Diversity Preservation in Minimal Criterion Coevolution through Resource Limitation

In: *Proceedings of the Genetic and Evolutionary Computation Conference*, (GECCO 2020). New York, NY: ACM

Winner of the Best Paper Award in Complex Systems

Jonathan C. Brant and Kenneth O. Stanley
Dept. of Computer Science, University of Central Florida, Orlando, FL
{jbrant,kstanley}@cs.ucf.edu

ABSTRACT

Minimal Criterion Coevolution (MCC) is a recently-introduced algorithm that demonstrates how interactions between two populations, each subject to a simple reproductive constraint, can produce an open-ended search process. Unlike conventional quality diversity (QD) algorithms, which also promote divergence, MCC does not require an explicit characterization of behavior or a comparison of performance, thereby addressing bottlenecks introduced by an intrinsically-finite behavior descriptor and by an assessment of comparative quality. Genetic speciation, a common method of diversity preservation, maintains population diversity in MCC; however, it requires an unnatural explicit comparison of genetic similarity. In nature, organisms are implicitly segregated into niches that each have a carrying capacity dictated by the amount of available resources. To show that MCC can be simpler and more natural while still working effectively, this paper introduces a method of diversity preservation through resource limitation, thereby alleviating the need to formalize and compare genetic distance. Experimental results in a maze navigation domain demonstrate that resource limitation not only maintains higher population diversity in both the maze and agent populations, but also accelerates evolution by forcing individuals to explore new niches, thereby suggesting that resource limitation is an effective, simpler, and more natural alternative for diversity preservation in MCC.

ACM Reference Format:
Jonathan C. Brant and Kenneth O. Stanley. 2020. Diversity Preservation in Minimal Criterion Coevolution through Resource LimitationIn *Genetic and Evolutionary Computation Conference (GECCO '20), July 8–12, 2020, Cancún, Mexico.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3377930.3389809

1 INTRODUCTION

Natural evolution has long served as an inspiration for the field of evolutionary computation (EC) [9, 11, 12, 17]. For most of its history, however, simulated evolution has been primarily harnessed for producing solutions to specific problems [9], ignoring perhaps the most unique and profound attribute of its in-vivo counterpart: the perpetual generation of increasingly complex and diverse forms. Rather than converging to a defined optimization objective, nature branches out into multiple evolutionary trajectories, producing a rich ecosystem that bears little resemblance to its humble origins.

The creative properties of natural evolution that led to a complexity explosion are inspirational to the notion of open-endedness in artificial systems – a concept with a long history of investigation in the field of artificial life (alife) [1, 25, 27, 48], but which has nevertheless only recently been framed as a tool that can be leveraged for solving problems of practical interest [49]. Quality diversity (QD) algorithms [38] are one such example of divergent search methods that capture some flavor of evolution's affinity for functional diversity. Instead of converging on a global optimization target, QD algorithms explicitly seek out novel discoveries while ranking those that are behaviorally similar against an objective measure of performance. However, while conventional QD algorithms embody some of the attributes of an open-ended system, they require an explicit characterization of behavior (i.e. to maintain a record of points visited in the behavior space) to assess novelty. Consequently, as a wider breadth of the finite behavior space is sampled, pressure toward novelty diminishes, thus limiting the open-ended expression of most QD algorithms.

In contrast, a recently-introduced algorithm called *minimal criterion coevolution* (MCC) [4, 5] is predicated on the observation that nature requires no such behavioral formalism; rather, reproduction is the sole criterion for the continuation of one's lineage, and within that constraint, divergence is a byproduct of genetic drift. In MCC, two populations are coevolved, each subject to a *minimal criterion* (MC) of reproduction with respect to the other. MCC was initially benchmarked in a maze navigation domain where mazes are coevolved with maze navigating agents controlled by neural networks (NNs). To meet their MC, agents are required to solve at least one maze, while mazes meet their MC by being successfully navigated by at least one agent. Within this simple evaluation framework, MCC discovers diverse mazes that vary in both size and structure, with agents continually solving larger and increasingly non-trivial challenges. The idea in MCC of coevolving agents and environments has also informed alternative open-ended approaches like the recent POET algorithm [55].

While MCC has shown promising results that suggest its potential for open-ended innovation, its method of diversity preservation remains ad-hoc and without natural precedent. Like many other evolutionary algorithms (EAs), conventional [9, 12] or QD-based [22, 35, 38, 39], MCC employs *speciation* to segregate members of both populations into one of a predetermined number of species based on an explicit comparison of genetic similarity. Though speciation in nature is a well-established ecological phenomenon, its

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GECCO '20, July 8–12, 2020, Cancún, Mexico
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7128-5/20/07…$15.00
https://doi.org/10.1145/3377930.3389809

implementation in MCC is contrived and furthermore, determining how to meaningfully compare similarity imposes a priori structure on the system, which conflicts with the intention in the long run to produce novel and perhaps unanticipated artifacts.

Responding to this opportunity to simplify MCC while making it also more natural, this paper introduces an alternative method of diversity preservation through *resource limitation.* As with prior MCC experiments, the benchmark maze domain demonstrates coevolutionary complexification involving mazes and agents; however, rather than segregating the two populations by genetic similarity, a resource limit is imposed on mazes such that a given maze can only be used (i.e. successfully navigated) by a finite number of agents for satisfying any given agent's MC. That way, agents are forced to satisfy their MC through different mazes. The results of this simple constraint demonstrate that enforcing resource limitation boosts population diversity by producing mazes with highly variable solution paths, while also accelerating evolution compared to explicit speciation by exposing agents to a broader array of new and challenging environments.

Resource limitation ensures that evolution remains unencumbered by potentially biased, a priori assumptions by alleviating the need to explicitly characterize and compare solution structure, while also facilitating the application of MCC to new domains for which such characterizations may be non-trivial or computationally-expensive. In this way, it opens up MCC to a wider breadth of potential applications while also simplifying it and making it easier to implement in practice.

2 BACKGROUND

This section provides an overview of recent research in open-endedness, discusses conventional diversity-preservation methods in EC, and reviews the MCC algorithm.

2.1 Open-endedness

Evolution in nature has consistently captivated researchers in artificial life (alife) and evolutionary computation (EC) because of its unparalleled propensity for creativity, and its ability to produce increasingly complex organisms and adaptations that continue to diverge along many dimensions of variation [2, 3, 32, 40, 51]. This characteristic of evolution is broadly known as *open-endedness*, but the conditions that lead to an open-ended process are an active area of study that is hotly-debated within multiple scientific disciplines [27, 42, 48].

