

000  
001  
002  
003  
004  
005  
006  
007  
008  
009  
010  
011  
012  
013  
014  
015  
016  
017  
018  
019  
020  
021  
022  
023  

# LIFELONG CONTROL THROUGH NEUROEVOLUTION

**Anonymous authors**

Paper under double-blind review

**ABSTRACT**

Reinforcement learning (RL) under continual environmental changes has remained a central challenge for decades. Novel designs of loss functions, training procedures and neural network architectures have not yet managed to alleviate the main mode of failure in lifelong learning: loss of plasticity. Here, we turn to a very different family of optimisers: neuroevolution (NE). Through an extensive evaluation on diverse lifelong control tasks, we see that NE exhibit a remarkable ability to adapt where RL fails catastrophically. We observe that, in the present of environmental shifts, NE naturally increases its diversity of solutions, evolving the ability to rapidly discover well-performing specialist individuals. We propose that NE can be a promising approach towards tackling the need for lifelong adaptation and that future work in both RL and NE should focus on the benefit of diversity in population-based approaches.

**1 INTRODUCTION**

Reinforcement learning (RL) has produced impressive feats in recent years, offering ways to advance robotic agents (Silver et al., 2016b), improve the capabilities of Large Language Models (Ouyang et al., 2022) and perform on-par with humans in complex decision-making problems (Silver et al., 2016a). It is not surprising that RL, the *de facto* formalism for learning through interaction with an environment, is driving the increasing adoption of Artificial Intelligence (AI) in the real world. However, today’s RL algorithms require immense training experience, extensive hyper-parameter tuning and, nevertheless, remain brittle to unexpected variations in their environment (Kirk et al., 2023; Pan et al., 2025; Kudithipudi et al., 2022). Continuing on this path requires moving out of our current paradigm of close human supervision and towards AI that can autonomously learn in shifting, *open-ended* worlds (Hughes et al., 2024; Clune, 2020). A major mode of failure for RL in such settings is a *loss of plasticity* (Klein et al., 2024; Sokar et al., 2023; Muppudi et al., 2024). Plasticity lies at the other end of stability and, since a large focus of past progress, RL has been on improving stability (Schulman et al., 2017b; Kirkpatrick et al., 2017), RL’s best-performing algorithm often exhibits a remarkable tendency to ignore shifts in their environment (Klein et al., 2024). The recent resurgence of RL has, however, triggered an interest in bringing plasticity back. A natural approach is the development of techniques that re-trigger learning (Sokar et al., 2023). However, such approaches often require extensive tuning, defeating their original motivation (Muppudi et al., 2024; Klein et al., 2024). Lifelong RL is today in need of low-cost, hyperparameter-free solutions for battling loss of plasticity. Here we propose to look for such a solution in a fundamentally different family of optimisers: neuro-evolution.

Evolutionary algorithms are a long-standing and diverse family of black-box optimization methods that, when applied for the optimization of artificial neural networks (ANNs), are referred to as neuroevolution (Risi et al., 2025; Stanley et al., 2019). A core component of any evolutionary algorithm is the presence of a *population*. The field originally drew its inspiration from biological evolution, repurposing the processes of mutation, selection, and reproduction for black-box optimization (Koza). Since then, the field has grown into an engineering-focused discipline with a pluralism of methods that leverage the population in diverse ways Risi et al. (2025). NE methods have recently been shown to perform competitively with RL in control tasks, with previous works noting that their population-based nature brings several advantages: scalability through parallelization, increased exploration that helps avoid local optima thanks to population diversity, and smaller sensitivity to hyperparameters (Such et al., 2018; Salimans et al., 2017; Chalumeau et al., 2023). In this work, we propose that there is another, yet unexamined, benefit of NE in control tasks: avoiding loss of plasticity in the face of environmental shifts.

054 Environmental change is both a setback and a driver for evolution. In a constant environment and  
 055 in the absence of an explicit mechanism for preserving diversity, a population converges to minimal  
 056 diversity imposed by its mutation rate (Giraud et al., 2001). When a shift occurs, it causes increased  
 057 fitness variation in the population, retriggering competition amongst the population and, thus, ex-  
 058 ploration. If the shift is too large or the population is not sufficiently diverse, the population may  
 059 experience a mass extinction (Nisioti & Moulin-Frier, 2022). If, however, some individuals survive,  
 060 this increased diversity can act as a buffer from future environmental changes. Thus, the popula-  
 061 tion becomes increasingly better at dealing with variation. The form of environmental change plays  
 062 an important role in this with some types of variation favoring adaptability (Nisioti & Moulin-Frier,  
 063 2022; Grove, 2014). The idea is central in the field of artificial open-endedness (Soros, 2017), where  
 064 evolving environments alongside agents trigger an automated curriculum driving continual change  
 065 (Clune, 2020; Wang et al., 2019). Works in this community have, in particular, shown that evolution  
 066 can handle and often benefits from environmental variability (Lehman & Miikkulainen, 2015).  
 067

068 Despite its relevance, to the best of our knowledge, no previous study has attempted to benchmark  
 069 evolution’s abilities in lifelong learning against those of reinforcement learning. Here, we offer  
 070 such a study. We consider a collection of tasks that pose a wide diversity of challenges, such as  
 071 sparse exploration and control of large ANNs, including feedforward, convolutional networks and a  
 072 Transformer-based architecture. We examine two distinct NE approaches: genetic algorithms (GA)  
 073 and evolution strategies (ES). We benchmark them against PPO (Schulman et al., 2017a), a state-of-  
 074 the-art RL algorithm as well as a recent variant explicitly designed for such lifelong settings. Our  
 075 study shows that NE exhibits an impressive ability at learning in the phase of environmental shifts,  
 076 surpassing the state-of-the-art in lifelong control.

