HELI: Stabilising Structural Mutation in Neuroevolution through Lineage Incubation

Stefan Sieg
evolib@dismail.de

December 2025

Abstract

Structural mutations in neuroevolution are fragile: topological edits disrupt activation flow, degrade fitness immediately, and are therefore removed by selection before they can mature. We introduce Hierarchical Evolution with Lineage Incubation (HELI), a mechanism that evaluates structurally mutated individuals in temporary sub-populations before reintegration into the main search. This brief incubation period provides local adaptation time and stabilises new topologies before they re-enter global selection. Across three small deterministic benchmarks (classification, sequential prediction, and regression) HELI retains more structural mutations and achieves lower error than both weight-only evolution and direct structural mutation under equalised evaluation budgets. On these benchmarks, HELI offers a simple mechanism that improves the stability and practical usability of structural mutation.

1 Introduction

While evolutionary algorithms are widely used to optimise neural-network weights, evolving architectures remains challenging due to the fragility of structural mutations. Weight perturbations are typically preserved by selection, whereas structural edits (adding or removing neurons or connections) often cause immediate fitness degradation by disrupting the activation flow and introducing unoptimised parameters.

As a result, most structural changes fail to produce functional improvements. In our benchmarks, direct structural mutation frequently yields altered or enlarged architectures without corresponding error reduction, illustrating the instability of unguided topological edits.

This work introduces *Hierarchical Evolution with Lineage Incubation* (HELI), a mechanism that temporarily isolates structurally mutated individuals in short-lived sub-populations. During a brief incubation phase, these lineages evolve under relaxed competitive pressure before each lineage's best individual is reintegrated into the main population. HELI does not modify mutation operators or the fitness function; instead, it provides a limited period of local adaptation before global selection resumes.

We evaluate HELI on minimal diagnostic benchmarks designed to probe sensitivity to structural mutation. Across these tasks, HELI retains more structural innovations, exhibits sustained architectural growth, and achieves lower error than both fixed-topology evolution and direct structural mutation under evaluation-budget normalisation.

The remainder of this paper is organised as follows: Section 2 reviews related work, Section 3 describes the HELI mechanism, Section 4 presents the experimental setup and results, and Section 5 concludes. Additional analyses and supplementary figures are provided in the Appendix.

2 Related Work

Protecting structural innovations

NEAT [9] introduced speciation and historical markings to protect topological innovations by restricting competition primarily within species. Related diversity-driven approaches such as Novelty Search [3] and Quality-Diversity methods (e.g. MAP-Elites [5]) reduce objective-driven selection pressure and thereby implicitly protect newly generated structures. Indirect encodings such as CPPNs and HyperNEAT [10] avoid explicit structural edits by generating architectures from compositional pattern networks, trading direct topological control for representational regularity.

Complexification and structural growth

Methods based on complexification progressively expand network structure during evolution [9, 10]. Topological growth is coupled with speciation to prevent premature rejection of larger or newly formed architectures. However, these mechanisms rely on long-term species protection rather than short-term localised adaptation and do not isolate structurally altered individuals for dedicated recovery.

Modern weight-space methods

Large-scale evolutionary strategies (CMA-ES [1], OpenAI-ES [7], NES [12]) excel at optimising high-dimensional continuous parameters but assume fixed architectures. As a result, direct structural mutations are rarely used in practice because they introduce abrupt functional changes that these methods are not designed to accommodate.

Stabilising disruptive mutations

Safe Mutation techniques [4] adjust parameter perturbations to limit functional disruption caused by weight mutations and, in some cases, implicitly mitigate the effects of structural edits. Aging Evolution [6] protects young individuals by reducing selective pressure on recently created genomes. Other work on regularised or constrained mutation magnitudes similarly aims to stabilise search dynamics but does not provide isolated recovery for structurally altered individuals.

Multi-population and age-structured evolutionary systems

Island models [11] and distributed evolutionary approaches maintain multiple partially isolated subpopulations to balance exploration and exploitation. Age-Layered Population Structure (ALPS) [2] separates individuals by evolutionary age to shield young genomes from premature selection pressure. These methods operate at the population level and protect global diversity, but do not provide short-lived, mutation-triggered isolation, nor do they target structural mutation specifically. In contrast, HELI performs temporary lineage incubation at the level of individual structural mutants, offering localised recovery without altering global selection dynamics or introducing persistent subpopulations.