In alife, hypotheses regarding how individuals might interact are formulated and often tested in a simulation, or *alife worlds* [6, 33, 37, 40, 56]. Ultimately, the hope is to discover a set of conditions that lead to a complexity explosion reminiscent of that observed on earth [49]. Prevailing evolutionary theory posits that selective pressure induces a drive toward increasing complexity [31]; however, recent research has suggested that complexity may instead be an artifact of random genetic drift [26, 27, 32].

Soros and Stanley [47] recently introduced an alife world called Chromaria that relies on a strategy of genetic drift to test several hypothesized conditions for producing open-ended evolution. Chromaria has no fitness function or method of quantifying and comparing solution quality; instead, individuals are required to satisfy a simple, binary minimal criterion (MC) to reproduce and propagate their lineage. An important insight, however, is that forcing individuals to interact while satisfying their MC results in a continual flux of MC difficulty, leading to increasingly novel and complex discoveries.

Early hints of a more open search process in EC were first demonstrated by an algorithm called Viability Evolution (ViE) [28, 30]. ViE is a form of multi-objective search that encloses the space of possible solutions inside of *viability boundaries*, each of which correspond to a constraint that individuals must satisfy to reproduce. The boundaries are incrementally tightened, producing a set of solutions that satisfy the most stringent constraint values while permitting variation within the enclosed space. While ViE has no requirement for an objective function, it is still a convergent algorithm that prunes away population diversity instead of encouraging it.

Rather than converging toward a set of a priori optimization targets, a new class of *non-objective* methods induce evolutionary *divergence* by promoting behavioral novelty. For example, the novelty search algorithm [20, 21] ignores objective-oriented performance entirely; instead, individuals are rewarded according to their degree of behavioral novelty as compared to past discoveries, thereby encouraging diversity and exploration by pushing outward across the search space. This approach is particularly effective for deceptive domains, where following the gradient of fitness may lead to sub-optimal regions of the search space. Novelty search inspired a succession of new *quality diversity* (QD) algorithms [22, 35, 38, 39] that combine the search for novelty with objective performance, collecting many diverse artifacts, each as high-performing as possible with respect to a separate measure of quality.

2.2 Diversity Preservation in EC

Most evolutionary algorithms (EAs) model search as a *convergent* process that selects only the most fit individuals for reproduction, iteratively honing in on high-fitness areas of the search space [41]. As a population-based search method, however, EAs rely on maintaining a diverse sample of candidate solutions to avoid convergence to areas of the search space that are only locally optimal. Diversity preservation techniques are generally loosely based on niching in natural evolution, where organisms are segregated into distinct species and competition occurs primarily within each species' niche rather than at a population level [43, 54]. In EC, such methods are employed to manage the conflicting goals of exploration and exploitation, ensuring that multiple evolutionary lineages are maintained to avoid naively committing to a single evolutionary path without adequately vetting the alternatives [29].

Conventionally, similarity is assessed at the genotype-level [15, 29], though individuals may also be grouped based on age [18, 46] or fitness [19]. The Neuroevolution of Augmenting Topologies (NEAT) method [50], which evolves and complexifies NNs (and is modified to control maze-navigating agents in this paper), typically speciates genotypes by comparing shared evolutionary lineage between each topological component (nodes and connections). Speciation and other niching methods encourage diversity in the genotype space without regard for whether genetic disparity translates into meaningful phenotypic differences; however, recent QD research has found that many distinct genotypes may collapse into the same phenotypic expression, suggesting that augmenting genetic speciation with an explicit search for behavioral novelty may be more effective at preserving *functional* diversity in deceptive or ambitious domains [21]. Nature, however, does not explicitly measure genotypic or behavioral similarity. Instead, diversity is in part a byproduct of finite, yet essential resources [24, 43] – an insight that motivates the present work.

2.3 Minimal Criterion Coevolution

While QD algorithms capture some of the divergent aspects of natural evolution, their capacity for open-ended discovery is limited by the requirement to characterize and compare behavioral novelty. Defining all possible dimensions of behavioral variation a priori for a system meant to continue indefinitely is not practical, and assessing the comparative novelty of an ever-expanding evolutionary repertoire is computationally infeasible. Nature does not implement an explicit drive toward novelty, nor does it rank individuals against an optimization criterion. Fundamentally, natural evolution imposes only one constraint: *survive long enough to reproduce.*

Inspired by this perspective, Minimal Criterion Coevolution (MCC) offers a different take on divergent evolutionary search [4, 5] – one that is inspired by the simplicity of nature's binary, reproductive criterion for survival. MCC is a coevolutionary algorithm that evolves two populations in accordance with their respective minimal criterion (MC), which is a baseline constraint on functionality. Where conventional coevolutionary algorithms issue reward according to a competitive or cooperative objective, MCC evaluates individuals only against their MC, allowing diversity to flourish while avoiding trivial discoveries (which would fail the MC).

MCC adopts the population structure and selection process originally introduced in Chromaria [47], the alife world developed to examine evolutionary interactions subject only to the MC. Individuals who satisfy the MC are admitted to a population queue and selection from the queue is determined only by the order of insertion (i.e. there is no notion of explicit fitness beyond the MC). Removal from the queue is then age-based, thereby ensuring that each individual who satisfies their MC gets at least one chance to reproduce. This selection and removal method is free of fitness and novelty bias, which keeps many evolutionary paths open while avoiding expensive comparative assessments of behavioral similarity and facilitating parallel evaluation by leveraging the MC as an independent measure of performance.

While most EAs begin evolution with a randomly-initialized population, MCC requires that all individuals in the population satisfy a non-trivial MC, which necessitates a bootstrap process in which individuals are evolved with the MC as an explicit initial objective. Those who satisfy the MC seed their respective population queue. Prior MCC experiments have utilized non-objective algorithms, such as novelty search [21], for evolving seed genomes because of their propensity to produce a diverse initial population; however, in principle, any EA, fitness-based or otherwise, could be used.

When MCC was first introduced, two variants were evaluated: one utilizing genetic speciation and the other without any method of diversity preservation. Speciation was found to markedly increase population diversity by maintaining reproductive viability among several distinct lineages [4]. The approach to speciation in the original MCC is adapted to work with population queues: Species assignment is determined through k-means clustering, where pre-evolved individuals that seed the two population queues act as the initial cluster centroids. Genetic similarity is computed between each pair of population occupants to determine species membership, and individuals are assigned to the species with whom they share the greatest genetic similarity. Queue capacity is subdivided into equally-sized partitions such that each species i has a carrying capacity equivalent to $capacity(i) = n/s$, where n is the maximum population queue capacity and s is the number of species. Selection extracts a proportionate number of individuals from each species for reproduction in accordance with their queue insertion order. If adding a new individual to the queue causes a species to exceed its carrying capacity, the oldest member of the assigned species is removed.

