COLLABORATIVE PRINCIPLE EVOLUTION VIA EVIDENCE TRANSFER FOR SCIENTIFIC DISCOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Model (LLM)-based agents promise to automate scientific discovery, yet exploring the vast hypothesis space remains costly. Existing principle-evolution methods accelerate this loop, but operate sequentially, which caps exploration breadth and wastes wall-clock time on challenging problems. To address this, we formulate collaborative scientific discovery as *evidence transfer* between parallel principle-evolution branches. We present **COEVOLVE**, which realizes this transfer through a coordination core over parallel branches. By integrating value-of-information-gated routing and context-discounted likelihood injection, COEVOLVE enables branches to collaborate through shared measurements while keeping their principle posteriors separate. Across six scientific-discovery tasks under a matched evaluation budget, COEVOLVE attains a mean solution quality of 66.5% versus 57.0% for single-branch principle evolution, with a $1.80\times$ mean wall-clock speedup on the GPT-5.6-Terra backbone; on five auto-research tasks delegated to an autonomous research harness, it is the only arm whose mean stays above the published SOTA anchor on every task. These results establish when evidence sharing accelerates parallel discovery and when transfer safeguards are necessary to limit negative or inert transfers.

1 INTRODUCTION

The large language model (LLM)-based AI Scientist is often realized as a hypothesis-driven loop: an agent proposes a hypothesis, evaluates it, and designs the next from observations (Wei et al., 2025; Gridach et al., 2025; Herron et al., 2026; Boiko et al., 2023; Mitchener et al., 2025). While these systems automate discovery (Ghareeb et al., 2025; Ghafarollahi & Buehler, 2024; Li et al., 2026), each still operates alone. Yet real discovery is rarely solitary, so how could AI Scientists collaborate on the same problem?

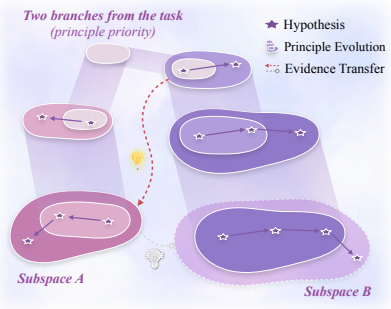


Figure 1: **Principle co-evolution.** Independent branches explore different principle subspaces through a hypothesis-testing loop. COEVOLVE shares evidence, allowing the branches to collaborate while retaining separate search histories.

Existing systems coordinate plans, roles, and natural-language states (Su et al., 2025; Ghareeb et al., 2025; Lyu et al., 2026; Feng et al., 2026; Xin et al., 2026; Tang et al., 2025), leaving evidence-level transfer rules unspecified, whereas principle-aware methods (Pu et al., 2026; 2025) adapt scientific principles as a structured medium, maintaining one context-specific evidence stream per cycle. Naively extending this principle optimization to many branches invites failure: one branch's observation may duplicate an existing measurement, come from a different context, or need replication before its relevance is clear; principles then grow overconfident or noisy, and performance degrades. This creates a concrete requirement for existing systems: *parallel AI Scientists must share measurements without treating redundant or context-shifted evidence as independent local evidence*.

These limitations culminate in three challenges: (a) concurrent branches must provide **complementary coverage**, not repetition; (b) transfers must reflect **source compatibility**; and (c) **wall-clock efficiency**: any speedup must exceed the cost of screening.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To bridge this gap, we propose **CoEvolve**, treating collaborative scientific discovery as an evidence transfer problem via *principle co-evolution* (Definition 1.1), as shown in Figure 1. Each branch retains its own principle posterior (Pu et al., 2026). A *coordination core* collects branch-level evidence, deduplicates and discounts each record, and routes them to the target branch for principle posterior updates. This COLLECT–MERGE–SCORE–ROUTE–INJECT loop fosters a parallel exploration that maximizes the information gained from each branch based on information theory. Consequently, **CoEvolve** connects concurrent exploration to local updates while preserving branch autonomy.

Definition 1.1 (Principle co-evolution)**.** *In this paper, we use the term* principle co-evolution *to describe a parallel hypothesis-testing protocol in which $K \geq 2$ branches run independent principle-evolution from different sub-domains, over a shared Universal Principle Space (Pu et al., 2026). The branches are coupled exclusively through a shared pool of tested evidence. Evidence may update a target branch only as a likelihood factor (Section 3.2). Branches thus co-evolve through shared measurements only, no global posterior or belief is formed.*

We evaluate **CoEvolve** on six scientific-discovery tasks and five autoresearch tasks. Empirical results show that **CoEvolve**: **(a) attains the best average solution quality**, a mean of 66.5% versus 57.0% for PiEvo (Pu et al., 2026); **(b) converts parallel throughput into quality**, completing the same evaluation budget $1.80\times$ faster than PiEvo on the GPT-5.6-Terra backbone with a $+8.3$ equal-time solution quality gap at a $0.97\times$ per-token conversion rate; and **(c) generalizes to open-ended autoresearch**, standing as the only method above the published SOTA anchor on all five harness tasks (mean improvement $+16.7\%$ vs. PiEvo's $+5.8\%$), with the margin growing with task explorativeness: on the most explorative task, CoEvolve nearly triples the strongest baseline (Pu et al., 2026)'s improvement over the anchor ($+45.4\%$ vs. PiEvo's $+15.4\%$).

In summary, our contributions are:

- **(a)** We introduce the concept of **Principle Co-evolution**, transforming principle-evolving agents to parallel scientific discovery, widening exploration with high efficiency.
- **(b)** We propose **CoEvolve**, a framework that evolves scientific principles via evidence transfer in parallel, sharing branch-level evidence and speeding up discovery.
- **(c)** We conduct **extensive evaluations** across six closed-world scientific tasks and five autoresearch tasks, showing that CoEvolve attains the best average solution quality and is the only system above the published SOTA anchor on all autoresearch tasks, with the gains traced to disciplined evidence transfer rather than parallelism alone.

2 Related Work

Agentic AI for scientific discovery. Recent surveys describe a shift from workflows to *agentic science*, where LLMs plan studies, invoke domain tools, and iterate over hypotheses (Wei et al., 2025; Gridach et al., 2025; Herron et al., 2026). End-to-end systems instantiate this loop at increasing autonomy and horizon: AI Researcher and DeepScientist explore solution spaces, Co-Scientist organizes deliberative agents with experimental validation, and EvoScientist, EvoSci, and InternAgent-1.5 maintain evolving populations or long-horizon research state (Tang et al., 2025; Weng et al., 2025; Gottweis et al., 2025; Lyu et al., 2026; Xiong et al., 2026; Feng et al., 2026). Other systems redesign the loop through scientific-method controllers, stateful workflows, environment engineering, or outer-loop optimization (Smith, 2026; Zhao et al., 2026; Xin et al., 2026; Qu & Lu, 2026). Specifically, PiEvo (Pu et al., 2026) casts discovery as Bayesian optimization over an expanding set of principles; its single evidence stream, however, confines exploration to one evolution. CoEvolve lifts this confinement: evidence validated in one branch becomes evidence for all.

Language model-based collaboration. Multi-agent discovery systems distribute roles, plans, or reasoning across specialized LLMs: SciAgents (Ghafarollahi & Buehler, 2024) and Robin (Ghareeb et al., 2025) automate workflow-level coordination, Su et al. (2025) improves idea generation through agent plurality, and domain systems such as GenoMAS (Liu et al., 2025) and modular materials agents couple collaboration to code or task structure (Chaudhari et al., 2026). PiFlow (Pu et al., 2025) further frames search as principle-guided exploration-exploitation balance, while EDAgent addresses concurrent orchestration and incremental state synchronization (Yuan et al., 2026). These systems show that coordination improves discovery diversity, but communication remains prompt-level: which

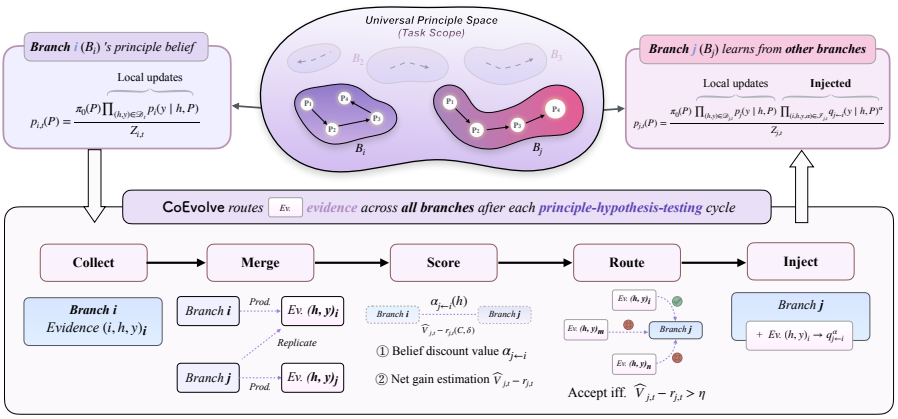


Figure 2: **Overview of CoEvolve.** After every hypothesis-testing cycle, the *coordination core* (bottom) collects hypothesis–outcome pairs as evidence, merges them into a global evidence pool, scores each source–target pair, and routes and injects only evidence satisfying the gate. An accepted record enters the target branch as the discounted likelihood factor $q_{j\leftarrow i}(y \mid h, P)^{\alpha}$; rejected records remain in the evidence pool.

records may cross agents, and at what weight, is left unspecified. CoEvolve leverages a record-level protocol: what crosses agents is a tested observation whose weight is computed.