077 To understand how NE achieves this, we analyze the population dynamics in the presence of envi-  
 078 ronmental shifts. In particular, we focus on the diversity within the population. We observe that,  
 079 under environmental shifts, populations evolve higher diversity compared to those in a constant en-  
 080 vironment. In a long evolutionary run, we see that, after prolonged environmental variability, the  
 081 population experiences a phase transition: a gradual increase in diversity leads the population to-  
 082 wards an abrupt shift to the optimal solution. This ability is contingent on the size of the population,  
 083 with a small population instead experiencing a collapse. Isolating the highest performing individuals  
 084 in a given shift reveals they differ significantly from each other: evolution adapts by finding special-  
 085 ist agents rather than a single generalist agent. ES differs significantly in how it deals with variation  
 086 in comparison to the GA: diversity remains low, and the population progresses slowly. Overall, the  
 087 GA outperforms ES, except when faced with sparse reward problems.

088 We provide code for reproducing our study at an anonymous github repo.

## 089 2 BACKGROUND AND RELATED WORKS

090 The problem of *plasticity loss* in artificial neural networks (ANNs) has been recognized since the  
 091 1980s, when catastrophic forgetting was identified as a key weakness of gradient-based learning  
 092 compared to symbolic or non-connectionist approaches (McCloskey & Cohen, 1989). In deep rein-  
 093 forcement learning (RL), where ANNs serve as policy approximators, this issue is magnified by the  
 094 non-stationarity of the environment. Algorithms such as PPO introduced stabilizing mechanisms  
 095 like clipped objectives and trust regions (Schulman et al., 2017a), which improved reliability but  
 096 further reduced adaptability. Continual RL methods have attempted to restore plasticity through  
 097 techniques like dynamic regularization or masked networks (Muppidi et al., 2024), but they often  
 098 require privileged information about when shifts occur and remain brittle under rapid change. Even  
 099 large-scale pre-trained models such as LLMs exhibit similar fragility when deployed out of distri-  
 100 bution (Kirk et al., 2023), suggesting that the stability–plasticity dilemma persists across domains.

101 Population-based methods, such as neuroevolution (NE), provide a fundamentally different way to  
 102 address this challenge. Instead of optimizing a single solution, they maintain a diverse set of can-  
 103 didates that evolve over time. Diversity allows populations to explore multiple adaptive pathways,  
 104 recover from dead ends, and naturally specialize after environmental shifts (Salimans et al., 2017;  
 105 Chalumeau et al., 2023). This makes NE well-suited for lifelong learning, where continuous adap-  
 106 tation is essential. Related approaches in multi-task learning, such as mixtures of experts or modular  
 107 networks, also leverage specialization, but they rely on a fixed set of experts and explicit coordina-  
 108 tion mechanisms, making them less open-ended than evolutionary search.

108 In this work, we compare two representative evolutionary algorithms. The *genetic algorithm* (GA)  
 109 explicitly maintains a population, selecting top-performing individuals and generating new candidates  
 110 through mutation while preserving an elite subset for stability (Such et al., 2018). *OpenES*  
 111 (Salimans et al., 2017) provides a scalable alternative by maintaining only the mean and sampling  
 112 perturbations, sacrificing rich modeling for efficiency.  
 113

### 114 3 EMPIRICAL STUDY

116 How does NE behave in control tasks with shifting dynamics? Does it exhibit similar limitations to  
 117 RL, in particular when it comes to loss of plasticity? To answer these questions, we here evaluate  
 118 a variety of NE and RL algorithms in lifelong learning tasks with the aim of observing and under-  
 119 standing differences in their ability to handle environmental shifts. We focus on two distinct NE  
 120 approaches: a genetic algorithm (GA) (Such et al., 2018) and an evolution strategies (ES) (Salimans  
 121 et al., 2017). We benchmark these methods against PPO, a state-of-the-art RL algorithm, and Trac-  
 122 PPO, a variant that adds adaptive regularization to PPO to address the loss of plasticity (Muppidi  
 123 et al., 2024)<sup>1</sup>.

124 To ensure a fair comparison, we employ the same budget of experience for the different approaches.  
 125 We have matched the sample complexity of the methods by considering a fixed number of episodes  
 126 for NE in a certain environment and multiplying this number by the population size and episode  
 127 length to determine the number of steps for PPO. While it is possible to match complexity through  
 128 other metrics, such as execution time, we believe that, from the perspective of lifelong learning in  
 129 the real world, environment steps are what matters. All conditions have been run for 10 trials, and  
 130 we provide mean and variance estimates.