Incubation and localised adaptation

Only limited prior work explores mechanisms conceptually related to HELI. Staged evaluation or localised subpopulation updates appear in some multi-population evolutionary systems, but typically serve exploration–exploitation balancing rather than recovery from structural disruption. To our knowledge, no existing approach explicitly performs *temporary incubation of structural mutants* in short-lived isolated lineages to allow local adaptation before reintroducing them into the main population.

Positioning of HELI

HELI provides a complementary mechanism to existing approaches for protecting structural innovations. Unlike speciation, safe mutation, and complexification, HELI does not modify mutation operators, mutation magnitudes, or global selection pressure. Instead, it isolates structurally mutated individuals into short-lived lineage populations that evolve for a limited number of generations before re-entering the main evolutionary loop. This mechanism targets the short-term instability introduced by structural edits and offers localised recovery at the level of individual mutants, without introducing persistent subpopulations or long-term protective structures.

3 Methods

3.1 HELI Overview

Structural mutations in neuroevolution often cause immediate fitness degradation because newly introduced neurons and connections disrupt the existing activation flow and are not yet supported by optimized parameters. As a result, structurally mutated individuals are typically removed before potential benefits can emerge. Existing approaches mitigate this effect by modifying mutation operators or by relaxing selection pressure globally.

HELI takes a different direction: structurally mutated individuals are isolated in short-lived lineage populations. Each lineage is given a limited adaptation phase before reintegration into the main evolutionary loop. This temporary relaxation of selection pressure allows structural innovations to stabilise locally without altering mutation operators or fitness evaluation.

3.2 HELI Algorithm

After offspring generation, all structurally mutated individuals are removed from the offspring set and placed into temporary lineage populations. Each lineage evolves independently for T_{inc} incubation generations. Structural mutation is disabled during incubation to prevent compounding topological disruptions. After incubation, the best individual of each lineage is reintegrated into the offspring set. Standard selection then forms the next main population. HELI introduces no new evolutionary operators; its effect results solely from the temporary decoupling of structural exploration and main-loop selection.

Algorithm 1 HELI extension to a standard evolutionary loop

Initialize population P
for each generation **do**
  Generate offspring O using weight and structural mutations
  $S \leftarrow \{\, o \in O \mid o \text{ underwent structural mutation} \,\}$
  $O \leftarrow O \setminus S$
  for each $s \in S$ **in parallel do**
    Create lineage $L_s \leftarrow \{s\}$
    for $t = 1$ to T_{inc} **do**
      Evolve L_s (structural mutation disabled)
    end for
    $c_s \leftarrow$ best individual in L_s
  end for
  $O \leftarrow O \cup \{\, c_s \mid s \in S \,\}$
  Select next population P from $P \cup O$
end for

3.3 Implementation Details

HELI is implemented as an optional module in EvoLib [8]. Structural mutants are marked with metadata and automatically routed into lineage populations. Lineage evolution uses the same mutation operators, fitness functions, and seed hierarchy as the main evolutionary loop; only structural mutation is disabled during incubation.

Importantly, HELI itself does *not* prescribe any modification to weight-mutation magnitudes or operator probabilities. The mechanism is defined solely by (i) isolating structurally mutated individuals, (ii) evolving each lineage for T_{inc} generations with structural mutation disabled, and (iii) reintegrating the best individual into the main offspring set.

All additional design decisions used in our experiments such as reducing the weight mutation strength inside lineages (Section 4), are part of the experimental configuration and not inherent to the HELI mechanism.

3.4 Evaluation Budget and HELI Overhead

HELI does not modify mutation operators or the fitness function, but it introduces additional evaluations because structurally mutated individuals are incubated before reintegration. Structural mutation itself does not increase the evaluation count, since all variants (Baseline, NoHELI, HELI) generate the same number of offspring per generation and each offspring is evaluated exactly once.

For HELI, additional evaluations arise exclusively from incubating structurally mutated individuals. The total evaluation count is:

$$E_{\text{HELI}} = E_{\text{base}} + \sum_{g=1}^{G} N_{\text{mut}}^{(g)} \cdot T_{\text{inc}} \cdot \mu_{\text{inc}},$$

where $N_{\text{mut}}^{(g)}$ is the number of structural mutants in generation g, T_{inc} is the incubation duration, and μ_{inc} denotes the lineage population size (one parent and eight offspring, totalling nine individuals).