Evidence transfer in parallel search. Classical parallel search provides analytical primitives for controlling information flow. Batch and asynchronous Bayesian optimization reduce redundant acquisitions and provide regret guarantees (Ren & Li, 2025; Sugiura et al., 2026); safe optimization constrains admissible actions (Wei et al., 2024); physics-informed composite models structure prediction targets (Dong et al., 2026); multi-output Gaussian processes characterize conditions for avoiding negative transfer (Li & Kontar, 2022); and island-model evolution uses controlled migration to preserve population diversity (Żychowski et al., 2025). In LLM-based search, DeltaEvolve records structured semantic deltas, but only along the successive nodes of one lineage (Jiang et al., 2026). These primitives control information flow but leave the experiment-level protocol open: an import may be dependent or require replication. CoEvolve supplies that protocol: observations cross branches only as gated, discounted imports, certified before they write into a posterior.

3 Methodology

3.1 Problem Setup and Notations

We formulate collaborative scientific discovery as a *principle co-evolution* problem (Definition 1.1). Each branch j explores a hypothesis space $\mathcal{H}_j$ and maintains a posterior over a shared finite principle universe $\bar{\mathcal{P}}$, where a principle P models how hypotheses produce outcomes and $P^\star$ denotes the true principle (Pu et al., 2026). At round t, branch j selects $h \in \mathcal{H}_j$, evaluates it with the task validator, and observes $y \sim p_j(\cdot \mid h, P^\star)$; the tested pairs (h, y) form the local evidence set $\mathcal{D}_{j,t}$.

CoEvolve leaves the branch-level loop unchanged and adds a *coordination core*. As shown in Figure 2, each branch owns its tested pairs, publishes them as candidate records, and may admit other branches' records as imports; Definition 3.1 delimits this *evidence-only* interface, and Section 3.2 formulates these rules.

Definition 3.1 (Evidence objects and system history)**.** *A* local observation *is a pair* $(h, y) \in \mathcal{D}_{j,t}$ *tested by branch* j *itself. To share one, source branch* i *publishes a* candidate record $C = (i, h, y, E_i, m)$ *carrying source context* E_i *and metadata* m*. An* import *for target branch* j *is a record that passes* j*'s routing decision, with the discount* $\alpha_{j\leftarrow i,t}(h)$ *attached at scoring time; those accepted by round* t *form* $\mathcal{I}_{j,t}$*. Let* H_t^{sys} *denote the branch histories and evidence pool available at round* t*; discounts and routing decisions use only the history available before an import is injected.*

3.2 Theoretical Framework

In CoEvolve, a target-specific router in the coordination core decides whether and with what weight a pooled record may update one branch. An accepted record enters the target posterior as the

discounted tuple (i, h, y, α). Below, we derive in turn how a record is *scored*, *routed*, and *injected* into the target posterior.

Score: information value of a candidate record. From first principles, the synergy between branches resides in their differences: *an observation is worth transferring to branch j precisely to the extent that it carries information branch j does not already have*. The natural measure of this worth is the entropy reduction its update would induce in the target's principle posterior (Pu et al., 2026), net of the transfer cost. Let $H(p) = -\sum_{P\in\bar{\mathcal{P}}} p(P)\log p(P)$ and let $p^{+}_{j,t}$ denote the posterior after the update for C. For an already observed candidate record C, with the transfer cost $\Delta^{\mathrm{imp}}_{j,t}(C)$ measured on the same normalized entropy scale, its conditional (post-outcome) net value is

$$V_{j,t}(C) = \mathbb{E}\big[H(p_{j,t}) - H(p^{+}_{j,t}) \mid H^{\mathrm{sys}}_{t}, C\big] - \lambda\,\Delta^{\mathrm{imp}}_{j,t}(C). \tag{1}$$

The first term prices the information C carries about the governing principle; the second charges the transfer at rate λ, so a record must be expected to pay for its own transfer. Only the expectation is certified: a realized import can still raise the target's entropy.

Route: certified admission and ranking. As Eq. (1) makes explicit, routing is scored after the source record, outcome included, is published; the surviving uncertainty is the verification and fitting randomness conditional on the observed C, not the outcome itself (a pre-outcome variant would additionally integrate over the predictive distribution of y). Admission is therefore decided from a computable estimate $\widehat{V}_{j,t}(C)$, and the estimation error must be accounted for explicitly. We require the estimation to come with a calibrated radius $r_{j,t}(C,\delta)$ such that $\Pr\Big(\big|\widehat{V}_{j,t}(C) - V_{j,t}(C)\big| \le r_{j,t}(C,\delta) \mid H^{\mathrm{sys}}_{t}, C\Big) \ge 1-\delta$. Admission then uses a lower-confidence rule with a margin $\eta > 0$ and a per-candidate confidence budget $\delta_{j,t,C}$: the router accepts a candidate only if

$$\widehat{V}_{j,t}(C) - r_{j,t}(C, \delta_{j,t,C}) > \eta, \tag{2}$$

so a candidate that merely looks valuable on average, but whose value is uncertain, is refused. Among the candidates that pass, the router ranks by

$$C^{\star}_{j,t} = \operatorname*{arg\,min}_{C\in\mathrm{passed}_j} \Delta^{\mathrm{imp}}_{j,t}(C)^2 \big/ \max\{\widehat{V}_{j,t}(C) - r_{j,t}(C,\delta_{j,t,C}), 0\} + \varepsilon, \tag{3}$$

which prefers candidates offering high certified value per unit transfer cost. Admission is further capped per round and target branch by a routing quota B (default 3), keeping the top-B candidates in ranked order, and by a redundancy cap $R_{\max}$ that drops a candidate whose text is a near-duplicate (token-Jaccard similarity $\ge R_{\max}$, default 0.3) of one already admitted in the same round; both are specified in Appendix D. The gate and the ratio decide *which* records may flow, and in what order; *how* an accepted record enters the target posterior is specified by the injection step that follows.

Inject: discounted likelihood update. We assume that branch contexts may be heterogeneous. A cross-branch observation is therefore valid for target branch j only through an explicit likelihood $q_{j\leftarrow i}(y \mid h, P)$; to keep this transfer disciplined, we introduce a source–target discount $\alpha_{j\leftarrow i,t}(h) \in [0,1]$ that downweights the contribution of a source observation to a target posterior, formulated as:

$$\alpha_{j\leftarrow i,t}(h) = \mathrm{clip}_{[0,1]}(\rho_{i,t}(h) s_{ij,t}(h) v_{i,t}(h))\,, \tag{4}$$

where $\rho_{i,t}$ scores the source's accuracy in predicting its own outcomes, $s_{ij,t}$ scores how well the source's setting matches the target's, and the optional $v_{i,t}$ scores whether the observation has been independently replicated. Therefore, branch j's posterior after local evidence and accepted imports is

$$p_{j,t}(P) = \pi_0(P) \prod_{(h,y)\in\mathcal{D}_{j,t}} \underbrace{p_j(y \mid h, P)}_{\text{Local updates}} \prod_{(i,h,y,\alpha)\in\mathcal{I}_{j,t}} \underbrace{q_{j\leftarrow i}(y \mid h, P)^{\alpha}}_{\text{Evidence transfer}} \big/ Z_{j,t}, \tag{5}$$

where $\pi_0(P) > 0$ is the prior, $\mathcal{D}_{j,t}$ contains locally tested observations, $\mathcal{I}_{j,t}$ contains target-specific accepted imports, and $Z_{j,t}$ is the normalizer. When the observations are conditionally independent and each $q_{j\leftarrow i}$ is the correct law for an observation generated in source context i and interpreted by target j, this is an ordinary posterior; otherwise it is a generalized-Bayes or composite-likelihood posterior, and an import must not be read as a full independent sample.

Each injection multiplies the target posterior by exactly one factor of Eq. (5). The three concerns thus stay separate by construction: the shared pool governs system-level coverage, the discounted posterior governs branch-level concentration, and the calibrated gate governs negative-transfer control.

Instantiation of CoEvolve. Every quantity above is computed from logged statistics and record metadata alone. The value estimate $\widehat{V}_{j,t}(C)$ is evaluated exactly: clone the target posterior, apply the candidate at weight α, and measure the induced entropy change. This is tractable because the principle universe is finite. The transfer cost $\Delta^{\mathrm{imp}}_{j,t}(C)$ combines fixed bookkeeping rates with a surcharge that grows with the same setting mismatch that downweights α, so a less compatible source is discounted and charged more at once. Each accepted import takes effect only once its acceptance is recorded, so no posterior update is silent. Exact enumeration certifies the numerical update ($r = 0$) but not Gaussian Process misspecification or residual context shift (refer to the $p_j(y \mid h, P)$ modeling in PiEvo (Pu et al., 2026)); the gate therefore remains a conservative heuristic rather than a deployed guarantee (Remark B.4). Exact forms and the per-round loop are given in Appendix D.

Theorem 3.2 (Concentration and negative-transfer control, informal)**.** *Under a finite principle universe with positive prior (Appendix A),* CoEvolve *guarantees: (a)* ***concentration under discounted imports****: if the discounted log-likelihood contrast in favor of $P^\star$ grows linearly in the effective evidence count $N^{\mathrm{eff}}_{j,T}$, i.e., local observations entering at unit weight, imports at their discount α, then each branch's belief $p_{j,T}(P^\star)$ converges to one exponentially fast in $N^{\mathrm{eff}}_{j,T}$ (Theorem A.2); and (b)* ***negative-transfer control****: if the value estimates are calibrated, the gate of Eq. (2) bounds the probability of accepting any import whose net value falls below the margin η by the calibration budget $\sum_{j,t,C} \delta_{j,t,C}$ (Theorem A.3).*

In summary, we detail the implementation of CoEvolve in Algorithm 1, and CoEvolve provides theoretical guarantees (Theorem 3.2, formal statements are in Appendix A, with proofs in Appendix B) and empirical evidence, as validated in Sections 4.4 and 4.6 (Figures 4 and 7).