Like other methods of diversity preservation, speciation in MCC is inspired by ecological niching, where each niche has a limited carrying capacity, thereby encouraging population divergence by founding new niches. In practice, however, speciation requires an experimenter to dictate several critical parameters, including the species count and distance metric, *a priori*. For a method like MCC aiming to take inspiration from natural evolution while introducing as little additional machinery as possible, the overhead of the speciation component is undesirable. To address this challenge, this paper introduces an alternative method for diversity preservation that exploits the mutually-dependent MC of two populations to instead induce a competition for limited resources, allowing diversity to vary more naturally as both populations coevolve in accordance with their respective MCs.

3 APPROACH: RESOURCE LIMITATION

In nature, species occupy niches whose carrying capacity is limited by environmental factors, including the availability of natural resources (i.e. consumable material that is required for the persistence of an organism, such as habitat and nutrients) and the frequency of predation [43, 54], both of which impose a form of local regulation on niche membership. A long-standing hypothesis in population ecology is that competition for limited resources has played a critical role in adaptive radiation and the formation of ecological communities [44, 52, 53]. In particular, as resources become increasingly scarce, organisms are forced to diverge and found new niches in which to continue their lineages. Those organisms capable of exploiting underutilized resources will be favored over those who cling to resource-depleted niches [45].

Using the Avida digital evolution platform [37], Cooper and Ofria [7] demonstrated that competition for limited resources can have a stabilizing affect on simulated community structures. Resource limitation was also shown to be effective in diversity preservation in EC [10, 13, 14], but it has primarily been leveraged in the context of objective-driven, convergent search processes. Recent work in EC, however, has shown that genetic drift (a process on which MCC heavily relies) can lead to the founding of new niches [8, 23], thereby creating novel and varied opportunities for individuals to persist. This effect is especially true for organisms who exhibit a high degree of *evolvability* because they are innately more adaptable to new niches, thereby promulgating lineages of similarly evolvable offspring.

Viewing speciation as a process that emerges from limited environmental resources and random drift across the genotype space is well-aligned with MCC's open-ended approach to search. Rather than computing a genetic distance between individuals and grouping the most similar into one of a predefined number of species, in this new version of MCC a limitation is placed on the number of times an individual in one population can "use" an individual from the other population to satisfy their MC. This approach is particularly intuitive when individuals in one population are framed as evaluation environments and those in the other population as organisms whose MC is to interact with an environment in some desirable manner.

This way, each member of the population on which a resource limit is imposed maintains a tally of the number of times it has been utilized by a member of the opposite population for satisfying their MC. When the usage tally reaches the predefined resource limit, it can no longer be used for MC satisfaction. Algorithm 1 formalizes the MCC selection, evaluation and removal process with the resource-limited population. Maintaining diversity through resource limitation negates the requirement for computing genetic distance, creating a dynamic, self-regulating system that exploits MCC's propensity for open-ended divergence to discover new and varied adaptations.

4 EXPERIMENT

Prior work demonstrated MCC through an evolving maze domain [4, 5], where maze-navigating agents are coevolved with maze environments whose size and structure varies throughout evolution. Maze navigation is often used as a test-bed for non-objective algorithms because solution path structure serves as a proxy for overall task complexity, while deceptive fitness gradients are easily visualized as trajectories that move in the direction of the goal, but ultimately lead to a dead-end [20, 21, 34, 36, 38]. However, while other non-objective algorithms may produce multiple novel solutions to a single maze, MCC coevolves agents and mazes in tandem, creating a potentially endless curriculum of new and varied mazes, each with effective adaptations.

Agents are controlled by NNs whose weights and topology are evolved using a modification of the Neuroevolution of Augmenting Topologies (NEAT) algorithm [50] (without speciation in the new MCC variant in this paper). NEAT starts with minimal architectures (few nodes and connections) and gradually adds structure to the extent that those additions result in improved task performance. NEAT has a long history in complex control tasks [50] and is particularly well-suited to open-ended domains where the addition of tunable parameters can be exploited to produce increasingly complex behaviors. The NN architecture for the maze domain is identical to that of Brant and Stanley [4, 5], with six rangefinder sensors that measure distance to line-of-sight obstructions and four pie slice sensors that activate when the goal is within the sensor's arc.

Mazes are represented as square grids with a fixed starting location in the upper-left corner and a fixed goal location in the lower-right corner. Agents begin a trial at the starting location, and their objective is to reach the target location within a limited simulation time, which is a function of maze size and solution path length (i.e. agents are allotted more time to navigate larger mazes with longer solution paths). The maze encoding is first briefly reviewed, after which resource limitation for preserving population diversity and encouraging coevolutionary divergence is explained.

Algorithm 1 MCC Evaluation Process with Resource Limitation

```
Require:
  batchSize - # of individuals to evaluate simultaneously
  numSeeds - # of seed genomes to evolve that satisfy the MC
  resourceLimit - max # of evaluations that count toward MC
  ▷ Evolve seed genomes that satisfy MC
  $randPop \leftarrow GenerateRandomPopulation()$
  $viablePop \leftarrow EvolveSeedGenomes(randPop, numSeeds)$
  loop
    ▷ Reproduce children and add parents back into queue
    $parents \leftarrow viablePop.Dequeue(batchSize)$
    $children \leftarrow Reproduce(parents)$
    $viablePop.Enqueue(parents)$
    for all $childinchildren$ do
      if $popType(child) = agent$ then
        for all $env \in environments$ do
          ▷ Only select environments below resource limit
          if $env.Usage < resourceLimit$ then
            $evalIndiv \leftarrow env$
            break
          end if
        end for
      else
        $evalIndiv \leftarrow next(agents)$
      end if
      $mcSatisfied \leftarrow EvaluateMC(child, evalIndiv)$
      if $mcSatisfied$ then
        $viablePop.Enqueue(child)$
      end if
    end for
    if $viablePop.Size > viablePop.Capacity$ then
      $numRemovals \leftarrow viablePop.Size - viablePop.Capacity$
      $RemoveOldest(viablePop, numRemovals)$
    end if
  end loop
```

4.1 Maze Encoding

The enhanced maze encoding of Brant and Stanley [5] is again the testbed in this paper. Maze genomes specify the overall maze size (i.e. length of the outer boundaries) and consist of genes that encode waypoints (referred to as "path" genes) and walls. Waypoints are coordinates that are linked by a set of rules, thus forming a solution path. Mutation operators probabilistically shift waypoints a single unit in one of four directions (up, down, left or right), while also controlling the rate at which new waypoints are added.

Wall genes dictate the position and orientation (i.e. horizontal or vertical) of partitions that bisect sub-spaces carved out by the solution path. Each wall gene also encodes a passage location, which specifies the location of an opening in the wall through which an agent can pass. Wall locations are relative, in the range $[0, 1]$, and are scaled to the dimensions of the space in which they are placed. Passage locations adopt the same relative range and are scaled to the length of the wall. Wall genes are subject to mutation operators that probabilistically add and remove walls, and shift wall and passage location. The maze size is modified by a separate mutation parameter that controls the rate at which the maze expands.