Although the EvoLib implementation includes optional early-stopping heuristics to reduce lineage evaluation overhead, these mechanisms were disabled in all experiments to keep evaluation budgets strictly comparable across variants.

The resulting overhead depends on the frequency of structural mutations and the duration for which individual lineages remain active. Table 1 summarises the mean evaluation counts per variant across all seeds. To ensure fair comparison in the Results section, all fitness curves are aligned by cumulative evaluation count so that Baseline, NoHELI, and HELI operate under identical computational budgets.

Table 1 quantifies the additional evaluations introduced by lineage incubation. When performance is expressed as a function of cumulative evaluation count, HELI achieves lower error at equal evaluation budgets, indicating higher sample efficiency within the observed range.

Table 1: Evaluation budget per variant (mean over all seeds).

Task	Baseline	NoHELI	HELI	Overhead
Parity-3	12000	12000	83212	6.93
Delay-5	12000	12000	83760	6.98
Sine	12000	12000	82795	6.89

3.5 Reproducibility

All configuration files, seeds, and scripts used in this study are available in the EvoLib repository. Exact configurations and logs for all experiments are archived on Zenodo (DOI: 10.5281/zenodo.17880334).

4 Experiments

4.1 Experimental Setup

All experiments use a $(\mu + \lambda)$ evolutionary strategy with identical hyperparameters and evaluation procedures. Results are reported over 30 independent random seeds (first 30 prime numbers).

We compare three variants:

- **Baseline**: weight mutation only,
- **NoHELI**: structural mutation without incubation,
- **HELI**: structural mutation with incubation.

Structural mutation operators include adding or removing neurons and connections. In HELI, each structurally mutated individual spawns a temporary lineage that evolves for $T_{\text{inc}} = 10$ generations using a $(1 + 8)$ strategy with reduced weight–mutation strength (scaled by 0.5). This reduction is part of the experimental setup and not inherent to the HELI mechanism itself. After incubation, the best individual from each lineage is reintegrated into the main offspring set.

Architectural sizes (hidden-unit counts) differ across tasks but remain identical across variants within each task. The following initialization regimes are used across tasks:

Fully connected feedforward initialization: For Parity-3 and Sine, all variants (Baseline, NoHELI, HELI) start from the same fully connected feedforward architecture of sufficiently large size for the task. This provides a uniform initialization without restricting structural exploration.

Discovery-based initialization: For Delay-5, the Baseline uses a fixed fully recurrent starting architecture, while the structural variants (NoHELI and HELI) begin without recurrent connections and must evolve recurrence through structural

mutation. This creates a setting in which memory mechanisms must be discovered rather than provided a priori.

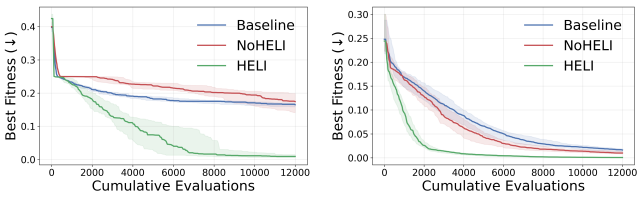
Only the structural variants (NoHELI and HELI) are allowed to modify their architectures; the Baseline retains its initial topology throughout. Baseline and NoHELI perform exactly 12 000 fitness evaluations. HELI incurs additional evaluations due to the incubation process.

4.2 Tasks

We deliberately restrict our evaluation to small, deterministic benchmarks that are sufficiently challenging for plain $(\mu + \lambda)$ neuroevolution without gradient information, yet simple enough to allow controlled analysis of structural mutation dynamics. The purpose of these tasks is not to compete with large-scale deep neuroevolution on complex RL environments, but to isolate the contribution of HELI to stabilising structural mutations under strictly controlled and fully reproducible conditions.

- **Parity-3**: a deterministic 3-bit parity classification task that requires precise nonlinear feature interactions and is highly sensitive to topological disruptions, making it an effective diagnostic for structural stability.
- **Delay-5**: a sequential prediction task where the network must output the input from five steps earlier (x_{t-5}) over fixed-length sequences of 512 steps. Successful optimisation depends on maintaining stable temporal dependencies despite structural modifications.
- **Sine**: regression of $\sin(x)$ on 80 fixed input points. Structural mutations can introduce local distortions in the smooth approximation landscape, making this task a controlled probe for function-approximation stability.