Remark 3.1 (Evidence-level privacy scope)**.** CoEvolve *coordinates branches through* scientific evidence*: it keeps each branch's posterior and search history private by construction. For the sensitive evidence, privacy-preserving variants of the shared pool compose with the routing and gating machinery unchanged, and we leave them as future work.*

4 Experiments

4.1 Experiment Setup

Benchmarks. We evaluate CoEvolve on six scientific tasks, spanning Molecular Bio-activity Optimization (MBO) (Gaulton et al., 2011), Antimicrobial Peptide Design (AMP) (Santos-Junior et al., 2020), Promoter Expression Optimization (Promoter) (de Almeida et al., 2022) in biology, Nanohelix Optimization (NHO) (Wu et al., 2025) and Transition-Metal-Complex Design (TMC) (Song et al., 2025) in materials science, and Superconductor Critical-Temperature Optimization (SPO) (Hamidieh, 2018) in physics. Each task utilizes surrogates (Pu et al., 2026) and released databases, and gates candidates by the standard validity rules of the field. Details refer to Appendix I.

Baselines. We compare CoEvolve against three families of baselines. (i) *Vanilla MAS*: a plain multi-agent system without any hypothesis-selection or principle-evolution mechanism, which anchors the per-task reference. (ii) *Autonomous research agents*: The AI Scientist v1 (Lu et al., 2024), The AI Scientist v2 (Yamada et al., 2025), AI Researcher (Tang et al., 2025), EvoScientist (Lyu et al., 2026), and InternAgent-1.5 (Feng et al., 2026); to ensure a fair comparison on hypothesis discovery, each system is restricted to its core hypothesis-generation loop (no literature retrieval or report drafting). (iii) *Principle-guided search*: PiFlow (Pu et al., 2025), which performs principle-aware exploration-exploitation balance, and PiEvo (Pu et al., 2026), which evolves its principle space.

Implementation details. We use Gemma-4-31B-IT (Gemma Team et al., 2026) and GPT-5.6-Terra (OpenAI, 2026a) for evaluation, with three independent seeds, under a total evaluation budget of 72 queries without any web or retrieval access; CoEvolve runs $K{=}3$ branches by default (24 queries per branch). Per-task wall-clock time is measured from launcher start to final evaluation on identical shared hardware, using the same LLM endpoint.

4.2 Evaluation Metrics

Following Pu et al. (2026), our evaluation of CoEvolve against baselines focuses on (a) objective attainment, (b) exploration breadth, and (c) per-evaluation anytime efficiency. The elapsed-time and

Table 1: **Performance comparison with baselines on 6 benchmarks under the Solution Quality (SQ %) metric.** Categorized by LLMs: `Gemma-4-31B-IT` and `GPT-5.6-Terra` with both non-thinking mode. We report mean ± std over three seeds. On the GPT-5.6-Terra backbone, CoEvolve's paired per-seed SQ gain over PiEvo is positive in 15 of 18 task–seed pairs (two-sided sign test $p<0.01$; Wilcoxon $p=0.001$; full tests in Appendix G). **Bold** marks the best mean per task within each backbone block and the best Average. The colored ratio after each cell highlights superior (↑) or inferior (↓) SQ relative to the worst non-zero mean on the same backbone and task, which is underlined.

Model / Method	MBO	NHO	SPO	TMC	AMP	Promoter	Average
Gemma-4-31B-IT							
Vanilla MAS	39.2 ± 0.8	77.2 ± 1.4	0.0 ± 0.0 ↓0.0x	74.4 ± 0.0 ↑1.3x	27.4 ± 2.3 ↑1.1x	84.4 ± 0.3 ↑4.5x	50.4 ↑1.2x
The AI Scientist v1 (Lu et al., 2024)	48.6 ± 2.2 ↑1.2x	78.3 ± 4.2 ↑1.0x	0.0 ± 0.0 ↓0.0x	84.5 ± 0.0 ↑1.5x	24.1 ± 1.5	18.7 ± 32.4	42.4 ↑1.0x
The AI Scientist v2 (Yamada et al., 2025)	48.5 ± 5.1 ↑1.2x	84.3 ± 3.9 ↑1.1x	8.2 ± 1.6	83.7 ± 0.8 ↑1.5x	27.7 ± 3.0 ↑1.2x	0.0 ± 0.0 ↓0.0x	42.1
AI Researcher (Tang et al., 2025)	50.0 ± 2.4 ↑1.3x	88.3 ± 1.8 ↑1.1x	24.6 ± 2.0 ↑3.0x	81.5 ± 6.1 ↑1.5x	26.7 ± 14.6 ↑1.1x	19.3 ± 33.5 ↑1.0x	48.4 ↑1.2x
EvoScientist (Lyu et al., 2026)	45.3 ± 1.5 ↑1.2x	93.3 ± 0.7 ↑1.2x	9.2 ± 0.7 ↑1.1x	85.4 ± 0.0 ↑1.5x	29.7 ± 2.6 ↑1.2x	54.0 ± 7.8 ↑2.9x	52.8 ↑1.3x
InternAgent 1.5 (Feng et al., 2026)	53.6 ± 2.1 ↑1.4x	90.5 ± 5.5 ↑1.2x	10.5 ± 0.1 ↑1.3x	76.8 ± 1.6 ↑1.4x	28.7 ± 1.7 ↑1.2x	86.2 ± 0.6 ↑4.6x	57.7 ↑1.4x
PiFlow (Pu et al., 2025)	64.5 ± 5.7 ↑1.6x	91.0 ± 3.7 ↑1.2x	15.7 ± 7.9 ↑1.9x	56.0 ± 48.5	32.7 ± 4.5 ↑1.4x	72.2 ± 9.1 ↑3.9x	55.3 ↑1.3x
PiEvo (Pu et al., 2026)	61.8 ± 2.7 ↑1.6x	**94.9** ± 0.1 ↑1.2x	10.6 ± 0.0 ↑1.3x	84.7 ± 0.0 ↑1.5x	31.4 ± 5.8 ↑1.3x	54.4 ± 6.3 ↑2.9x	56.3 ↑1.3x
CoEvolve (ours)	**77.1** ± 4.6 ↑2.0x	**94.9** ± 0.1 ↑1.2x	**30.0** ± 0.6 ↑3.7x	**88.3** ± 4.2 ↑1.6x	**37.6** ± 2.6 ↑1.6x	**86.4** ± 0.4 ↑4.6x	**69.1** ↑1.6x
GPT-5.6-Terra							
Vanilla MAS	51.0 ± 2.6	73.2 ± 2.6	0.0 ± 0.0 ↓0.0x	66.6 ± 1.7	22.1 ± 3.2 ↑1.0x	70.8 ± 14.3 ↑1.1x	47.3
The AI Scientist v1 (Lu et al., 2024)	60.0 ± 6.5 ↑1.2x	89.8 ± 4.7 ↑1.2x	0.0 ± 0.0 ↓0.0x	85.4 ± 0.0 ↑1.3x	23.8 ± 2.6 ↑1.1x	62.1 ± 16.7	53.5 ↑1.1x
The AI Scientist v2 (Yamada et al., 2025)	51.5 ± 5.8 ↑1.0x	84.9 ± 9.7 ↑1.2x	10.8 ± 0.3 ↑1.1x	83.6 ± 3.0 ↑1.3x	26.7 ± 0.0 ↑1.2x	76.4 ± 15.6 ↑1.2x	55.6 ↑1.2x
AI Researcher (Tang et al., 2025)	61.2 ± 5.7 ↑1.2x	93.6 ± 2.5 ↑1.3x	11.8 ± 0.1 ↑1.2x	89.3 ± 3.4 ↑1.3x	23.4 ± 2.5 ↑1.1x	73.6 ± 8.3 ↑1.2x	58.8 ↑1.2x
EvoScientist (Lyu et al., 2026)	53.7 ± 5.6 ↑1.1x	86.8 ± 6.6 ↑1.2x	10.2 ± 0.2	78.5 ± 6.0 ↑1.2x	21.5 ± 2.9	72.6 ± 11.7 ↑1.2x	53.9 ↑1.1x
InternAgent 1.5 (Feng et al., 2026)	61.0 ± 6.0 ↑1.2x	91.8 ± 3.3 ↑1.3x	11.4 ± 0.1 ↑1.1x	91.8 ± 2.2 ↑1.4x	24.1 ± 2.1 ↑1.1x	74.1 ± 10.7 ↑1.2x	59.0 ↑1.2x
PiFlow (Pu et al., 2025)	52.3 ± 14.6 ↑1.0x	76.3 ± 5.9 ↑1.0x	15.0 ± 7.6 ↑1.5x	77.6 ± 2.9 ↑1.2x	22.8 ± 1.7 ↑1.1x	76.3 ± 8.2 ↑1.2x	53.4 ↑1.1x
PiEvo (Pu et al., 2026)	54.3 ± 1.7 ↑1.1x	86.2 ± 1.4 ↑1.2x	10.4 ± 0.1 ↑1.0x	91.5 ± 1.3 ↑1.4x	23.8 ± 0.0 ↑1.1x	79.4 ± 5.8 ↑1.3x	57.6 ↑1.2x
CoEvolve (ours)	**71.3** ± 19.1 ↑1.4x	**93.9** ± 0.7 ↑1.3x	**15.3** ± 8.3 ↑1.5x	**93.0** ± 0.0 ↑1.4x	**27.7** ± 1.7 ↑1.3x	**82.1** ± 3.0 ↑1.3x	**63.9** ↑1.4x

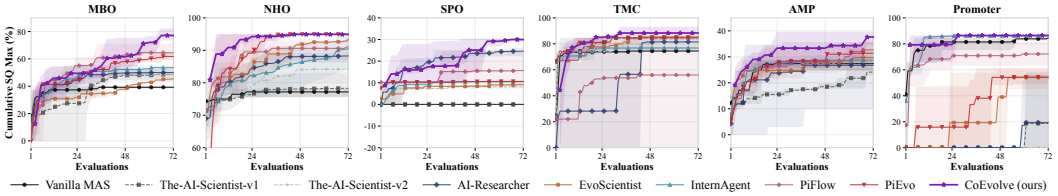


Figure 3: **Running-best SQ w/ Gemma-4-31B-IT.** We plot the cumulative best SQ over evaluations for each task per Table 1. CoEvolve attains the highest final running best on five of the six tasks and ties PiEvo (Pu et al., 2026) on NHO (94.9%).