131 **Note on hyperparameter tuning** : We have tuned all methods separately for each family of  
 132 tasks (but not for each task; we picked a random task within a family) to ensure that the methods  
 133 work well for the classical version of all tasks. In some cases, we have employed well-performing  
 134 hyperparameters suggested in previous works. We have not performed exhaustive tuning for NE:  
 135 we have employed the default hyperparameters provided by evosax (Lange, 2022a) and manually  
 136 searched for well-performing values when needed. While we expect benefits in some conditions by  
 137 further tuning, we believe that it should not play a central part in our study. First, a major reason for  
 138 the attractiveness of NE approaches is their remarkable robustness to hyperparameters compared to  
 139 other deep learning algorithms (Chalumeau et al., 2023; Such et al., 2018). Second, when comparing  
 140 NE approaches, our focus is not on their performance but on their distinct dynamics, which, due to  
 141 the aforementioned robustness, do not depend on the hyper-parameterization except for edge cases.

142 We first, describe the tasks we have considered in Section 3.1. We, then, take an overall look at  
 143 performance differences. across all tasks and methods in Section 3.2. In Section 3.3, we dive into  
 144 an analysis of the behavior of NE.

145 Our study is accompanied by appendices that provide implementation details and hyperparameters  
 146 employed, and additional empirical results that we refer to throughout the rest of the paper.  
 147

#### 148 3.1 TASKS

150 We consider three task families for lifelong learning, which we explain next.  
 151

152 **Classic control** We use the tasks Cartpole, Acrobot, and MountainCar, implemented in the gym-  
 153 max library (Lange, 2022b). These are test-beds with discrete actions and continuous observations  
 154 that pose simple control challenges and can be solved by small feedforward networks. Cartpole is an  
 155 easy balancing task with immediate and dense rewards. Acrobot and MountainCar are more chal-  
 156 lenging sparse reward environments, with the latter being significantly easier as good solutions can  
 157 be reached through random actions.

158 For each of the three tasks, we employ a lifelong-learning variation originally introduced in Muppidi  
 159 et al. (2024). Every 200 generations, we sample a vector from a normal distribution with standard  
 160 deviation 1.0 and add it element-wise to the observations (i.e., the vector remains constant for 200  
 161

<sup>1</sup>We currently provide the performance of Trac-PPO only in the simple control environments. We have a

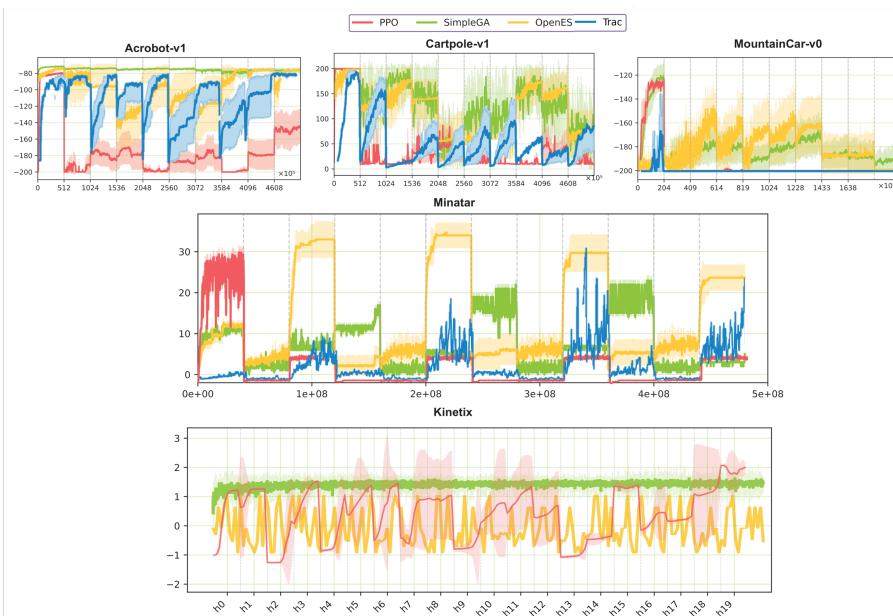


Figure 1: **Overall comparison across tasks:** we benchmark two NE approaches (SimpleGA and OpenES) against two RL variants (PPO and Trac-PPO, a variant designed specifically for lifelong learning). (Top) Simple control tasks (Middle) Minatar tasks: we consider the order Breakout, Asterix, Space-Invaders, going through four phases. (Bottom) Kinetix environments of medium difficult.

generations). This variation was found to be more challenging for RL compared to other approaches that vary the dynamics of these tasks. It can be seen as a model of distribution shifts in the sensory perception of the agent.

**Minatar** From the popular Atari benchmark we use Breakout, Asterix, SpaceInvaders, all implemented in the gymnas library (Lange, 2022b). Originally introduced by Young & Tian (2019), these tasks simplify the original Atari games by reducing the size of the grid and replacing RGB observations with pixel-centric symbolic information. Due to their lower complexity, they are often solved with feedforward networks by collapsing the input pixels. Here, we employ CNNs to extend the breadth of architectures considered in our study. Achieving a high score in these games requires sophisticated strategies to deal with complex challenges, such as exploration under high risk in Asterix, temporal credit assignment in Breakout and long-term planning in SpaceInvaders. Despite this reduction in complexity, these tasks still pose challenges for NE (Lange et al., 2023) and some RL methods.