4.3 Results



(a) Delay-5 (b) Sine

Figure 1: Budget-normalised fitness (median ± IQR).

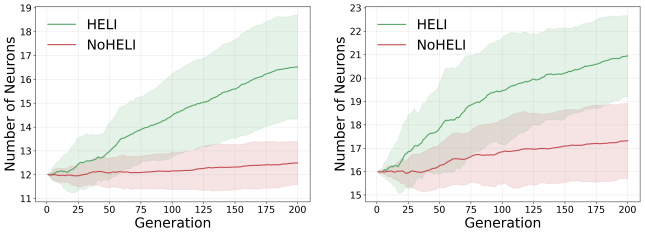
All performance curves are normalised by cumulative evaluation count to ensure fair comparison across variants. HELI incurs a bounded overhead due to temporary incubation; the exact evaluation budgets are reported in Table 1.

The Parity-3 task is omitted from the main figures because it offers only limited structural complexity and shows no meaningful structural growth. HELI reliably reaches zero error across seeds, whereas Baseline and NoHELI converge to low but non-zero error with similarly small architectures. The task is included as a diagnostic control to assess convergence behaviour under minimal structural demands. A summary of the Parity-3 results is provided in the Appendix.

Figure 1 shows the budget-normalised performance for Delay-5 and Sine. HELI achieves the lowest error across the evaluation budget in both tasks. Baseline improves steadily but remains worse than HELI throughout. NoHELI shows consistently higher median error. Its optimisation dynamics are less stable, particularly in the Delay-5 task, where integrating structural mutants without incubation leads to frequent disruptions in fitness progress.

Beyond the median trends, the curves mainly reflect how structural mutations interact with the selection process. In NoHELI, structurally mutated offspring often show reduced fitness in the immediate generation after mutation. Because these individuals are evaluated in the main loop without any stabilisation phase, disruptive edits directly affect the population and slow optimisation, which appears as small regressions and delayed convergence, particularly in Delay-5. HELI avoids this effect by separating structurally mutated offspring from the main population. As a result, disruptive edits do not influence global selection, and beneficial structures can recover fitness before re-entering the search. This leads to smoother fitness trajectories and task-dependent variability across seeds: HELI stabilises the dynamics on Sine, while Delay-5 shows broader divergence due to structural exploration.

The Sine task shows a milder effect: NoHELI exhibits greater variability across seeds, whereas HELI remains more consistent and improves more steadily. Together, these results indicate that temporarily isolating structurally mutated individuals reduces the immediate fitness disruptions that would otherwise suppress structural exploration.



(a) Delay-5 (b) Sine

Figure 2: Structural growth (neurons; mean ± SD).

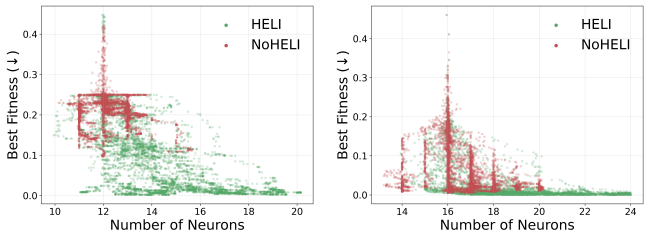
Structural properties differ substantially across variants. HELI shows a steady increase in the number of neurons over generations, whereas NoHELI exhibits only small and irregular changes (Fig. 2). Consequently, HELI explores a wider range of network sizes than NoHELI, consistent with its sustained structural growth. The monotonic structural expansion

in HELI also aligns with the stabilised fitness trajectories: additional capacity is not discarded but progressively incorporated into the search. In NoHELI, by contrast, structural mutations rarely persist long enough to provide functional benefit, leading to minimal and highly variable growth.

Table 2: Budget-normalised final error at 12 000 evaluations (30 seeds).