Maze genomes are decoded by first placing waypoints to form the solution path, then arranging walls within the adjoining sub-spaces. Walls iteratively bisect each sub-space according to a randomly assigned orientation. Each wall's relative position is scaled to the dimensions of the space in which it is placed, and the position of the wall's passage is scaled to the length of the wall. For example, if a wall with a horizontal orientation, a relative position of 0.5 and a passage position of 0.6 is placed within a square space of length 10, the wall will start at coordinate $(0, 5)$ and end at coordinate $(10, 5)$, and the passage will be placed at $(6, 5)$. Figure 1 visually depicts the effect of each mutation operation.

4.2 Resource Limitation

While a differing arrangement of walls may increase or decrease the level of domain deception, it may or may not necessitate modification of an agent policy for successful navigation. Conversely, perturbing a path gene directly modifies the solution path, which is more likely to require explicit adaptation. Path genes are the fundamental units of maze genomes, and were therefore used to assess maze similarity for speciation in the original MCC experiments. Specifically, maze similarity was computed as the Manhattan distance between every pair of path waypoints. If mazes had an unequal number of path genes, the last waypoint in the maze with fewer path genes was used for remaining distance calculations. In the agent population, the standard NEAT method of speciation was used [50] wherein similarity is determined by the number of connections with matching historical markings.

While deriving an intuitive distance metric is fairly straightforward for simple structures such as mazes, there may exist other domains for which quantifying distance is more difficult or less informative. For example, in a robot locomotion domain where MCC is used to coevolve morphology and control, the dimensions by which to characterize morphology distance are perhaps less intuitive, and could also vary substantially based on the physical substrate (e.g. rigid joints and materials vs. soft body structures with varying degrees of material compliance). More importantly, nature requires no such ad-hoc means of explicitly measuring organism distance to encourage diversity. Instead, evolutionary divergence is in part a byproduct of niche carrying capacities, or *resource limits*.

To emulate such limitation, each maze is assigned a fixed resource limit, and only successful navigations within that limit are counted toward satisfying an agent MC. For example, if a maze has a resource limit of five, and five agents have already solved the maze, then it has reached its resource limit and is no longer available for agent evaluation (because successful navigation would not count toward satisfying an agent's MC, and would thus be an inefficient use of computation). Note also that resource limitation is applied *asymmetrically* in that it is only imposed on the maze population; agents do not have an associated resource limit, so there is no restriction on the number of times they can be used to satisfy a maze MC. In practice, implementing resource limitation in only one direction is sufficient to promote diversity in both populations, because as individuals in one population maintain viability by using alternate, under-utilized individuals in the other, they naturally evolve a wider range of adaptations that enable continued divergence in the opposite direction as both populations interlock to satisfy a mutual MC. For example, in the maze domain, agents must evolve diverse navigation strategies to solve mazes that have not reached their resource limit, which, conversely, enables them to successfully navigate similarly diverse mazes, thus satisfying the MC of those mazes.

4.3 Experimental Parameters

The bootstrap process required by MCC is executed using the novelty search algorithm with parameters identical to those in Lehman and Stanley [21]. Agents are evaluated on ten simple, randomly-generated mazes of identical size, each with two path genes and two wall genes. Twenty agents are evolved such that each agent is able to solve at least one maze, while respecting the maze resource constraints that no more than five agents can solve a single maze. This resource limit was shown to maintain a healthy equilibrium and promote sustained evolutionary divergence in both populations during an initial parameter sweep. Mazes retain the resource usage incurred during initialization, and seed the maze population queue while the evolved agents seed the agent queue.

MCC experiments are executed in 20 runs of 2,000 batches, where 40 agents and 10 mazes are evaluated within each batch. Agents have a maximum velocity of 3 units per second and are allotted a simulation time of two times the maze solution path length. SharpNEAT version 3.0 [16] is the neuroevolution platform, which was extended to implement novelty search and MCC with speciation and resource limitation, and to support encoding and evolving maze genomes. Configuration parameters (e.g. NEAT and maze genome mutation rates) for both the resource-limited version of MCC and the original speciated version (the control in this paper) are identical to those used in Brant and Stanley [5], and the speciation configuration for the control is also identical, thereby isolating effects to the selected diversity preservation method. Source code for the experiments is available at https://bit.ly/33WsgKI.

5 RESULTS

Recall that the aim of these experiments is to investigate whether a simple resource limitation can maintain similar levels of population diversity as speciation, and thus serve as a much simpler and more natural method of diversity preservation. The results of both methods (i.e. resource limitation and speciation) are evaluated by comparing the diversity of mazes both discovered and solved during evolution. Diversity is calculated by measuring the Manhattan distance between the solution paths of every pair of mazes in the population, and dividing by the number of path segments to normalize by path length, thereby accounting for larger mazes that may have longer, but not necessarily more diverse, solutions paths. Formally, the diversity score of a maze is given by

$$\mathrm{div}(m) = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{1}{l} \sum_{j=1}^{l} \mathrm{dist}(m_j, \mu_{i,j}) \right), \tag{1}$$

where m represents the maze whose diversity is being measured, m_j is the j-th location in the maze solution path, $\mu_{i,j}$ corresponds to the same solution path index in the maze against which m is compared, l is the length of the solution path, n is the number