As commonly done in continual learning studies, we chain these environments one after another in repeated phases with the order: Breakout, Asterix, SpaceInvaders. (Muppudi et al., 2024).

**Kinetix** Kinetix is a test-bed for testing the generalisation capabilities of RL agents (Matthews et al., 2024). It contains a variety of procedurally-generated and handcrafted environments that vary in their complexity and size. A Transformer-based architecture enables controlling robots of varying morphology and solving tasks differing in their input/output size. The ANN employed here is significantly larger than the networks we used in the previous tasks (about 750000 parameters). To the best of our knowledge, our study is the first to attempt optimizing a Transformer of this size with NE. As we will see, the GA exhibited impressive performance in these tasks while for ES we did not manage to find a well-working solution (even without shifts, see our note on tuning above).

To formulate a lifelong learning set-up, we employ the set of 20 manually designed tasks of medium difficulty and go through them in a sequence. While they have not been specifically designed to exhibit a curriculum, these tasks become increasingly progressive.

216 We refer to intervals between shifts in the environments as phases. When we employ the original  
 217 version of an environment, without environment shifts, we refer to it as the original version.  
 218

219 **3.2 OVERALL COMPARISON**

220 Figure 1 presents an overall comparison for the different methods and tasks. In particular, we visualize  
 221 the progression of rewards accumulated in a given episode across training, where the horizontal  
 222 axis indicates the number of environmental steps passed for a method. Shifts in the environment are  
 223 indicated with vertical dotted lines. For the RL methods, these values are computed by evaluating  
 224 the current policy in deterministic episodes, where we average across 20 seeds for the environment.  
 225 For the evolutionary approaches, the fitness of the best individual in the current generation/episode  
 226 is reported (which is an average across 20 environment seeds). This process is repeated 10 times  
 227 to get the reported means and confidence intervals. These results are accompanied by a table with  
 228 cumulative fitness scores and tests for statistical significance in Figure 7.  
 229

230 To facilitate our analysis, we also provide in Figure 5 of Appendix B the performance of methods  
 231 in the absence of environmental shifts. This information is necessary, as low performance in the  
 232 continual learning setting may be due to the inability of a method to master the task rather than the  
 233 added challenge of shifts.  
 234

235 **Overall failure of RL** We observe that methods exhibit significant differences in their perfor-  
 236 mance. PPO succeeds in the first phase but fails in subsequent ones, being the lowest performing  
 237 method in most tasks (the only exception is OpenES in Kinetix, but as we will see later, OpenES  
 238 could not master this task). As we see in Figure 5, PPO is the best-performing method in the nor-  
 239 mal versions of these tasks, so its failure is caused by the shifts. The failure of PPO is particularly  
 240 pronounced in the simple control tasks, where it performs worse than random search (Acrobot and  
 241 Cartpole can be easily solved through random search (Oller et al., 2020)). In Minatar, PPO con-  
 242 verges to complete failure in two of the tasks, while accumulating a small reward in one of them  
 243 (Breakout). This result indicates a complete loss of plasticity. In the Kinetix environments, PPO  
 244 solves some of the tasks (9 out of 20) in some trials, exhibiting high instability. In contrast, in the  
 245 original version of these tasks (Figure 5), PPO solves 19/20 tasks. Trac-PPO improves upon the per-  
 246 formance of PPO in all cases, but its performance remains unstable and lower than the one achieved  
 247 by the NE methods.  
 248

249 Turning to the NE methods, we observe that they both accumulate high rewards during environmen-  
 250 tal shifts, performing comparably to each other. The relative performance of these methods varies in  
 251 the original version of these tasks, so to compare them, we need to carefully examine each condition  
 252 in isolation.  
 253

254 **NE in lifelong classic control** We first turn to Acrobot, where we observe that both methods  
 255 perform well, with the GA exhibiting impressively steady good performance. In the absence of  
 256 shifts, both methods solve the task (this is true for all three tasks in this family) but ES converges  
 257 significantly more slowly, requiring about 200 generations. This is arguably the reason for its lower  
 258 performance in the lifelong setting: as we see in Figure 1 ES improves within each phase and  
 259 solves some phases perfectly.<sup>2</sup> In Cartpole, methods perform comparably and do not achieve a full  
 260 recovery (we will look more closely into this task and see how the GA can master it under long-term  
 261 evolution in Section 3.3). In MountainCar, we observe that OpenES outperforms the GA to some  
 262 extent. This is particularly interesting as, in the original version of this task, GA converges to the  
 263 optimal solution much quicker. This suggests that the GA is more challenged than ES when facing  
 264 environment shifts in this task. We will look deeper into this behavior in the next section  
 265

266 <sup>2</sup>Here we should note an important feature we noticed in our experiments: adding a vector to the observa-  
 267 tions of the environment can make a task more difficult for NE even in the absence of shifts. This is arguably  
 268 due to the change in the range of the observations NE methods received. Differently from RL, NE here and  
 269 commonly does not employ techniques for normalizing its inputs. This means that depending on its initializa-  
 270 tion it can start off in a bad region. Therefore, we should note that the performance reported for the original  
 271 version of these tasks is an upper bound rather than an expected value for the performance in phases during  
 272 lifelong learning.  
 273