Variant	Median	IQR	Mean	SD
Delay-5				
Baseline	0.1649	0.0117	0.1638	0.0094
NoHELI	0.1744	0.0570	0.1706	0.0360
HELI	0.0092	0.0093	0.0168	0.0247
Parity-3				
Baseline	0.0625	0.1250	0.0625	0.0625
NoHELI	0.1250	0.1250	0.0750	0.0612
HELI	0.0000	0.0000	0.0000	0.0000
Sine				
Baseline	0.0163	0.0063	0.0168	0.0069
NoHELI	0.0099	0.0060	0.0109	0.0038
HELI	0.0005	0.0006	0.0006	0.0005

Table 2 summarises the budget-normalised error at 12 000 evaluations. HELI achieves lower median error and reduced variability compared to NoHELI on Delay-5 and Sine, consistent with the trends observed in Fig. 1. These values provide a concise reference point for the curves shown above and complement the qualitative analysis of optimisation dynamics.



(a) Delay-5 (b) Sine

Figure 3: Fitness as a function of structural size (neurons).

The functional relationship between network size and fitness is shown in Fig. 3. In NoHELI, variations in network size do not correspond to improvements in error. HELI maintains low error across a broad range of network sizes, suggesting effective utilisation of the additional capacity on these tasks. The consistency of this trend across both tasks indicates that HELI allows structural innovations to become functionally meaningful rather than disruptive.

5 Conclusion

We introduced HELI, an incubation-based mechanism for stabilising structural mutation in evolutionary neural optimisation. By temporarily isolating structurally modified individuals in short-lived lineage populations, HELI reduces the short-term disruption caused by topological edits and enables their integration without altering mutation operators or the objective function.

Across three controlled benchmarks (deterministic classification, sequential prediction, and continuous regression) HELI achieves lower error under evaluation-budget normalisation compared to variants without incubation.

The benchmarks used in this study are intentionally low-dimensional and deterministic. They pose meaningful challenges for plain $(\mu + \lambda)$ evolution but do not represent large-scale or high-dimensional real-world problems. Accordingly, this work should be viewed as an empirical demonstration of HELI under controlled diagnostic conditions rather than a definitive statement about its behaviour in deep neuroevolution.

HELI incurs additional evaluation cost because structurally mutated individuals are evolved within lineage populations before reintegration. Incubation is applied uniformly, without attempting to identify whether a given structural change is beneficial or detrimental.

While HELI shows consistent benefits on the examined diagnostic tasks, its behaviour on higher-dimensional, stochastic, or more complex temporal problems remains an open question. A natural direction for future work is to evaluate HELI under richer temporal structure, moderate noise, and larger function classes to assess the robustness and scalability of the incubation principle beyond the controlled settings studied here.

Overall, HELI provides a simple mechanism for stabilising structural mutation in neuroevolution. On the examined benchmarks, it supports reliable optimisation and enables structured exploration of architectural change.

Acknowledgements

This work received no external funding. Compute resources were provided by personal hardware.

References

[1] Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001. doi: 10.1162/106365601750190398.

[2] Gregory S. Hornby. Alps: The age-layered population structure for reducing premature convergence. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 815–822. ACM, 2006. doi: 10.1145/1143997.1144142.

[3] Joel Lehman and Kenneth O. Stanley. Abandoning objectives: Evolution through the search for novelty alone.

Evolutionary Computation, 19(2):189–223, 2011. doi: 10.1162/evco_a_00025.

[4] Joel Lehman, Jay Chen, Jeff Clune, and Kenneth O. Stanley. Safe mutations for deep and recurrent neural networks through output gradients. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 117–124. ACM, 2018. doi: 10.1145/3205455.3205473.

[5] Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites, 2015.

[6] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V. Le. Regularized evolution for image classifier architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4780–4789. AAAI Press, 2019. doi: 10.1609/aaai.v33i01.33014780.

[7] Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning, 2017.

[8] Stefan Sieg. Evolib: A modular framework for evolutionary computation. Zenodo Repository, 2025. URL https://doi.org/10.5281/zenodo.17793862. Version 0.2.0b3.

[9] Kenneth O. Stanley and Risto Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2):99–127, 2002. doi: 10.1162/106365602320169811.

[10] Kenneth O. Stanley, David B. D'Ambrosio, and Jason Gauci. A hypercube-based encoding for evolving large-scale neural networks. *Artificial Life*, 15(2):185–212, 2009. doi: 10.1162/artl.2009.15.2.15202.