cost metrics (wall-clock speedup, the equal-time quality gap $\Delta\text{SQ@}T_c$, the token conversion ratio, and token-normalized AUOC) are defined in Section 4.4, where they are used.

a. Solution Quality (SQ). $\text{SQ} = {}^{\max\{y_k \mid (h_k,y_k)\in\mathcal{T}\}-y_{\text{lo}}}/_{y_{\text{hi}}-y_{\text{lo}}} \times 100\%$ is the maximum outcome along a trajectory $\mathcal{T}$, normalized to the domain reference scale $[y_{\text{lo}}, y_{\text{hi}}]$ (see Appendix I). Here 100% marks the task's domain reference scale: a definitional bound on four tasks and a domain-anchored target on MBO and SPO, never approached in our runs (Appendix I); this keeps the cross-task average an arithmetic mean of comparable quantities. For a multi-branch system such as CoEvolve, $\mathcal{T}$ is the union of the run's branch trajectories (best-of-seed), and APD and AUOC below are computed over the same pooled trajectory, so every metric prices the run as a whole.

b. Average Pairwise Distance (APD). $\text{APD} = {}^{2}/_{M(M-1)} \sum_{1\le i<j\le M} d\left(\phi(h_i), \phi(h_j)\right)$ is the mean pairwise distance among the M valid hypotheses under the task-specific feature map $\phi(h)$ (see Appendix I). Higher APD indicates a wider spread of exploration.

c. Area Under the Optimization Curve (AUOC). $\text{AUOC} = {}^{1}/_{T} \sum_{t=1}^{T} \frac{\max_{k\le t}\{y_k\}-y_{\text{lo}}}{y_{\text{hi}}-y_{\text{lo}}}$. measures the cumulative running-best performance over the fixed evaluation budget. Higher AUOC indicates that a method reaches strong running-best values earlier in the evaluation sequence.

4.3 Performance Analysis

Table 1 compares CoEvolve against eight baselines on six benchmarks under two backbones, Figure 3 traces the running-best solution quality (SQ) over the evaluation sequence, and Table 2 profiles exploration (APD) against exploitation (AUOC). Three observations follow from these tables:

Obs.❶: CoEvolve attains the best average SQ on these six benchmarks. CoEvolve attains the highest average SQ on both backbones, 63.9% ∼ 69.1%. This corresponds to an 8% ∼ 20% relative gain over the strongest baseline, InternAgent-1.5 (Feng et al., 2026), and 11% ∼ 23% over

Table 2: **Exploration (APD ↑) and exploitation (AUOC ↑) comparison with baselines on 6 benchmarks.** Both over the same three seeds as Table 1. **Bold** marks the best mean per column within each backbone and underline the second-best; the colored ratio after each Average highlights superior (↑) or inferior (↓) performance relative to Vanilla on the same backbone. CoEvolve attains the best exploration–exploitation balance.

Model / Method	MBO		NHO		SPO		TMC		AMP		Promoter		Average	
	APD ↑	AUOC ↑	APD ↑	AUOC ↑	APD ↑	AUOC ↑	APD ↑	AUOC ↑	APD ↑	AUOC ↑	APD ↑	AUOC ↑	Avg APD ↑	Avg AUOC ↑
Gemma-4-31B-IT														
Vanilla MAS	**79.1** ± 1.1	37.5 ± 0.9	24.0 ± 1.1	76.5 ± 1.0	0.0 ± 0.0	0.0 ± 0.0	29.8 ± 4.8	73.7 ± 0.5	76.1 ± 0.9	25.7 ± 2.6	36.9 ± 1.1	80.2 ± 4.4	41.0	48.9
The AI Scientist v1 (Lu et al., 2024)	21.7 ± 17.0	38.2 ± 3.4	5.9 ± 3.4	76.8 ± 3.6	0.0 ± 0.0	0.0 ± 0.0	53.1 ± 1.1	81.5 ± 1.0	58.5 ± 3.0	17.2 ± 1.1	0.0 ± 0.0	18.7 ± 32.4	23.2 ↓0.6x	38.7 ↓0.8x
The AI Scientist v2 (Yamada et al., 2025)	34.7 ± 12.5	43.6 ± 3.5	12.9 ± 2.2	80.5 ± 3.0	4.6 ± 2.9	7.7 ± 1.0	47.4 ± 4.2	78.5 ± 0.7	34.3 ± 6.2	25.7 ± 3.5	0.0 ± 0.0	0.0 ± 0.0	22.3 ↓0.5x	39.3 ↓0.8x
AI Researcher (Tang et al., 2025)	46.5 ± 9.0	46.1 ± 2.1	16.6 ± 4.8	84.5 ± 1.4	4.7 ± 0.1	**21.2** ± 1.9	52.3 ± 5.3	81.1 ± 5.9	73.1 ± 8.7	23.5 ± 14.8	18.3 ± 31.7	18.3 ± 31.7	35.3 ↓0.9x	45.8 ↓0.9x
EvoScientist (Lyu et al., 2026)	57.0 ± 11.4	35.7 ± 1.7	23.3 ± 1.5	87.3 ± 2.0	8.3 ± 0.9	8.3 ± 0.8	61.8 ± 0.7	79.5 ± 0.5	60.7 ± 2.2	25.8 ± 2.4	0.0 ± 0.0	54.0 ± 7.8	35.2 ↓0.9x	48.4 ↓1.0x
InternAgent 1.5 (Feng et al., 2026)	55.3 ± 5.0	47.3 ± 5.0	14.3 ± 3.4	83.3 ± 4.6	4.7 ± 4.9	9.9 ± 0.6	51.5 ± 2.3	73.4 ± 2.4	51.8 ± 6.3	26.3 ± 1.4	30.6 ± 1.2	**85.4** ± 0.5	34.7 ↓0.8x	54.3 ↑1.1x
PiFlow (Pu et al., 2025)	78.9 ± 3.6	**56.2** ± 5.2	34.3 ± 5.7	87.5 ± 3.1	**15.4** ± 0.7	13.3 ± 4.7	45.0 ± 16.0	52.0 ± 45.2	**78.1** ± 1.3	26.1 ± 0.6	42.9 ± 2.2	64.9 ± 4.7	49.1 ↑1.2x	50.0 ↑1.0x
PiEvo (Pu et al., 2026)	77.3 ± 5.1	50.1 ± 2.7	27.3 ± 3.8	86.7 ± 6.3	6.1 ± 0.2	10.3 ± 0.1	67.2 ± 2.4	**82.3** ± 2.0	76.2 ± 4.8	26.8 ± 3.1	**46.9** ± 7.5	33.1 ± 16.6	50.2 ↑1.2x	48.2 ↓1.0x
CoEvolve (ours)	73.9 ± 7.6	54.7 ± 6.5	**39.1** ± 0.1	**92.7** ± 1.7	13.3 ± 2.2	20.7 ± 6.9	**68.1** ± 5.0	80.0 ± 10.6	75.6 ± 1.7	**31.7** ± 4.9	42.2 ± 9.5	83.6 ± 4.6	**52.0** ↑1.3x	**60.6** ↑1.2x
GPT-5.6-Terra														
Vanilla MAS	57.4 ± 6.0	47.7 ± 1.8	22.0 ± 1.9	70.7 ± 4.8	0.0 ± 0.0	0.0 ± 0.0	49.2 ± 2.4	65.0 ± 1.4	83.8 ± 0.2	20.4 ± 1.5	33.6 ± 0.6	63.2 ± 13.8	41.0	44.5
The AI Scientist v1 (Lu et al., 2024)	6.5 ± 5.6	54.6 ± 4.5	13.7 ± 1.4	80.8 ± 12.6	0.0 ± 0.0	0.0 ± 0.0	56.3 ± 2.3	75.9 ± 6.5	38.3 ± 11.0	20.3 ± 1.4	19.1 ± 4.2	53.9 ± 9.0	22.3 ↓0.5x	47.6 ↑1.1x
The AI Scientist v2 (Yamada et al., 2025)	28.1 ± 9.3	45.3 ± 3.3	15.6 ± 4.3	72.4 ± 9.9	1.5 ± 0.3	10.5 ± 0.1	57.3 ± 2.3	75.3 ± 2.6	33.3 ± 3.9	**24.2** ± 0.4	29.6 ± 2.7	72.0 ± 15.3	27.6 ↓0.7x	49.9 ↑1.1x
AI Researcher (Tang et al., 2025)	74.8 ± 1.6	44.6 ± 6.9	23.1 ± 5.3	88.6 ± 5.7	6.6 ± 2.3	10.8 ± 0.3	57.1 ± 1.2	82.8 ± 1.4	87.3 ± 1.6	20.1 ± 3.0	33.7 ± 1.4	63.1 ± 5.0	47.1 ↑1.1x	51.7 ↑1.2x
EvoScientist (Lyu et al., 2026)	36.9 ± 7.7	46.8 ± 2.7	28.4 ± 4.8	73.6 ± 4.8	16.4 ± 1.6	9.7 ± 0.3	60.6 ± 2.6	75.2 ± 4.0	84.8 ± 3.9	19.5 ± 3.5	35.3 ± 5.4	65.7 ± 15.1	43.7 ↑1.1x	48.4 ↑1.1x
InternAgent 1.5 (Feng et al., 2026)	35.3 ± 23.7	**55.2** ± 6.6	26.4 ± 11.9	85.5 ± 7.2	1.0 ± 0.2	10.8 ± 0.1	60.6 ± 0.6	80.3 ± 1.9	58.1 ± 7.2	22.3 ± 1.7	29.7 ± 4.1	70.3 ± 11.9	35.2 ↓0.9x	54.1 ↑1.2x
PiFlow (Pu et al., 2025)	77.4 ± 7.5	45.2 ± 10.6	48.5 ± 2.3	64.4 ± 9.3	**16.5** ± 0.2	**12.3** ± 4.4	**79.4** ± 3.4	74.5 ± 0.7	86.9 ± 1.4	19.2 ± 1.8	37.0 ± 1.1	68.5 ± 4.3	57.6 ↑1.4x	47.3 ↑1.1x
PiEvo (Pu et al., 2026)	70.1 ± 11.4	41.8 ± 5.7	30.3 ± 3.0	76.1 ± 4.2	11.6 ± 1.5	9.3 ± 0.9	68.0 ± 2.4	**89.6** ± 2.8	84.8 ± 1.0	22.0 ± 0.9	40.5 ± 1.9	**73.7** ± 3.5	50.9 ↑1.2x	52.1 ↑1.2x
CoEvolve (ours)	**79.8** ± 4.4	45.5 ± 8.8	**54.7** ± 1.6	**88.8** ± 5.9	9.4 ± 6.7	11.5 ± 2.5	75.5 ± 0.7	87.3 ± 3.8	**87.5** ± 3.1	23.6 ± 4.7	**41.9** ± 1.0	70.8 ± 7.6	**58.1** ↑1.4x	**54.6** ↑1.2x