270 **NE in lifelong Minatar** While NE methods were significantly outperformed by RL in the classical  
 271 version of these tasks, we see that in the continual version, both NE methods perform significantly  
 272 better. In particular, the GA exhibits the performance in the first task (Breakout) while ES exhibits  
 273 the best performance in the third task (SpaceInvaders). With the exception of a large drop in the  
 274 performance of the GA for SpaceInvaders, NE has remained largely unaffected (yet low-performing)  
 275 by the shifts. In the next section, we will closer into this failure of the GA and relate it to its failure  
 276 in MountainCar

277 **NE in Kinetix** ES did not manage to solve this task, but the GA exhibited impressive performance.  
 278 We observed that:

279

- 280 • For the small tasks, PPO-Transformer can find the optimal solution in both the normal and  
 281 lifelong set-up. Interestingly, the last two tasks are only solved in the lifelong set-up, which  
 282 means that the agent benefits from being pre-trained. Thus, lifelong learning is not an issue  
 283 here.
- 284 • The large tasks cannot be solved in the normal setup. It is likely that these tasks are too  
 285 difficult to solve without pre-training.
- 286 • For the medium tasks, we observe that the normal setup works well, but under lifelong  
 287 learning, performance degrades: the tasks are solved either much more slowly or are never  
 288 solved.

289 We use the manually designed tasks of medium size

290

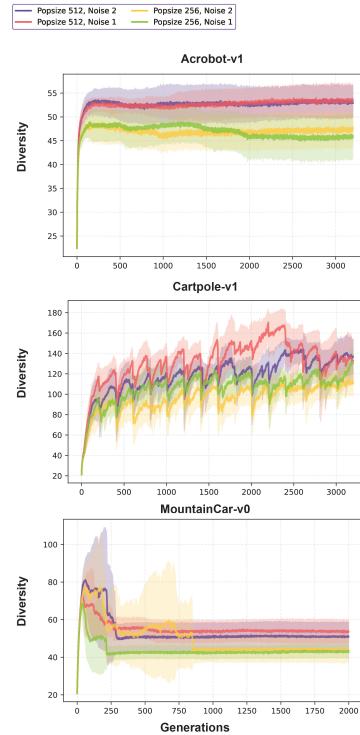
### 291 3.3 INSIGHTS INTO EVOLUTION

292 Having noted multiple intriguing behaviors in our overall  
 293 comparison, we turn towards a more in-depth look into the  
 294 dynamics of NE populations. Considering the surprising adaptability of NE, we would like to develop an intuition on how  
 295 populations react and manage to deal with shifts. We are  
 296 equally interested in understanding what happens when they  
 297 fail to do so.

298 Understanding why a method optimising a policy parameterised  
 299 by a neural network fails or succeeds in a control task  
 300 is not an easy feat. The complexity of environmental dynamics,  
 301 the black-box nature of ANNs, and our frequent lack of a  
 302 formal analysis of the search method stand in the way. Large  
 303 effort in the supervised and reinforcement learning community  
 304 is nevertheless put into developing analysis techniques  
 305 with many notable successes (Sokar et al., 2023).

306 Due to their population-based nature, NE approaches often lay  
 307 emphasis on properties such as the diversity of the population.  
 308 For some of the most popular approaches, such as Quality-  
 309 Diversity (Chalumeau et al., 2023), diversity is not just an af-  
 310 terthought but an explicit optimization objective or constraint.  
 311 When it comes to methods solely optimizing for performance,  
 312 studies rarely go into a post-hoc analysis. Exceptions to this  
 313 primarily come from work in evolutionary optimization that  
 314 employed small search spaces (Grove, 2014; Ouyang et al.,  
 315 2022). Applying such analysis in the search spaces considered  
 316 in modern NE significantly increases the memory and  
 317 computational requirements of a study. Thus, unsurprisingly  
 318 and to the best of our knowledge, studies of diversity in NE  
 319 remain underdeveloped.

320 Here we turn to such a study in the classic control tasks we  
 321 consider. We measure the diversity in a population as the mean



322 **Figure 2: Population diversity in**  
 323 **the classic control tasks.** We  
 324 study how diversity varies for two  
 325 different population sizes and two  
 326 levels of noise.

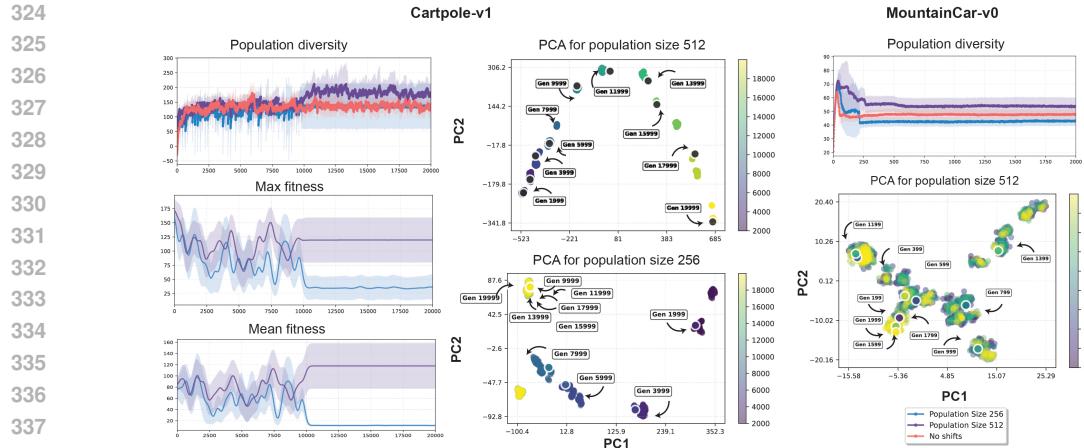


Figure 3: A diversity analysis of GA on (left) Cartpole, and (Right) Mountaincar.

pairwise distance computed in the original parameter space. In addition, we employ a dimensionality reduction technique (PCA) to visualize the trajectories that the population follows across evolution.