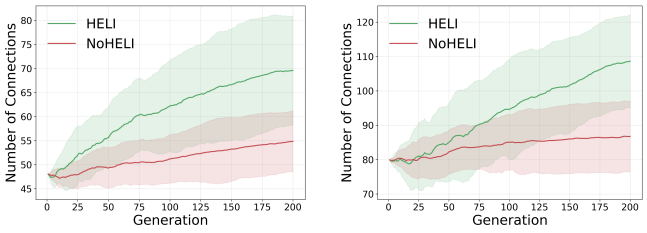
[11] L. Darrell Whitley, Soraya B. Rana, and Robert B. Heckendorn. Island model genetic algorithms and linearly separable problems. In *Evolutionary Computing: AISB Workshop*, pages 109–125, 1997.

[12] Daan Wierstra, Tom Schaul, Tobias Glasmachers, Yi Sun, Jan Peters, and Jürgen Schmidhuber. Natural evolution strategies. *Journal of Machine Learning Research*, 15(1): 949–980, 2014. URL https://www.jmlr.org/papers/v15/wierstra14a.html.

A Appendix

A.1 Additional Structural Metrics

Figure A.1 shows the number of active connections over generations, reported as mean values across 30 seeds with one standard deviation. Both tasks show the same qualitative pattern observed for neuron counts: HELI exhibits sustained structural expansion with broader variability across seeds, while NoHELI remains near its initial connectivity with only minor fluctuations.



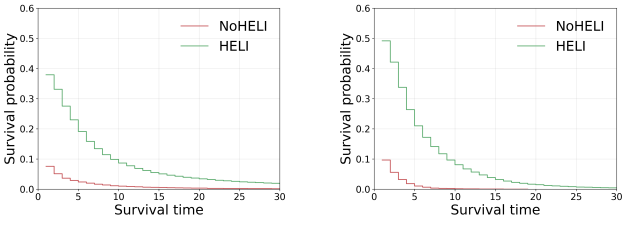
(a) Delay-5 (b) Sine

Figure A.1: Connection counts over generations (mean ± SD).

A.2 Survival Analysis of Structural Mutants

Survival time is defined as the number of main-loop generations in which a structural mutant remains present in the population (survival_time = exit_gen - birth_gen + 1). Survival time quantifies how long structurally altered individuals remain in the population before elimination by selection. HELI produces a markedly heavier survival tail than NoHELI in both tasks, reflecting the greater persistence of structural mutants once short-term disruption has been mitigated.

Kaplan–Meier curves

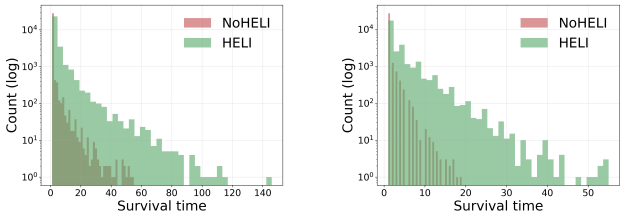


(a) Delay-5 (b) Sine

Figure A.2: Kaplan–Meier survival curves for structural mutants (individual level). The survival-time axis is truncated for clarity to highlight early survival dynamics.

Figure A.2 reports the survival probability of structural mutants across main-loop generations. HELI exhibits a noticeably slower decay in survival probability, while NoHELI curves drop rapidly, indicating short persistence of structural mutants in the main population.

Log-scale histograms.

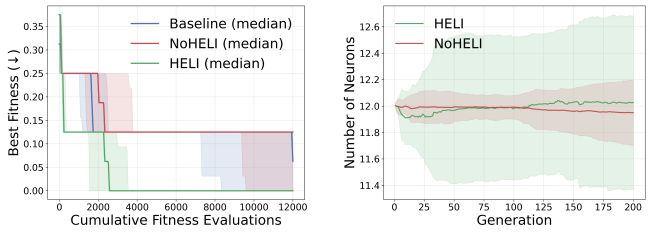


(a) Delay-5 (b) Sine

Figure A.3: Survival-time distributions on a log scale (individual level).

Figure A.3 summarises the distribution of individual survival times for structural mutants. HELI shows a broader distribution with substantially more mutants surviving beyond three generations, whereas NoHELI is dominated by survival times of exactly one generation.

A.3 Parity-3 (Control Task)



(a) Budget-normalised fitness (median ± IQR). (b) Structural size (neurons; mean ± SD).

Figure A.4: Parity-3 control task: budget-normalised fitness and structural size.