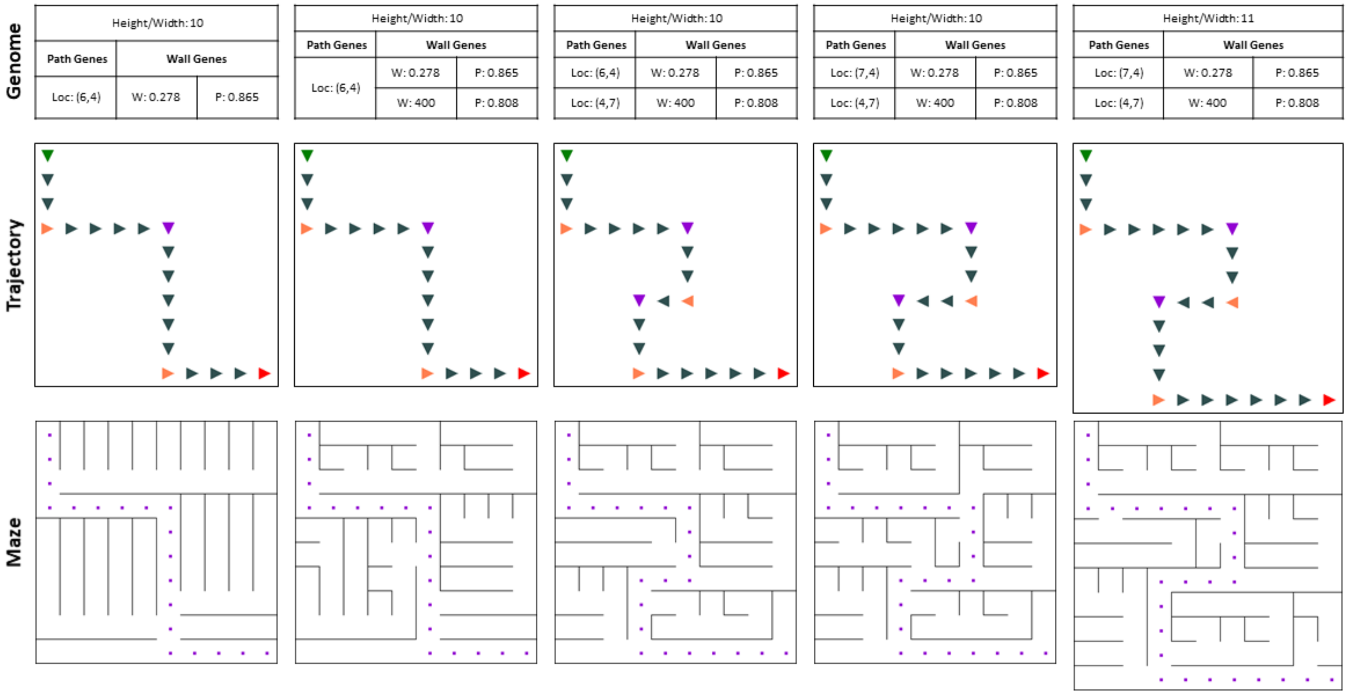

Figure 1: The maze evolution process. Maze genotypes (top), respective trajectories (middle), and respective phenotypes (bottom) are shown. The table above each maze shows the path (waypoint) and wall genes, while the visualization directly below depicts the resulting trajectory. In the maze phenotypes at bottom, dots are shown to depict the solution path. Each path gene specifies the coordinates of a waypoint, and each wall gene contains a wall position and passage position, both real numbers within the interval $[0, 1]$. The wall position (W) describes the relatives position of a wall within a sub-space carved out by the solution path, while the passage position (P) describes the relative position of a passage (opening) within that wall. The initial genome (first column) consists of one path gene and one wall gene; the single wall gene is repeated to fill out all sub-spaces. The second column depicts an *add wall* mutation, while the third column demonstrates an *add waypoint* mutation. In the fourth column, a *mutate waypoint* operation is carried out, shifting the first waypoint one unit to the right. Finally, the fifth column depicts an *expand maze* mutation, extending the outer boundaries down and to the right, and resulting in a corresponding extension of the solution path.

of mazes in the population, and *dist* represents the function that measures distance between two solution paths (in this case, the Manhattan distance). The figures in this section compare speciation and resource limitation across a range of metrics, depicting the significance of each measurement with a 95% confidence bound (shown as an error bar) on the means.

Figure 2 depicts the average distance between solution paths in mazes discovered by the speciation and resource limitation variants. Imposing a resource limit on mazes leads to significantly more diverse solution paths ($p < 0.001$; Welch's t-test) over the course of a run. Resource limitation also accelerates the rate at which larger mazes are discovered and solved. Figure 3 depicts a significantly more rapid maze expansion rate ($p < 0.001$ after batch 100; Welch's t-test) that culminates in mazes that are, on average, twice the size of mazes evolved by the speciated variant. Resource limitation also evolves a much broader range of maze sizes (figure 4). Note that the sharp decline in the number of mazes at the end of both experimental variants is caused by under-representation of mazes in that size range; runs were terminated before evolution fully explored the space of similarly-sized mazes.

Figure 5 depicts the rate at which new agent NN connections are incorporated successfully. While agent controllers in the resource limitation variant consistently add structure throughout evolution, they remain significantly more compact than those produced by the speciation variant ($p < 0.001$; Welch's t-test), despite learning advanced navigation policies for solving larger mazes. This trend indicates that resource limitation may be reducing bloat by finding effective pairings between NNs and maze niches whereas speciation produces arbitrary boundaries for both populations in isolation, and without regard to their interaction dynamics. Evidence for more advanced navigation policies also includes a significant increase ($p < 0.01$ between batches 250 and 600 and $p < 0.001$ thereafter; Welch's t-test) in the average number of deceptive junctures (figure 6) in the resource limitation variant, where agents at a juncture (i.e. a turn in the maze solution path) are forced to choose one of multiple possible routes with incomplete information of where the chosen route may lead. The second, third and fifth turns in figure 7a are examples of deceptive junctures.

Figure 7 showcases a sample of agent trajectories (from evolved NN controllers) through mazes evolved by a *single* run of MCC, each varying in size and solution path complexity. All runs of MCC with resource limitation discovered mazes larger than those produced with speciation, with the largest mazes being more than three times the size and with effective agent solutions.

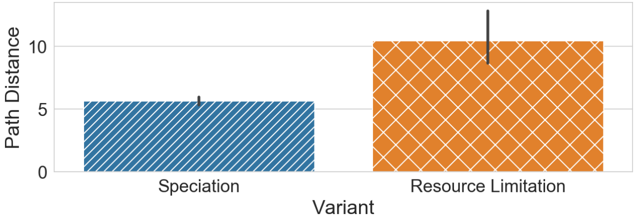

Figure 2: Solution Path Diversity by Experimental Variant. The average solution path distance between mazes for the speciation and resource limitation variants are compared. The resource limitation variant exhibits significantly higher solution path diversity.

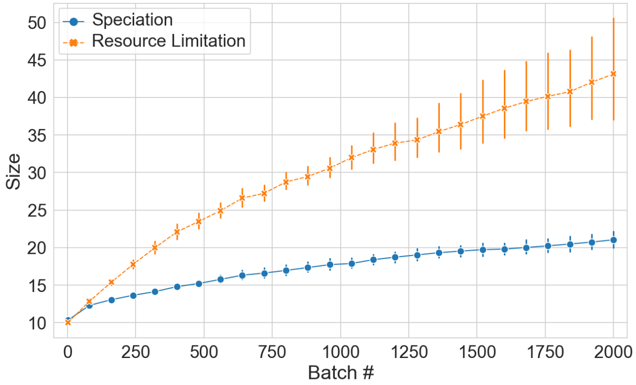

Figure 3: Maze Size Trend. The average maze size over evolution for the speciation and resource limitation variants are compared. The resource limitation variant discovers larger mazes, with an approximately linear rate of maze expansion throughout evolution.