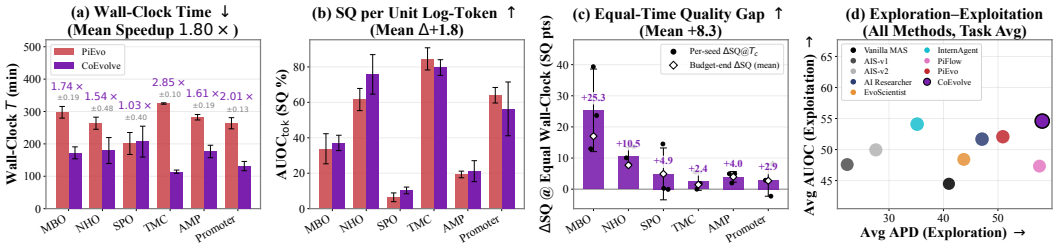


Figure 4: **Efficiency analysis on the GPT-5.6-Terra backbone (Table 1 & 2)** (a) End-to-end wall-clock time per task, with the paired speedup $T_{\text{PiEvo}}/T_{\text{CoEvolve}}$ annotated; (b) Token-normalized AUOC per task (AUOC_{tok}); (c) Equal-time quality gap $\Delta\text{SQ@}T_c$: PiEvo's running-best SQ read at CoEvolve's completion time T_c, subtracted from CoEvolve's final SQ; (d) CoEvolve is Pareto-optimal.

PiEvo (Pu et al., 2026) (paired per-seed gains positive on 15 of 18 task–seed pairs, $p<0.01$; Table 1), indicating that evidence transfer directs parallel exploration toward high-quality hypotheses.

Obs.❷: CoEvolve's lead holds throughout evaluations on average. In Figure 3, CoEvolve's running-best curve attains the highest final value on five of six tasks. CoEvolve achieves the best average AUOC on both backbones (Table 2). Strong candidates therefore arrive early and the advantage persists, which is what Section 4.4 prices in wall-clock time and tokens.

Obs.❸: CoEvolve is the unique Pareto-optimal method. CoEvolve attains the best average APD on both backbones together with the best average AUOC (Table 2). Each baseline, for example, InternAgent-1.5 (Feng et al., 2026) holds only the second-best average AUOC and explores 33% ∼ 39% more narrowly, while PiFlow nearly matches CoEvolve's exploration under GPT-5.6-Terra yet falls 7.3 AUOC points behind. This is consistent with the role of coordination core (Section 3.2): accepted cross-branch imports keep diversity while each branch's running best keeps improving.

4.4 Efficiency Analysis

As CoEvolve is built upon PiEvo (Pu et al., 2026), to measure wall-clock and token efficiency, we introduce three complementary measures (**Metric** 1 ∼ 3), then report two observations.

Metric 1. Wall-clock speedup and equal-time quality gap. We record the end-to-end wall-clock time T under the same device environment and report the paired speedup $T_{\text{PiEvo}}/T_{\text{CoEvolve}}$. We also compare the two systems after identical wall-clock investment: the *equal-time quality gap* $\Delta\text{SQ@}T_c = \text{SQ}_{\text{CoEvolve}}(T_c) - \text{SQ}_{\text{PiEvo}}(T_c)$ reads PiEvo's running best at CoEvolve's completion time T_c, using the per-evaluation timestamps of both.

Metric 2. Token conversion ratio $(\text{SQ}_{\text{CoEvolve}}/\text{SQ}_{\text{PiEvo}}) \,/\, (\text{Tok}_{\text{CoEvolve}}/\text{Tok}_{\text{PiEvo}})$**.** A value above 1 means each CoEvolve token produces more final SQ than a PiEvo token.

Metric 3. Token-normalized area under the optimization curve (AUOC_{tok}). AUOC measures efficiency per *evaluation*; AUOC_{tok} instead integrates the running-best curve over log-token-spend, normalized by the observed log-token window: $\text{AUOC}_{\text{tok}} =$

$1/\log \tau_N - \log \tau_1 \int_{\tau_1}^{\tau_N} \max_{k \le \tau}\{y_k\} - y_{\text{lo}} / y_{\text{hi}} - y_{\text{lo}} \, \mathrm{d} \log \tau$, where τ denotes cumulative token spend. We approximate the integral by the trapezoidal rule on the observed evaluations.

Obs.❹: Parallel advantage of CoEvolve costs a near-parity token premium. Branch-summed, CoEvolve uses $1.06\times \sim 1.62\times$ PiEvo's tokens per task. In return, the conversion ratio ranges $0.89\times \sim 1.07\times$ per task (mean $0.97\times$, above 1 on MBO and NHO), and token-normalized AUOC (Figure 4 (b)) favors CoEvolve on four of six tasks (mean $\Delta{+}1.8$). Together with Obs.❺: a $1.80\times$ wall-clock speedup (faster in 17 of 18 task–seed pairs, $p<10^{-3}$) at a $1.22\times$ token premium and a $0.97\times$ per-token rate means the parallel branches convert wall-clock into SQ.

4.5 Case Study on Autonomous Research Harness

Task configuration. Per our goal in Section 1 and following Fei et al. (2026), we port CoEvolve onto an autonomous research harness: a containerized coding agent must deliver an *artifact* that a sealed evaluator scores against each task's published SOTA anchor. We sample five representative tasks from AutoresearchEval (Fei et al., 2026): ALDE, D2D, and MolEdit from the *exploiting* category, and Deconv and InvScat from the *exploration* category (the exploiting category spans the free-oracle and recipe-saturated levels of the pre-registered exploitativeness index; Appendix H.5), and report anchor-relative improvement Δ under the same 72-evaluation budget as Section 4.1. Task definitions, the substrate, the budget contract, and all controls are in Appendix H.

We compare three types of harness: (a) *agent-only* harnesses (Claude Code (Anthropic, 2026), Codex (OpenAI, 2026b), Arbor (Jin et al., 2026)) act autonomously once given the task, (b) PiEvo (Pu et al., 2026) as a strategic layer inside Claude Code, injecting guidance per turn, and (c) CoEvolve runs $K{=}3$ guided PiEvo branches in parallel. With the same LLM backbone of `DeepSeek-v4-Flash-0731`, our analysis yields three observations:

Table 3: **Performance comparison on five autoresearch tasks (anchor-relative improvement $\Delta \times 100$, ↑).** Mean $\pm$ std over three seeds; **bold** = best per column. All five CoEvolve means are significantly above the anchor (one-sample t against zero, $p<0.025$ per task; Appendix G).

Arm	ALDE	Deconv	InvScat	D2D	MolEdit	Average
Claude Code	+2.3 ± 0.2	+14.0 ± 7.3	−5.8 ± 7.6	+9.0 ± 0.3	+0.2 ± 1.7	+3.9
Codex	+2.2 ± 1.0	+14.3 ± 5.4	−5.9 ± 6.7	+11.2 ± 0.9	+1.2 ± 1.9	+4.6
Arbor	+1.3 ± 1.9	+11.7 ± 23.8	−6.5 ± 5.9	+15.3 ± 0.1	+0.5 ± 0.6	+4.5
PiEvo	+2.4 ± 2.0	+15.4 ± 28.6	−5.3 ± 7.9	+15.4 ± 0.1	+1.3 ± 1.6	+5.8
CoEvolve	**+6.6** ± 0.7	**+45.4** ± 1.4	**+11.9** ± 3.7	**+15.5** ± 0.0	**+4.1** ± 0.6	**+16.7**

Obs.❻: CoEvolve surpasses all SOTA anchors (Fei et al., 2026). CoEvolve alone keeps all five means above the anchor (one-sample t, $p<0.025$ per task), lifting the cross-task average from PiEvo's $+5.8$ to $+16.7$ (see Table 3).

The two margins decompose as follows (Table 3): stage guidance (i.e., PiEvo) adds only $+0.4$ percentage points, while the coordination core (i.e., CoEvolve) adds $+10.9$ on top; the worst CoEvolve case (mean minus std) also matches or exceeds the best case (mean plus std) of every other arm on every task.

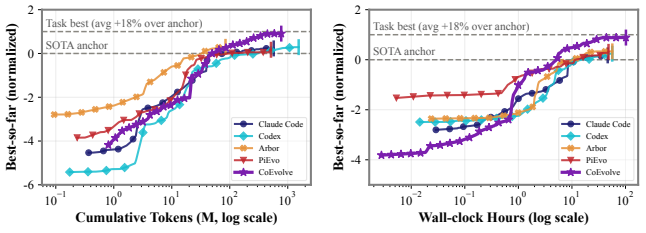


Figure 5: CoEvolve leads the token-spend curve from ∼1M tokens onward; the wall-clock view prices its parallel throughput.

