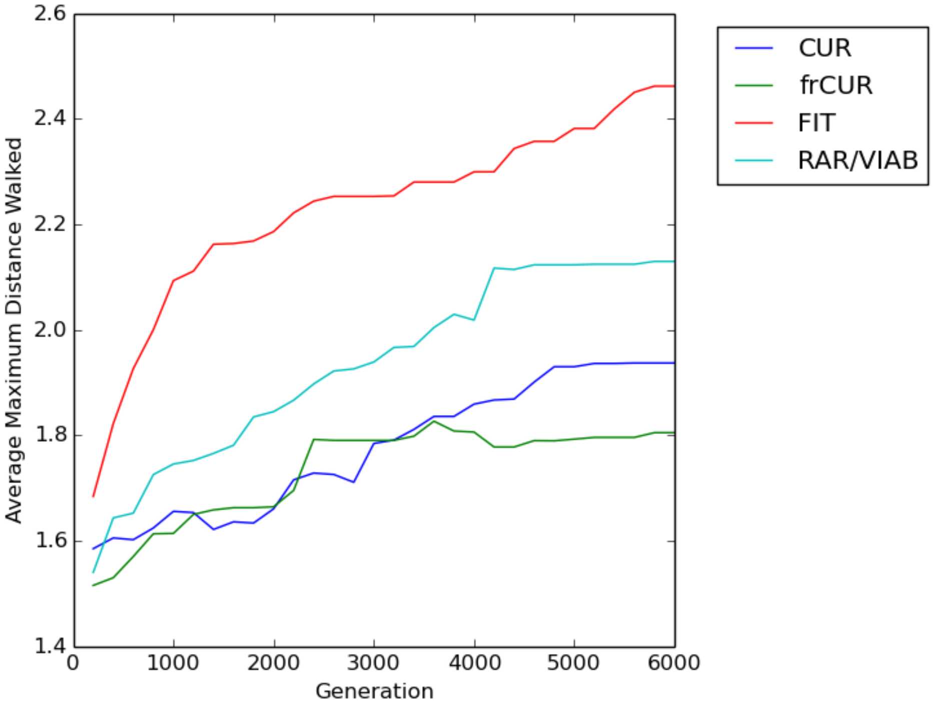


Figure B.1.: Average maximum distance walked for the different objectives for each generation (see 4.7.3). Results were averaged over 10 trials.