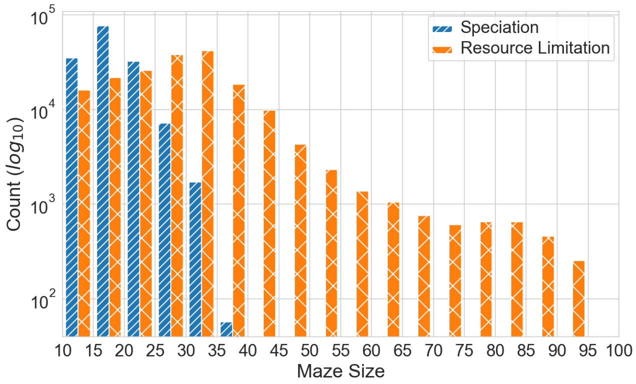

Figure 4: Maze Size Distribution. The count of mazes discovered over all runs for the speciation and resource limitation variants are compared. Each column pair depicts a bin of 5 consecutive maze sizes (e.g. the first contains sizes 10–14). Also, the y-axis is log-scaled so that the maze counts at the tail-end of both variants are visible. The resource limitation variant evolved a much wider range of maze sizes, some more than twice the size of the largest mazes discovered using speciation.

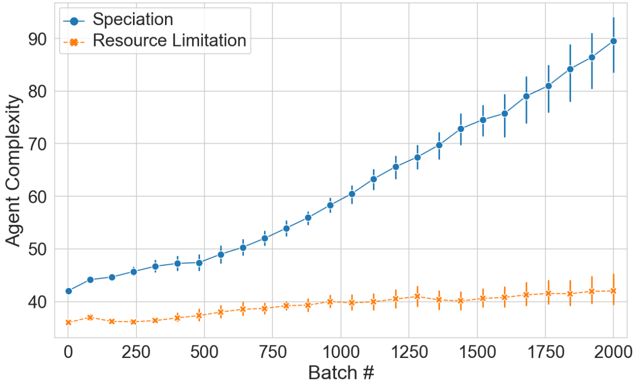

Figure 5: Agent NN Size Trend. The average agent NN connections over evolution for the speciation and resource limitation variants are compared. The resource limitation variant continues to add structure, but maintains significantly more compact NNs throughout evolution while solving more complex mazes than the speciation variant. This result suggests that resource limitation may reduce bloat by finding effective pairings between maze and agent.

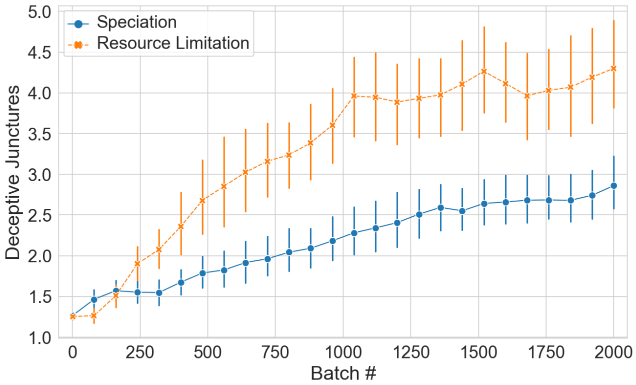

Figure 6: Maze Deceptive Junctures Trend. The average number of junctures (turns) in maze solution paths with multiple possible routes for the speciation and resource limitation variants are compared. A higher number of such junctures requires a more advanced policy because an agent must decide along which route to continue its trajectory with incomplete information of where that route may lead. The resource limitation variant discovers solution paths that are more deceptive, along with agent NN controllers that overcome such deception.

6 DISCUSSION

By restricting the number of times a single maze can be used for satisfying an agent MC, agents are forced to sample a wider variety of mazes that each vary in size and complexity. Those who are more evolvable are able to exploit underutilized resources, founding new niches and producing offspring who exhibit a similar degree of adaptability. This exploration across niche space creates a diverse curriculum of challenges that facilitate novel and complex agent adaptations, even without directly enforcing diversity through conventional methods such as speciation.

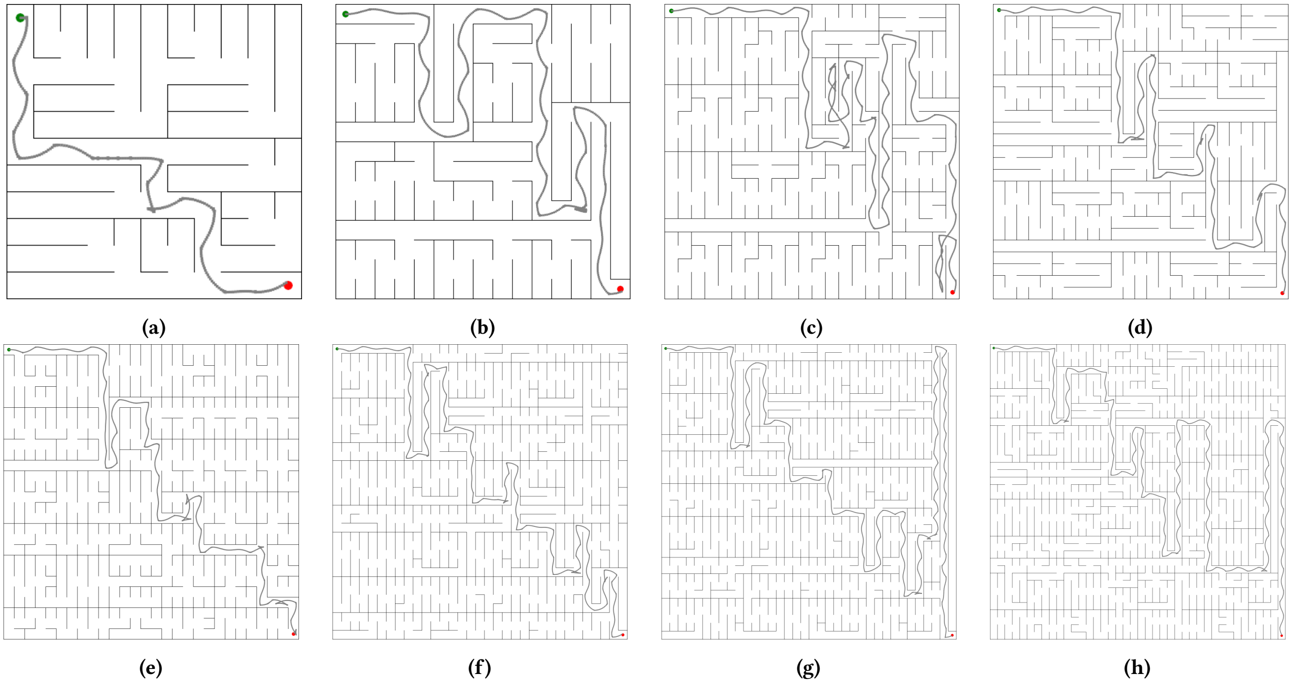

Figure 7: Mazes and agent trajectories discovered within a single run of MCC with resource limitation. A sample of eight solution trajectories through different mazes evolved within a single run of MCC are shown. Maze (a) is 11×11 (slightly larger than the 10×10 bootstrap mazes) with 2 waypoints and 2 walls (i.e. wall genes), maze (b) is 15×15 with 4 waypoints and 6 walls, and maze (c) is 22×22 with 6 waypoints and 8 walls. Maze (d) is the same size as the largest maze shown in Brant and Stanley [5], at 25×25 with 7 waypoints and 8 walls. Mazes (e) through (h) are larger and have more complex solution paths than any previously-demonstrated results in the MCC maze domain. Maze (e) is 28×28 with 7 waypoints and 10 walls, maze (f) is 33×33 with 10 waypoints and 13 walls, and maze (g) is 36×36 with 11 waypoints and 13 walls. The largest maze (h) is 40×40 with 10 waypoints and 22 walls.