Obs.❼: CoEvolve converts breadth into top-tier quality. PiEvo leads mid-budget on InvScat and D2D (and edges out CoEvolve on MolEdit in full-budget AUOC; Appendix H.4), yet CoEvolve gains complementary coverage at reduced single-line depth, and accepted imports enter the champion branch's posterior, at a $1.55\times$ mean billable-token spend (Figure 5, left; Table 10) for a $1.31\times$ end-to-end wall-clock speedup (Table 11).

Obs.❽: The payoff is bounded by what parallel coverage can find. The runner-up margin reaches $+17.2 \sim +30.0$ points on the open method-space tasks (Deconv, InvScat), against $+0.1 \sim +4.2$ where a saturated recipe or a free oracle governs the score, matching the pre-registered exploitativeness ordering (Appendix H.5). CoEvolve gains most where branches observe what others miss (Section 3.2), as also discussed in Section 4.6.

4.6 Ablation Study

We ablate the collaboration layer at three levels: evidence sharing across branches, the transfer-safeguard modules, and the branch count and coordination hyperparameters. The sharing study spans all six tasks; the module and hyperparameter studies run on MBO under the protocol of Table 1.

Evidence sharing lifts the whole branch population over the Isolated setting. Best-of-seed SQ of CoEvolve exceeds the coordination-only Isolated mode on all six tasks (Figure 6), and removing coordination entirely (Independent) degrades MBO, SPO, and Promoter further. On these tasks the gain decomposes into coordination and sharing components. Moreover, on NHO, Independent matches CoEvolve (95.0 vs 94.9) and coordination pays as convergence speed rather than as final quality (1.5× faster to the same ceiling; Appendix F). Worst-branch SQ (Table 5) improves on five of six tasks (up to +9.4 on MBO), indicating that sharing lifts the entire branch population. The flow itself is real: 244 accepted imports, 205 net-unique after deduplication, with window-level, associational uplift of up to +43.3 SQ points (Table 7).

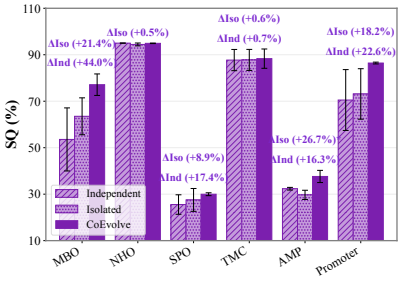


Figure 6: **Ablation of evidence transfer.** Per-task comparison from Independent (w/o coordination) through Isolated (w/o sharing) to CoEvolve.

Discounting is load-bearing; the VoI gate is a tail-risk control. Each knockout disables exactly one safeguard: import discounting ($\alpha \equiv 1$), the safe-VoI gate ($\eta \to -\infty$, every routed import admitted), or the redundancy cap ($R_{\max}=1$). As shown in Figure 7 and Table 6, removing discounting costs the most (-19.0 SQ), ahead of sharing itself (-13.6) and the redundancy cap (-9.6). Together with the shared mode's lead over Isolated on all six tasks (Figure 6), this is the joint signature of Theorem 3.2(a): calibrated sharing helps, and the same imports at full weight collapse. Removing the gate leaves the mean nearly intact (-2.1) but more than doubles the standard deviation ($4.6 \to 11.1$) while admitting over $3\times$ as many imports ($42 \to 137$), indicating that protection against rare harmful imports, exactly the tail-probability role Theorem 3.2(b) and Theorem A.3 assign it.

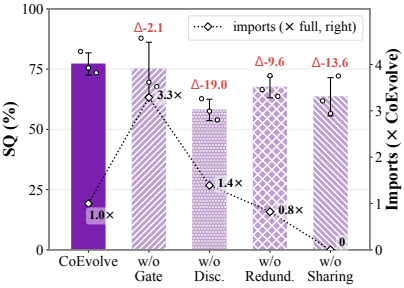


Figure 7: **Module-knockout ablation on MBO**. Accepted imports are normalized to the full system.

The collaboration gain grows with landscape width. Varying the branch count at fixed total budget (Figure 8) gives a task-dependent response: an inverted-U on MBO peaking at the default $K=3$ (77.1 SQ): more branches buy coverage at the price of per-branch depth, flat on AMP, and a rise into a plateau on TMC. The default $K=3$ is therefore not a universal optimum.

The coordination defaults sit near the local optimum. Varying one coordination hyperparameter at a time on MBO (Figure 9), attainment peaks at the default routing quota (3) and redundancy cap ($R_{\max}=0.3$, every off-value degrading by 8.3–16.4 SQ), sits on a plateau in the import cost weight $\lambda \in [1, 2]$, and shows no monotone response to the gate margin η, i.e., the single above-default point ($\eta=0.01$, $+4.3 \pm 18.5$) is within noise. Therefore, η most strongly controls admission volume, whereas $R_{\max}$ barely moves import volume, indicating that benefit is content selectivity.

5 Conclusion

We present CoEvolve, a framework that formulates collaborative scientific discovery as evidence transfer among parallel principle-evolution branches: a coordination core routes observations as gated, discounted factors, keeping the branch-level posteriors separate. CoEvolve attains a mean solution quality of 66.5% against 57.0% for the strong baseline PiEvo, with a $1.80\times$ wall-clock speedup, and is the only system above the published SOTA anchor on five representative auto-research tasks; removing the discount costs more than removing any other module, and the admission gate bounds rare harmful imports. These results make principle co-evolution a practical way for multiple AI Scientists to collaborate on one problem, with evidence transfer as the coupling mechanism.

AI USE STATEMENT

In this work, we used generative AI tools to improve the clarity, grammar, and presentation of the manuscript. All AI-assisted content was carefully reviewed and verified by the authors. All scientific ideas, methodological designs, experimental results, and conclusions were developed and validated by the authors. We take full responsibility for the final content of this work.

REFERENCES

Anthropic. Claude code documentation. https://code.claude.com/docs, 2026.

Jonathan B. Baell and Georgina A. Holloway. New substructure filters for removal of pan assay interference compounds (pains) from screening libraries and for their exclusion in bioassays. *Journal of Medicinal Chemistry*, 53(7):2719–2740, 2010. doi: 10.1021/jm901137j.

Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624:570 – 578, 2023.

Akshat Chaudhari, Janghoon Ock, and Amir Barati Farimani. Modular large language model agents for multi-task computational materials science. *Communications Materials*, 7, 2026.

Pranal Chhetri. Closed-loop agentic ai in drug discovery. *RSC Medicinal Chemistry*, 2026.

Bernardo P. de Almeida, Michael L. Cohn, Arttu Jolma, Peter Rast, Shubham Tripathi, Anna Lara, Pavan Yella, Gunnar Ratsch, Matthew T. Weirauch, and Boris Lenhard. Generation of synthetic human promoter sequences as a resource to study cell-type specific gene expression. *Nature Communications*, 13:3281, 2022.

Tianyu Ding, Aditya Nannapaneni, Bingfan Liu, and Ling Zhang. Autonomous research agents: A survey of ai scientists and the verification gap. *arXiv preprint arXiv:2608.05179*, 2026.

Liqiu Dong, Marta A. Zagorowska, and Mehmet Mercangöz. Bayesian optimization of a multi-product chemical reactor using composite models and partial physics knowledge. *ArXiv*, abs/2606.08611, 2026.

Ihar Faniayeu, Viktar S. Asadchy, and Ivan Fanyaev. Polarization control with helical metasurfaces. *Crystals*, 2020.

Yanlin Fei, Nazhou Liu, Xinmiao Yu, Shaolong Chen, Lei Li, Rahul Thapa, Madalina Ciobanu, Qingqing Mao, and Ritankar Das. How do agents fail on autoresearch: End-to-end diagnostic evaluation on 100 real-world frontier research tasks. *arXiv preprint arXiv:2608.14905*, 2026.

Shiyang Feng, Runmin Ma, Xiangchao Yan, Yue Fan, Yusong Hu, Songtao Huang, Shuaiyu Zhang, Zongsheng Cao, Tianshuo Peng, Jiakang Yuan, Zijie Guo, Zhijie Zhong, Shangheng Du, Weida Wang, Jinxin Shi, Yuhao Zhou, Xiaohan He, Zhiyin Yu, Fangchen Yu, Bihao Zhan, Qihao Zheng, Jiamin Wu, Mianxin Liu, Chi Zhang, Shaowei Hou, Shuya Li, Yankai Jiang, Wenjie Lou, Lilong Wang, Zifu Wang, Jiong Wang, Wanghan Xu, Yue Deng, Dongrui Liu, Yiheng Wang, Wenlong Zhang, Fenghua Ling, Shufei Zhang, Xiaosong Wang, Shuangjia Zheng, Xun Huang, Siqi Sun, Shuyue Hu, Peng Ye, Chunfeng Song, Bin Wang, Conghui He, Yihao Liu, Xin Li, Qibin Hou, Tao Chen, Xiangyu Yue, Bin Wang, Liang He, Dahua Lin, Bowen Zhou, Bo Zhang, and Lei Bai. Internagent-1.5: A unified agentic framework for long-horizon autonomous scientific discovery. *arXiv preprint arXiv:2602.08990*, 2026.

Anna Gaulton, Louisa J. Bellis, A. Patrícia Bento, Jon Chambers, Mark Davies, Anne Hersey, Yvonne Light, Shaun McGlinchey, David Michalovich, Bissan Al-Lazikani, and John P. Overington. Chembl: a large-scale bioactivity database for drug discovery. *Nucleic Acids Research*, 40:D1100 – D1107, 2011.

Gemma Team et al. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Alireza Ghafarollahi and Markus J. Buehler. Sciagents: Automating scientific discovery through multi-agent intelligent graph reasoning. *ArXiv*, abs/2409.05556, 2024.