**Shifts promote diversity** In Figure 1 we observed that the GA remains robust to shifts in Acrobot but exhibits some instability in Cartpole. We found this intriguing as the two tasks exhibit similar dynamics and complexity. Upon monitoring how diversity evolves for these tasks (Figure 2) we see that the diversity in Acrobot stabilised early on, while it was still ascending for CartPole at the end of this experiment. This observation motivated us to run a very long experiment, simulating 100 environmental shifts. To our surprise, the population exhibited a stark transition around 900 generations (see Figure 3). Whether the transition was a favorable outcome or not depended on the population. For a large population (512 individuals) 8 out of 10 trials optimally solved the task, and the rest converged to a relatively high value. When the population was smaller (256 individuals), 9 out of 10 trials converged to a minimal fitness. Looking at the diversity in this longer experiment, we see that, right at this transition point, the large population abruptly increases its fitness. Overall, this analysis shows that diversity is driven by non-stationarity in the environment and that, in its turn, drives the ability of populations to avoid a mass extinction. Different from the GA, OpenES does not adapt by increasing its diversity but by adapting gradually (see Figure 6). This is not surprising as OpenES is a distribution-based approach that does not carry over the population across generations but spawns it, assuming a Gaussian distribution around its learned mean.

**Diversity is not sufficient** We now turn towards the MountainCar, one of the cases where the GA exhibits a lower ability at handling non-stationary. As we see on the bottom of Figure 2 and on the right of Figure 3, diversity here exhibits a form qualitatively different from the other two tasks. It increases initially (up to encountering the environment shift and then drops randomly. When looking at the PCA plots for this task, we see a starkly different behavior: there is no clear progression with generations. Our hypothesis is that, due to the sparse reward nature of this task, the population does not exhibit sufficient variance in its fitness for fitness-based selection to offer an improvement. When looking at the mean performance of the population in this task, we observe that all agents are failing to collect rewards.

## 4 DISCUSSION

Our study demonstrates that neuroevolution (NE) offers a compelling alternative to reinforcement learning (RL) for lifelong learning in dynamic environments. While RL methods such as PPO excel in stationary settings, they suffer from severe loss of plasticity when faced with environmental shifts, often converging to complete failure modes. In contrast, population-based NE approaches adapt naturally to change by maintaining and exploiting diversity within the population. This adaptability enables NE to discover specialist solutions after each shift rather than relying on a single, increasingly rigid generalist policy. These findings position NE not as a replacement for RL, but as

378 a complementary paradigm for situations where continual adaptation and long-term autonomy are  
 379 essential.

380 While NE proved remarkably robust, our analysis also revealed its limitations. In particular, the  
 381 genetic algorithm (GA) struggled in sparse-reward tasks such as MountainCar and certain Minatar  
 382 environments, where diversity alone was insufficient to guide evolution toward higher-fitness solu-  
 383 tions. Similarly, the evolution strategies (ES) method showed slower adaptation overall and failed  
 384 to handle high-dimensional architectures like the Transformer-based controller in Kinetix. Further-  
 385 more, our diversity analysis suggests that population size plays a critical role: larger populations  
 386 can undergo phase transitions leading to successful adaptation, whereas smaller populations risk  
 387 collapse under rapid environmental change.

388 Looking ahead, our results open several exciting research directions. Future work could explore the  
 389 co-evolution of mutation rates, network architectures, and even environmental complexity to further  
 390 improve plasticity and scalability.

391 We believe that our observation that increased diversity can be useful in non-stationary environ-  
 392 ments can offer insights that span beyond the field of NE and into RL. In particular, distributed RL  
 393 has shown promising results not just in helping scale up RL but also improving upon its perfor-  
 394 mance, with solutions hypothesizing that the benefits come from increased diversity (Horgan et al.,  
 395 2018) Investigating how NE can operate alongside RL in hybrid frameworks may yield agents that  
 396 balance the stability of gradient-based learning with the adaptability of evolutionary search. Finally,  
 397 expanding benchmarks to more open-ended environments and real-world robotics tasks will help  
 398 clarify the limits of NE in practical applications. By demonstrating that populations can overcome  
 399 the plasticity-stability dilemma without extensive tuning or external supervision, this work suggests  
 400 that evolution remains a powerful and underutilized tool for building lifelong learning systems.