Resource limitation discovers complex mazes with more diverse solution paths than speciation, while also evolving agents capable of rapidly adapting their policies to new challenges. A notable side-effect of resource limitation is an acceleration of the evolutionary process: larger mazes are discovered while NN controller size remains modest, yielding compact encodings that reduce NN bloat while encoding more advanced control policies. The results suggest that resource limitation is a suitable, even superior, replacement for speciation. In addition to offering a more natural form of diversity preservation, resource limitation also alleviates the need for ad-hoc decisions about how to meaningfully measure distance between encodings, and the computational overhead of exhaustively comparing similarity, facilitating the application of MCC to domains for which such a measurement would be uninformative, non-trivial or computationally expensive to execute. As MCC is applied to complex tasks in practical, real-world domains, it stands to benefit from the simplicity and efficiency afforded by resource limitation as the primary method of diversity preservation.

It is also notable that the resultant algorithm is among the simplest in EC. There is no measure of fitness apart from the MC, no ranking, no proportional selection, and no measure of diversity or explicit speciation. And yet it is still producing an open-ended phenomenon of increasing complexity and diversity rarely observed in any other algorithm, hinting that the fundamental ingredients necessary for open-ended evolution to flourish may be surprisingly minimal.

7 CONCLUSIONS

Prior implementations of MCC utilized speciation, a common form of diversity preservation in EC, to encourage diversity in both populations. Such artificial speciation requires an explicit comparison of distance between individuals in a population – a process that lacks natural precedent, and one that is ad hoc for the purposes of an algorithm inspired by the simplicity of open-ended evolution in nature. This paper proposed a more organic method of diversity preservation through resource limitation. Rather than binning individuals into species, a limit is placed on the number of times an individual from one population can be used for satisfying the MC of individuals from the other population. This method was evaluated in a maze navigation domain where only a limited number of successful trials counted toward satisfying an agent MC. Imposing a resource limit on mazes drove evolution to discover a wider range of maze sizes with more complex and diverse solution paths, each with effective solutions in the agent population. These results suggest that resource limitation is a viable replacement for speciation, thereby eliminating one of the few remaining ad-hoc implementation decisions in MCC and opening up the space of its future applications.

ACKNOWLEDGMENTS

Special thanks to Emily Dolson and Charles Ofria for raising questions about resource limitation and its potential relevance to MCC.

REFERENCES

[1] Christoph Adami, Charles Ofria, and Travis C. Collier. 2000. Evolution of Biological Complexity. *Proceedings of the National Academy of Sciences* 97 (2000), 4463–4468.

[2] Mark A. Bedau. 1998. Four Puzzles About Life. *Artificial Life* 4, 2 (April 1998), 125–140. https://doi.org/10.1162/106454698568486

[3] Margaret A Boden. 1999. The philosophy of artificial life. *Minds Mach.* 9, 1 (Feb 1999), 139–143. https://doi.org/10.1023/A:1008373528631

[4] Jonathan C Brant and Kenneth O Stanley. 2017. Minimal criterion coevolution: a new approach to open-ended search. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 67–74.

[5] Jonathan C Brant and Kenneth O Stanley. 2019. Benchmarking open-endedness in minimal criterion coevolution. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 72–80.

[6] Alastair Daniel Channon and Robert I Damper. 2000. Towards the evolutionary emergence of increasingly complex advantageous behaviours. *International Journal of Systems Science* 31, 7 (2000), 843–860.

[7] Tim F. Cooper and Charles Ofria. 2003. Evolution of Stable Ecosystems in Populations of Digital Organisms. In *Proceedings of the Eighth International Conference on Artificial Life (ICAL 2003)*. MIT Press, Cambridge, MA, USA, 227–232. http://dl.acm.org/citation.cfm?id=860295.860334

[8] Richard Dawkins. 2003. The evolution of evolvability. *On growth, form and computers* (2003), 239–255.

[9] Kenneth A De Jong. 2006. *Evolutionary computation: a unified approach*. MIT press.

[10] Emily Dolson, Wolfgang Banzhaf, and Charles Ofria. 2018. Applying Ecological Principles to Genetic Programming. In *Genetic Programming Theory and Practice XV*, Wolfgang Banzhaf, Randal S. Olson, William Tozier, and Rick Riolo (Eds.). Springer International Publishing, Cham, 73–88.

[11] Dario Floreano and Claudio Mattiussi. 2008. *Bio-inspired artificial intelligence: theories, methods, and technologies*. MIT press.

[12] David B. Fogel. 2006. *Evolutionary computation: toward a new philosophy of machine intelligence, third edition*. Wiley-IEEE Press.

[13] Sherri Goings, Heather Goldsby, Betty H.C. Cheng, and Charles Ofria. 2012. An ecology-based evolutionary algorithm to evolve solutions to complex problems. In *Artificial Life Conference Proceedings 2012*. MIT Press, 171–177. https://doi.org/10.7551/978-0-262-31050-5-ch024

[14] S. Goings and C. Ofria. 2009. Ecological approaches to diversity maintenance in evolutionary algorithms. In *2009 IEEE Symposium on Artificial Life*. 124–130. https://doi.org/10.1109/ALIFE.2009.4937703

[15] David E. Goldberg and J. Richardson. [n.d.]. Genetic Algorithms with Sharing for Multimodal Function Optimization. 148–154.

[16] Colin Green. 2016. SharpNEAT v3. *URL: http://sharpneat.sourceforge.net* (2016).

[17] John H. Holland. 1975. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. University of Michigan Press, Ann Arbor, MI.