Ali E. Ghareeb, Benjamin Chang, Ludovico Mitchener, Angela Yiu, Caralyn J. Szostkiewicz, Jon M. Laurent, Muhammed Razzak, Andrew D. White, Michaela M. Hinks, and Samuel G. Rodriques. Robin: A multi-agent system for automating scientific discovery. *ArXiv*, abs/2505.13400, 2025.

Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, Artiom Myaskovsky, Felix Weissenberger, Keran Rong, Ryutaro Tanno, Khaled Saab, Dan George Popovici, Jacob Blum, Fan Zhang, Katherine Chou, Avinatan Hassidim, Burak Gokturk, Amin Vahdat, Pushmeet Kohli, Yossi Matias, Andrew Carroll, Kavita Kulkarni, Nenad Tomašev, Yuan Guan, Vikram Dhillon, Eeshit Dhaval Vaishnav, Byron Lee, Tiago R D Costa, José R. Penadés, Gary Peltz, Yunhan Xu, Annalisa Pawlosky, Alan Karthikesalingam, and Vivek Natarajan. Accelerating scientific discovery with co-scientist. *Nature*, 655:487 – 496, 2025.

Mourad Gridach, Jay Nanavati, Khaldoun Zine El Abidine, Lenon Mendes, and Christina DeFilippo Mack. Agentic ai for scientific discovery: A survey of progress, challenges, and future directions. *ArXiv*, abs/2503.08979, 2025.

Kam Hamidieh. A data-driven statistical model for predicting the critical temperature of a superconductor. *Computational Materials Science*, 154:346–354, 2018.

Emily Herron, Vanessa Lama, Sedrick Bouknight, and Tirthankar Ghosal. From rules to reasoning: A survey of large language model-based approaches to scientific hypothesis and idea generation. *ACM Computing Surveys*, 58:1 – 36, 2026.

Kevin Maik Jablonka, Qiang Ai, Aya Al-Feghali, Shruti Badhwar, Joshua D. Bocarsly, Andres M. Bran, Stefan Bringuier, L. Catherine Brinson, Christian Carbogno, Tobias Cerny, et al. Leveraging large language models for predictive chemistry. *Nature Machine Intelligence*, 6:432–442, 2024.

Jiachen Jiang, Tianyu Ding, and Zhihui Zhu. Deltaevolve: Accelerating scientific discovery through momentum-driven evolution. *ArXiv*, abs/2602.02919, 2026.

Jiajie Jin, Yuyang Hu, Kai Qiu, Qi Dai, Chong Luo, Guanting Dong, Xiaoxi Li, Tong Zhao, Xiaolong Ma, Gongrui Zhang, Zhirong Wu, Bei Liu, Zhengyuan Yang, Linjie Li, Lijuan Wang, Hongjin Qian, Yutao Zhu, and Zhicheng Dou. Toward generalist autonomous research via hypothesis-tree refinement, 2026.

Mingze Li, Yu Rong, Songyou Li, Lihong Wang, Jiacheng Cen, Liming Wu, Anyi Li, Zongzhao Li, Qiuliang Liu, Rui Jiao, Tian Bian, Pengju Wang, Hao Sun, Jianfeng Zhang, Ji-Rong Wen, Deli Zhao, Shifeng Jin, Tingyang Xu, and Wenbing Huang. Agentic fusion of large atomic and language models to accelerate superconductor discovery, 2026.

Moyan Li and Raed Kontar. On negative transfer and structure of latent functions in multi-output gaussian processes. *SIAM/ASA J. Uncertain. Quantification*, 10:1714–1732, 2022.

Haoyang Liu, Yijiang Li, and Haohan Wang. Genomas: A multi-agent framework for scientific discovery via code-driven gene expression analysis. *ArXiv*, abs/2507.21035, 2025.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Nicolaus Foerster, Jeff Clune, and David Ha. The ai scientist: Towards fully automated open-ended scientific discovery. *ArXiv*, abs/2408.06292, 2024.

Yougang Lyu, Xi Zhang, Xinhao Yi, Yuyue Zhao, Shuyu Guo, Wenxiang Hu, Jan Piotrowski, Jakub Kaliski, Jacopo Urbani, Zaiqiao Meng, Lu Zhou, and Xiaohu Yan. Evoscientist: Towards multi-agent evolving ai scientists for end-to-end scientific discovery. *ArXiv*, abs/2603.08127, 2026.

A. R. Miedema, P. F. de Châtel, and F. R. de Boer. Cohesion in alloys — fundamentals of a semi-empirical model. *Physica B+C*, 100(1):1–28, 1980.

Ludovico Mitchener, Angela Yiu, Benjamin Chang, Mathieu Bourdenx, Tyler Nadolski, Arvis Sulovari, Eric C. Landsness, Dániel L. Barabási, Siddharth Narayanan, Nicky Evans, Shriya Reddy, Martha S. Foiani, Aizad Kamal, Leah P. Shriver, Fang Cao, Asmamaw T. Wassie, Jon M. Laurent, Edwin Melville-Green, Mayk Caldas Ramos, Albert Bou, Kaleigh F. Roberts, Sladjana Zagorac, Timothy C. Orr, Miranda E. Orr, Kevin J. Zwezdaryk, Ali E. Ghareeb, Laurie McCoy, Bruna Gomes, Euan A Ashley, Karen E. Duff, Tonio Buonassisi, Tom Rainforth, Randall J. Bateman, Michael Skarlinski, Samuel G. Rodriques, Michaela M. Hinks, and Andrew D. White. Kosmos: An ai scientist for autonomous discovery. *ArXiv*, abs/2511.02824, 2025.

Ngoc Tuong Vy Nguyen, Felix D Childress, and Yunting Yin. Debate-driven multi-agent llms for phishing email detection. In *2025 13th International Symposium on Digital Forensics and Security (ISDFS)*, pp. 1–5. IEEE, 2025.

OpenAI. Gpt-5.6 system card. Technical report, OpenAI Deployment Safety Hub, July 2026a. URL https://deploymentsafety.openai.com/gpt-5-6. Accessed: 2026-09-20.

OpenAI. Unrolling the codex agent loop. https://openai.com/index/unrolling-the-codex-agent-loop/, 2026b.

Gihan Panapitiya, Emily Saldanha, Heather Job, and Olivia Hess. Autolabs: Cognitive multi-agent systems with self-correction for autonomous chemical experimentation. *ArXiv*, abs/2509.25651, 2025.

Yingming Pu, Tao Lin, and Hongyu Chen. Piflow: Principle-aware scientific discovery with multi-agent collaboration. *ArXiv*, abs/2505.15047, 2025.

Yingming Pu, Tao Lin, and Hongyu Chen. Principle-evolvable scientific discovery via uncertainty minimization. *ArXiv*, abs/2602.06448, 2026.

Yao Qu and Meng Lu. Bilevel autoresearch: Meta-autoresearching itself. *ArXiv*, abs/2603.23420, 2026.

Arno Rauschenbeutel. Chiral quantum optics. *2017 Conference on Lasers and Electro-Optics Europe & European Quantum Electronics Conference (CLEO/Europe-EQEC)*, pp. 1–1, 2017.

Zhaolin Ren and Na Li. Ts-rsr: A provably efficient approach for batch bayesian optimization. *SIAM J. Optim.*, 35:2155–2181, 2025.

Célio Dias Santos-Junior, Shaojun Pan, Xing-Ming Zhao, and Luis Pedro Coelho. Macrel: antimicrobial peptide screening in genomes and metagenomes. *PeerJ*, 8:e10555, 2020.

T. Savinov, N. Papasimakis, V. A. Fedotov, and N. I. Zheludev. Toroidal circular dichroism. *Science Advances*, 2(5):e1501716, 2016.

Travis Smith. The little scientist: Llm agent-driven discovery via the scientific method, 2026.

Zhangde Song, Jieyu Lu, Yuanqi Du, Botao Yu, Thomas M Pruyn, Yue Huang, Kehan Guo, Xiuzhe Luo, Yuanhao Qu, Yi Qu, et al. Evaluating large language models in scientific discovery. *arXiv preprint arXiv:2512.15567*, 2025.

Haoyang Su, Renqi Chen, Shixiang Tang, Zhenfei Yin, Xinzhe Zheng, Jinzhe Li, Biqing Qi, Qi Wu, Hui Li, Wanli Ouyang, et al. Many heads are better than one: Improved scientific idea generation by a llm-based multi-agent system. In *Proceedings of the 63rd annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 28201–28240, 2025.

Shuhei Sugiura, Ichiro Takeuchi, and Shion Takeno. Randomized kriging believer for parallel bayesian optimization with regret bounds. *ArXiv*, abs/2603.01470, 2026.

Jiabin Tang, Lianghao Xia, Zhonghang Li, and Chao Huang. Ai-researcher: Autonomous scientific innovation. *ArXiv*, abs/2505.18705, 2025.

Mengru Wang, Junfeng Fang, Shuofei Qiao, Zhenqian Xu, Haoming Xu, Haoxiong Wang, Shumin Deng, Linyi Yang, Xin Xu, Yunzhi Yao, Dan Zhang, Fei Shen, Zhixiang Cui, Buqiang Xu, Haozhe Luo, Yunxiang Wei, Ningyu Zhang, Julian McAuley, Tat Seng Chua, and Huajun Chen. Mechanist: Ai as a scientific instrument for discovering the mechanisms of intelligence, 2026.

Jiaqi Wei, Yuejin Yang, Xiang Zhang, Yuhan Chen, Zhuang Xiang, Zhangyang Gao, Dongzhan Zhou, Guangshuai Wang, Zhiqiang Gao, Juntai Cao, Zijie Qiu, Xuming He, Qiang Zhang, Chenyu You, Shuangjia Zheng, Ning Ding, Wanli Ouyang, Nanqing Dong, Yu Cheng, Siqi Sun, Lei Bai, and Bowen Zhou. From ai for science to agentic science: A survey on autonomous scientific discovery. *ArXiv*, abs/2508.14111, 2025.