401 Note: We have used LLMs solely for polishing text.

## 403 REFERENCES

405 Felix Chalumeau, Raphael Boige, Bryan Lim, Valentin Macé, Maxime Allard, Arthur Flajolet,  
 406 Antoine Cully, and Thomas Pierrot. Neuroevolution is a Competitive Alternative to Reinforce-  
 407 ment Learning for Skill Discovery, September 2023. URL <http://arxiv.org/abs/2210.03516> [cs].

409 Jeff Clune. AI-GAs: AI-generating algorithms, an alternate paradigm for producing general artificial  
 410 intelligence. *arXiv:1905.10985* [cs], January 2020. URL <http://arxiv.org/abs/1905.10985>. arXiv: 1905.10985.

413 Antoine Giraud, Ivan Matic, Olivier Tenaillon, Antonio Clara, Miroslav Radman, Michel Fons,  
 414 and François Taddei. Costs and benefits of high mutation rates: Adaptive evolution of bacteria  
 415 in the mouse gut. *Science*, 291(5513):2606–2608, 2001. doi: 10.1126/science.1056421. URL  
 416 <https://www.science.org/doi/abs/10.1126/science.1056421>.

417 Matt Grove. Evolution and dispersal under climatic instability: a simple evolutionary algo-  
 418 rithm. *Adaptive Behavior*, 22(4):235–254, August 2014. ISSN 1059-7123, 1741-2633. doi:  
 419 10.1177/1059712314533573. URL <http://journals.sagepub.com/doi/10.1177/1059712314533573>.

421 Dan Horgan, John Quan, David Budden, Gabriel Barth-Maron, Matteo Hessel, Hado van Has-  
 422 selt, and David Silver. Distributed Prioritized Experience Replay, March 2018. URL <http://arxiv.org/abs/1803.00933> [cs].

425 Edward Hughes, Michael Dennis, Jack Parker-Holder, Feryal Behbahani, Aditi Mavalankar, Yuge  
 426 Shi, Tom Schaul, and Tim Rocktaschel. Open-Endedness is Essential for Artificial Superhuman  
 427 Intelligence, June 2024. URL <http://arxiv.org/abs/2406.04268>. arXiv:2406.04268  
 428 [cs].

429 Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktaschel. A Survey of Zero-shot Gen-  
 430 eralisation in Deep Reinforcement Learning. *Journal of Artificial Intelligence Research*, 76:201–  
 431 264, January 2023. ISSN 1076-9757. doi: 10.1613/jair.1.14174. URL <http://arxiv.org/abs/2111.09794> [cs].

432 James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A  
 433 Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcom-  
 434 ing catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*,  
 435 114(13):3521–3526, 2017.

436

437 Timo Klein, Lukas Miklautz, Kevin Sidak, Claudia Plant, and Sebastian Tschiatschek. Plasticity  
 438 Loss in Deep Reinforcement Learning: A Survey, November 2024. URL <http://arxiv.org/abs/2411.04832> [cs].

439

440 John R Koza. GENETIC PROGRAMMING: A PARADIGM FOR GENETICALLY BREEDING  
 441 POPULATIONS OF COMPUTER PROGRAMS TO SOLVE PROBLEMS.

442

443 Dhiresha Kudithipudi, Mario Aguilar-Simon, Jonathan Babb, Maxim Bazhenov, Douglas Black-  
 444 iston, Josh Bongard, Andrew P Brna, Suraj Chakravarthi Raja, Nick Cheney, Jeff Clune, et al.  
 445 Biological underpinnings for lifelong learning machines. *Nature Machine Intelligence*, 4(3):196–  
 446 210, 2022.

447

448 Robert Tjarko Lange. evosax: Jax-based evolution strategies. *arXiv preprint arXiv:2212.04180*,  
 449 2022a.

450

451 Robert Tjarko Lange. gymnasx: A JAX-based reinforcement learning environment library, 2022b.  
 452 URL <http://github.com/RobertTLange/gymnasx>.

453

454 Robert Tjarko Lange, Yujin Tang, and Yingtao Tian. NeuroEvoBench: Benchmarking Evolutionary  
 455 Optimizers for Deep Learning Applications. November 2023. URL <https://openreview.net/forum?id=s6qtLyR6uJ>.

456

457 Joel Lehman and Risto Miikkulainen. Extinction Events Can Accelerate Evolution. *PLOS ONE*,  
 10(8):e0132886, August 2015. ISSN 1932-6203. doi: 10.1371/journal.pone.0132886. URL  
 458 <https://dx.plos.org/10.1371/journal.pone.0132886>.

459

460 Michael Matthews, Michael Beukman, Chris Lu, and Jakob Nicolaus Foerster. Kinetix: Investigat-  
 461 ing the Training of General Agents through Open-Ended Physics-Based Control Tasks. October  
 462 2024. URL <https://openreview.net/forum?id=zCxGCDzreM>.

463

464 Michael Matthews, Michael Beukman, Chris Lu, and Jakob Foerster. Kinetix: Investigating the  
 465 training of general agents through open-ended physics-based control tasks. 2025. URL <https://arxiv.org/abs/2410.23208>.

466

467 Michael McCloskey and Neal J. Cohen. Catastrophic Interference in Connectionist Net-  
 468 works: The Sequential Learning Problem. In Gordon H. Bower (ed.), *Psychology of Learn-  
 469 ing and Motivation*, volume 24, pp. 109–165. Academic Press, January 1989. doi: 10.  
 470 1016/S0079-7421(08)60536-8. URL <https://www.sciencedirect.com/science/article/pii/S0079742108605368>.