Parity-3 serves as a control task with minimal structural requirements. All variants converge rapidly to near-zero error under the same evaluation budget (Fig. A.4, left). Structural size remains essentially constant across generations for all variants, and HELI does not introduce additional growth relative to Baseline or NoHELI (Fig. A.4, right). This confirms that HELI does not introduce unnecessary architectural expansion when structural mutations are not beneficial. For this reason, Parity-3 is omitted from the main Results section and reported only as a control task in the Appendix.

A.4 Ablation Studies

We report three ablations examining the sensitivity of HELI to core internal parameters. All experiments use Delay-5 as the diagnostic task.

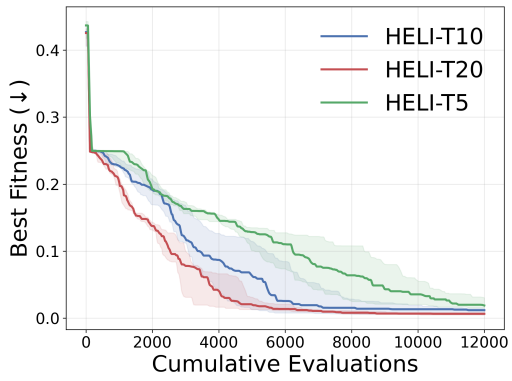


Figure A.5: Sensitivity of HELI to different incubation lengths on Delay-5.

Incubation Length

We evaluated HELI with incubation lengths $T_{\text{inc}} \in \{5, 10, 20\}$ on the Delay-5 benchmark in order to assess the sensitivity of the mechanism to this parameter. All variants achieve stable convergence and reach comparable final error under the same evaluation budget. Longer incubation generally improves convergence and reduces variance, with $T_{\text{inc}} = 10$ and $T_{\text{inc}} = 20$ performing similarly and consistently better than $T_{\text{inc}} = 5$. These results indicate that HELI is robust to reasonable changes in incubation length and does not rely on a narrowly tuned setting.

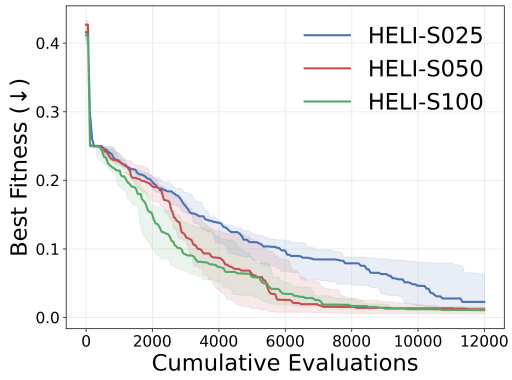


Figure A.6: Sensitivity of HELI to different mutation strengths within lineages.

Mutation Strength in Lineages

We varied the mutation strength within lineage populations using $\sigma \in \{0.25, 0.5, 1.0\}$ to assess the effect of local exploration intensity on HELI's behaviour. All settings converge reliably under the same evaluation budget, indicating that HELI does not rely on a narrowly tuned mutation scale. Lower mutation strength results in more conservative adaptation, while higher mutation strength increases variability but preserves the overall convergence trend. Taken together, the results show that HELI remains stable across a wide range of mutation strengths.

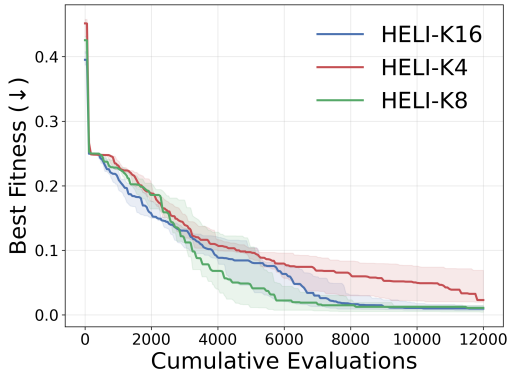


Figure A.7: Sensitivity of HELI to the number of offspring generated per lineage.

Lineage Offspring

To examine the role of local population size during incubation, we varied the number of offspring generated per lineage generation using $\{4, 8, 16\}$. All configurations reach comparable final error under the same evaluation budget, demonstrating that HELI is not sensitive to the specific choice of lineage offspring count. Smaller offspring counts reduce the capacity for rapid recovery after disruptive structural edits, whereas larger offspring counts increase exploration and variance. Overall, HELI maintains robust behaviour across a practical range of local population sizes.