[18] Gregory S. Hornby. 2006. 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 (GECCO '06)*. ACM, New York, NY, USA, 815–822. https://doi.org/10.1145/1143997.1144142

[19] M. Hutter and S. Legg. 2006. Fitness Uniform Optimization. *IEEE Transactions on Evolutionary Computation* 10, 5 (Oct. 2006), 568–589. https://doi.org/10.1109/TEVC.2005.863127

[20] Joel Lehman and Kenneth O Stanley. 2010. Revising the evolutionary computation abstraction: minimal criteria novelty search. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2010)*. ACM, 103–110.

[21] Joel Lehman and Kenneth O Stanley. 2011. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation* 19, 2 (2011), 189–223.

[22] Joel Lehman and Kenneth O Stanley. 2011. Evolving a diversity of virtual creatures through novelty search and local competition. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. ACM, 211–218.

[23] Joel Lehman and Kenneth O. Stanley. 2013. Evolvability is Inevitable: Increasing Evolvability Without the Pressure to Adapt. *PLoS ONE* 8, 4 (2013), e62186.

[24] M. A. Leibold. 1995. The niche concept revisited: Mechanistic models and community context. *Ecology* 76, 5 (1 1995), 1371–1382. https://doi.org/10.2307/1938141

[25] Richard E Lenski, Charles Ofria, Robert T Pennock, and Christoph Adami. 2003. The evolutionary origin of complex features. *Nature* 423, 6936 (2003), 139–144.

[26] Michael Lynch. 2007. The Evolution of Genetic Networks by Non-Adaptive Processes. *Nature reviews. Genetics* 8 (11 2007), 803–13. https://doi.org/10.1038/nrg2192

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

[28] Andrea Maesani, Pradeep Ruben Fernando, and Dario Floreano. 2014. Artificial evolution by viability rather than competition. *PloS one* 9, 1 (2014), e86831.

[29] Samir W. Mahfoud. 1995. *Niching Methods for Genetic Algorithms*. Ph.D. Dissertation. University of Illinois at Urbana-Champaign, Urbana, IL. http://citeseer.nj.nec.com/mahfoud95niching.html

[30] Claudio Mattiussi and Dario Floreano. 2003. *Viability Evolution: Elimination and Extinction in Evolutionary Computation*. Technical Report. EPFL.

[31] Daniel Mcshea. 1991. Complexity and evolution: What everybody knows. *Biology & Philosophy* 6 (07 1991), 303–324. https://doi.org/10.1007/BF00132234

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

[33] Thomas Miconi. 2008. Evosphere: evolutionary dynamics in a population of fighting virtual creatures. In *IEEE Congress on Evolutionary Computation*. IEEE, 3066–3073.

[34] Jean-Baptiste Mouret. 2011. Novelty-based multiobjectivization. In *New Horizons in Evolutionary Robotics*. Springer, 139–154.

[35] Jean-Baptiste Mouret and Jeff Clune. 2015. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909* (2015).

[36] Jean-Baptiste Mouret and Stéphane Doncieux. 2012. Encouraging behavioral diversity in evolutionary robotics: An empirical study. *Evolutionary computation* 20, 1 (2012), 91–133.

[37] Charles Ofria and Claus O Wilke. 2004. Avida: A software platform for research in computational evolutionary biology. *Artificial life* 10, 2 (2004), 191–229.

[38] Justin K Pugh, Lisa B Soros, and Kenneth O Stanley. 2016. Quality diversity: A new frontier for evolutionary computation. *Frontiers in Robotics and AI* 3 (2016), 40.

[39] Justin K Pugh, Lisa B Soros, Paul A Szerlip, and Kenneth O Stanley. 2015. Confronting the challenge of quality diversity. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2015)*. ACM, 967–974.

[40] Thomas S. Ray. 1991. An Approach to the Synthesis of Life. *Artificial Life II* (1991), 371–408. https://doi.org/10.1016/S0040-4039(01)97473-8

[41] G. Rudolph. 1996. Convergence of evolutionary algorithms in general search spaces. In *Proceedings of IEEE International Conference on Evolutionary Computation*. 50–54. https://doi.org/10.1109/ICEC.1996.542332

[42] Kepa Ruiz-Mirazo, Juli Peretó, and Alvaro Moreno. 2004. A universal definition of life: autonomy and open-ended evolution. *Origins of Life and Evolution of the Biosphere* 34, 3 (2004), 323–346.

[43] Howard D Rundle and Patrik Nosil. 2005. Ecological speciation. *Ecology letters* 8, 3 (2005), 336–352.

[44] Dolph Schluter. 1996. Ecological Causes of Adaptive Radiation. *The American Naturalist* 148 (11 1996), S40–S64. https://doi.org/10.1086/285901

[45] Dolph Schluter. 2001. Ecology and the origin of species. *Trends in Ecology & Evolution* 16 (08 2001), 372–380. https://doi.org/10.1016/S0169-5347(01)02198-X

[46] Michael D. Schmidt and Hod Lipson. 2010. Age-fitness Pareto Optimization. In *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation (GECCO '10)*. ACM, New York, NY, USA, 543–544. https://doi.org/10.1145/1830483.1830584

[47] L.B. Soros and Kenneth O Stanley. 2014. Identifying Necessary Conditions for Open-Ended Evolution through the Artificial Life World of Chromaria. In *ALIFE 14: The Fourteenth International Conference on the Synthesis and Simulation of Living Systems*. 793–800.

[48] Russell K Standish. 2003. Open-ended artificial evolution. *International Journal of Computational Intelligence and Applications* 3, 02 (2003), 167–175.

[49] Kenneth O. Stanley, Joel Lehman, and L. B. Soros. 2017. Open-endedness: The last grand challenge you've never heard of. https://www.oreilly.com/ideas/open-endedness-the-last-grand-challenge-youve-never-heard-of

[50] Kenneth O Stanley and Risto Miikkulainen. 2002. Evolving neural networks through augmenting topologies. *Evolutionary computation* 10, 2 (2002), 99–127.

[51] Charles Taylor and David Jefferson. 1993. Artificial life as a tool for biological inquiry. *Artificial Life* 1, 1_2 (1993), 1–13.

[52] David Tilman. 1982. Resource competition and community structure. *Monographs in population biology* 17 (1982), 1–296.

[53] David Tilman. 2000. Causes, consequences and ethics of biodiversity. *Nature* 405 (06 2000), 208–11. https://doi.org/10.1038/35012217

[54] Sara Via. 2009. Natural selection in action during speciation. *Proceedings of the National Academy of Sciences* 106, Supplement 1 (2009), 9939–9946.

[55] Rui Wang, Joel Lehman, Jeff Clune, and Kenneth O. Stanley. 2019. POET: Open-Ended Coevolution of Environments and Their Optimized Solutions. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '19)*. Association for Computing Machinery, New York, NY, USA, 142–151. https://doi.org/10.1145/3321707.3321799

[56] Larry Yaeger. 1993. PolyWorld: Life in a new context. *Artificial Life III* (1993), 263–298.