471

472 Aneesh Muppidi, Zhiyu Zhang, and Heng Yang. Fast TRAC: A Parameter-Free Optimizer for  
 473 Lifelong Reinforcement Learning. November 2024. URL <https://openreview.net/forum?id=QEaHE4TUgc&noteId=Oot4q3xdzJ>.

474

475 Eleni Nisioti and Clément Moulin-Frier. Plasticity and evolvability under environmental vari-  
 476 ability: the joint role of fitness-based selection and niche-limited competition. In *Proceedings  
 477 of the Genetic and Evolutionary Computation Conference*, GECCO '22, pp. 113–121, New  
 478 York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392372. doi:  
 479 10.1145/3512290.3528826. URL <https://doi.org/10.1145/3512290.3528826>.

480

481 Declan Oller, Tobias Glasmachers, and Giuseppe Cuccu. Analyzing Reinforcement Learning  
 482 Benchmarks with Random Weight Guessing. In *Proceedings of the 19th International Con-  
 483 ference on Autonomous Agents and MultiAgent Systems*, AAMAS '20, pp. 975–982, Richland,  
 484 SC, May 2020. International Foundation for Autonomous Agents and Multiagent Systems. ISBN  
 485 978-1-4503-7518-4.

486 Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong  
 487 Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kel-  
 488 ton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike,  
 489 and Ryan Lowe. Training language models to follow instructions with human feedback, March  
 490 2022. URL <http://arxiv.org/abs/2203.02155>. arXiv:2203.02155 [cs].  
 491

492 Chaofan Pan, Xin Yang, Yanhua Li, Wei Wei, Tianrui Li, Bo An, and Jiye Liang. A Survey of Con-  
 493 tinual Reinforcement Learning, June 2025. URL <http://arxiv.org/abs/2506.21872>.  
 494 arXiv:2506.21872 [cs].

495 Sebastian Risi, David Ha, Yujin Tang, and Risto Miikkulainen. *Neuroevolution: Harnessing Cre-  
 496 ativity in AI Model Design*. MIT Press, Cambridge, MA, 2025. URL <http://www.cs.utexas.edu/users/ai-lab?risi:book25>.  
 497

498 Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution Strategies as  
 499 a Scalable Alternative to Reinforcement Learning, September 2017. URL <http://arxiv.org/abs/1703.03864>. arXiv:1703.03864 [stat].  
 500

502 John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy  
 503 optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017a.

504 John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy  
 505 Optimization Algorithms, August 2017b. URL <http://arxiv.org/abs/1707.06347>.  
 506 arXiv:1707.06347 [cs].  
 507

508 David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche,  
 509 Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman,  
 510 Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine  
 511 Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go  
 512 with deep neural networks and tree search. *Nature*, 529(7587):484–489, January 2016a. ISSN  
 513 1476-4687. doi: 10.1038/nature16961. URL <https://www.nature.com/articles/nature16961>. Number: 7587 Publisher: Nature Publishing Group.  
 514

515 David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George van den Driessche,  
 516 Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneershelvam, Marc Lanctot, Sander  
 517 Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P Lillicrap,  
 518 Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game  
 519 of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016b. doi:  
 520 10.1038/nature16961.

521 Ghada Sokar, Rishabh Agarwal, Pablo Samuel Castro, and Utku Evci. The Dormant Neuron Phe-  
 522 nomenon in Deep Reinforcement Learning, June 2023. URL <http://arxiv.org/abs/2302.12902>. arXiv:2302.12902 [cs].  
 523

524 Lisa Kenneth O. Stanley Soros, Joel Lehman. Open-endedness: The last grand challenge  
 525 you've never heard of, December 2017. URL <https://www.oreilly.com/radar/open-endedness-the-last-grand-challenge-youve-never-heard-of/>.  
 526

527 Kenneth O. Stanley, Jeff Clune, Joel Lehman, and Risto Miikkulainen. Designing neural networks  
 528 through neuroevolution. *Nature Machine Intelligence*, 1(1):24–35, January 2019. ISSN 2522-  
 529 5839. doi: 10.1038/s42256-018-0006-z. URL <https://www.nature.com/articles/s42256-018-0006-z>. Publisher: Nature Publishing Group.  
 530

531 Felipe Petroski Such, Vashisht Madhavan, Edoardo Conti, Joel Lehman, Kenneth O. Stanley, and  
 532 Jeff Clune. Deep Neuroevolution: Genetic Algorithms Are a Competitive Alternative for Training  
 533 Deep Neural Networks for Reinforcement Learning, April 2018. URL <http://arxiv.org/abs/1712.06567>. arXiv:1712.06567 [cs].  
 534

535 Rui Wang, Joel Lehman, Jeff Clune, and Kenneth O. Stanley. Paired Open-Ended Trailblazer  
 536 (POET): Endlessly Generating Increasingly Complex and Diverse Learning Environments and  
 537 Their Solutions. *arXiv:1901.01753 [cs]*, February 2019. URL <http://arxiv.org/abs/1901.01753>. arXiv: 1901.01753.  
 538