Yunyue Wei, Zeji Yi, Hongda Li, Saraswati Soedarmadji, and Yanan Sui. Safe bayesian optimization for the control of high-dimensional embodied systems. *arXiv preprint arXiv:2412.20350*, 2024.

Yixuan Weng, Minjun Zhu, Qiujie Xie, Qiyao Sun, Zhen Lin, Sifan Liu, and Yue Zhang. Deepscientist: Advancing frontier-pushing scientific findings progressively. *ArXiv*, abs/2509.26603, 2025.

Juanshu Wu, Yingming Pu, Jin Wang, Bing Gu, Xin Chen, and Hongyu Chen. Machine learned structure-property correlation between nanohelices and circular dichroism. *Advanced Optical Materials*, 13(9):2402595, 2025.

Amy Xin, Jiening Siow, Junjie Wang, Zijun Yao, Fanjin Zhang, Jian Song, Lei Hou, and Juanzi Li. Eurekagent: Agent environment engineering is all you need for autonomous scientific discovery. *ArXiv*, abs/2606.13662, 2026.

Xiaoyu Xiong, Yuqi Ren, and Deyi Xiong. Evosci: A bio-inspired multi-agent framework for the evolution of scientific discovery. *ArXiv*, abs/2605.24018, 2026.

Yutaro Yamada, Robert Tjarko Lange, Cong Lu, Shengran Hu, Chris Lu, Jakob Nicolaus Foerster, Jeff Clune, and David Ha. The ai scientist-v2: Workshop-level automated scientific discovery via agentic tree search. *ArXiv*, abs/2504.08066, 2025.

Jianwei Yang, Ohm Rishabh Venkatachalam, Mohammad Kianezhad, Sharvaree P. Vadgama, and Rose Yu. Think like a scientist: Physics-guided llm agent for equation discovery. *ArXiv*, abs/2602.12259, 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *ArXiv*, abs/2210.03629, 2022.

Zongyang Yuan, Zechang Zhang, Qinbin Li, Lailong Luo, Deke Guo, and Mingrui Lao. Edagent: A concurrent orchestration method for collaborative llm multi-agent system. In *2026 IEEE 46th International Conference on Distributed Computing Systems (ICDCS)*, pp. 261–271. IEEE, 2026.

Mingming Zhao, Jiqian Dong, Kangping Xu, Zadid Hasan, Chengrui Fan, Shan Jiang, Shuai Mao, Yating Ling, Linyi Zou, Tailin Zhou, Yun Hin Chan, Wenkai Zhang, Zhanhong Zhou, Guowei Huang, Hongliang Li, Wenjing Cun, Zhitang Chen, Mingxuan Yuan, and Yanhui Geng. Scienceflow: A long-horizon agent for ml research, scientific discovery and beyond, 2026.

Adam Żychowski, Andrew Perrault, and Jacek Mandziuk. Cultivating archipelago of forests: Evolving robust decision trees through island coevolution. In *AAAI Conference on Artificial Intelligence*, 2025.

A Formal Assumptions and Guarantees

We state the assumptions used by the methodology before the corresponding results; complete proofs appear in Section B. Let H_t^{sys} be the complete system history after round t, let $\mathcal{F}_t = \sigma(H_t^{\mathrm{sys}})$, and define the information acquired by policy Π as

$$\mathcal{U}_\Pi(T) := I_\Pi(P; H_T^{\mathrm{sys}}). \tag{6}$$

The four results below follow a common pattern: each states the assumptions it needs on the learning process, followed by the guarantee they buy. In order, they certify that parallel branches cover information faster (Theorem A.1), that discounted sharing does not break posterior concentration (Theorem A.2), that the VoI gate controls harmful imports in probability (Theorem A.3), and that parallel search shortens the time to discovery (Theorem A.4).

Assumption 1 (Finite support and positive prior)**.** *The principle universe $\bar{\mathcal{P}}$ is finite and $\pi_0(P) > 0$ for every $P \in \bar{\mathcal{P}}$.*

Coverage. The first guarantee asks that branches do not duplicate one another. Assumption 2 makes this precise through a redundancy factor $\underline{\gamma}_T$: if all branches observed the same evidence, the joint information would collapse to a single branch's share and $\underline{\gamma}_T$ would be near $1/K$; if their evidence were fully complementary, $\underline{\gamma}_T$ would approach 1.

Assumption 2 (Complementary information)**.** *On each active round, let $I_{j,t} := I(P; Y_{j,t} \mid \mathcal{F}_{t-1}, h_{j,t})$. For a deterministic $\underline{\gamma}_T \in [0,1]$,*

$$\underbrace{I(P; Y_{1,t}, \ldots, Y_{K,t} \mid \mathcal{F}_{t-1}, h_{1:K,t})}_{\text{what the system learns jointly}} \geq \underbrace{\underline{\gamma}_T}_{\text{complementarity}} \underbrace{\sum_{j=1}^{K} I_{j,t}}_{\text{sum of per-branch information}}. \tag{7}$$

Assumption 3 (Per-branch information rate)**.** *There are $\mu_j \geq 0$ such that $I_{j,t} \geq \mu_j$ on active rounds. If $\mathcal{A}_T$ is the set of active rounds, then $|\mathcal{A}_T| \geq (1-\chi_T)T$ for a deterministic $\chi_T \in [0,1]$.*

Assumption 3 adds two floors: every active round teaches branch j at least μ_j nats, and at least a $(1-\chi_T)$ fraction of rounds are active. Together they yield the first guarantee.

Theorem A.1 (Parallel coverage)**.** *Under Assumptions 2–3, before entropy saturation,*

$$\mathcal{U}_{\textsc{CoEvolve}}(T) \geq \underbrace{(1-\chi_T)T}_{\text{active rounds}} \underbrace{\underline{\gamma}_T \sum_{j=1}^{K} \mu_j}_{\text{complementary per-round rate}}.$$

In words, the system accumulates mutual information at a rate proportional to the number of branches, discounted only by redundancy ($\underline{\gamma}_T$) and idle rounds (χ_T): when branches explore complementary directions, K parallel branches certify nearly K times the information of one branch over the same wall-clock horizon.

Concentration. Coverage alone does not say the system identifies the true principle; the second guarantee does. Assumption 4 requires the accumulated log-likelihood contrast in favor of $P^\star$ to grow linearly in the number of effective samples $N_{j,T}^{\mathrm{eff}}$, where imported evidence enters with discounted (hence fractional) weights, up to a slack term $\mathfrak{r}_j$ that holds uniformly over time and over all wrong principles at once.

Assumption 4 (Calibrated discounted contrast)**.** *For each branch j and $P \neq P^\star$, let $L_{j,T}(P)$ be the accumulated local and discounted-import log-likelihood ratio in favor of $P^\star$, and let $N_{j,T}^{\mathrm{eff}}$ be the sum of its predictable weights in $[0,1]$. There is a $\kappa_j(P) > 0$ and a time-uniform radius $\mathfrak{r}_j(n,\delta)$ such that, with probability at least $1-\delta$, simultaneously over T and $P \neq P^\star$,*

$$L_{j,T}(P) \geq \underbrace{\kappa_j(P) N_{j,T}^{\mathrm{eff}}}_{\text{contrast accumulating at rate } \kappa_j(P)} - \underbrace{\mathfrak{r}_j(N_{j,T}^{\mathrm{eff}}, \delta)}_{\text{time-uniform confidence slack}}. \tag{8}$$

We assume further that every likelihood ratio entering $L_{j,T}$ is finite on data generated under $P^\star$.

Theorem A.2 (Concentration under discounted imports)**.** *Under Assumptions 1 and 4, with probability at least* $1-\delta$*,*

$$p_{j,T}(P^\star) \geq 1 - \underbrace{\frac{|\bar{\mathcal{P}}|-1}{\pi_0(P^\star)}}_{\text{prior scale}} \exp\{-\kappa_j N_{j,T}^{\text{eff}} + \mathfrak{r}_j(N_{j,T}^{\text{eff}}, \delta)\},$$

where $\kappa_j = \min_{P \neq P^\star} \kappa_j(P)$.

Since $\mathfrak{r}_j(n,\delta)$ grows sublinearly in n (of order $\sqrt{n \log\log n}$ for bounded increments), the exponent is eventually dominated by $-\kappa_j N_{j,T}^{\text{eff}}$, so the posterior mass on the true principle converges to 1 exponentially fast in effective samples—even though part of those samples arrived as discounted imports from other branches.

Negative-transfer control. The third guarantee concerns the gate that decides which imports to accept. It needs one ingredient: the gate's estimate $\widehat{V}_{j,t}(C)$ of a candidate's true net value $V_{j,t}(C)$ must come with a calibrated error radius.

Assumption 5 (Calibrated value estimates)**.** *For every adaptively considered candidate,*

$$|\widehat{V}_{j,t}(C) - V_{j,t}(C)| \leq r_{j,t}(C,\delta) \tag{9}$$

with conditional probability at least $1-\delta$.

Theorem A.3 (Negative-transfer control)**.** *Under Assumption 5, the probability of accepting any candidate with* $V_{j,t}(C) \leq \eta$ *is at most* $\sum_{j,t,C} \delta_{j,t,C}$ *when the gate in Eq. 2 is used.*

Intuitively, a harmful candidate can pass the gate only if its estimate errs by more than its own radius, an event whose probability is capped by $\delta_{j,t,C}$; summing these caps over all candidates bounds the total failure budget. The gate therefore acts as a tail-probability control on harmful imports, with δ as a user-chosen budget, and the guarantee is exactly as strong as the calibration of the radius $r_{j,t}$ feeding it (see Remark B.4).

Discovery. The last guarantee models discovery as a race: each branch independently rolls a die every round, and the first success ends the race. Assumption 6 only asks that each branch's per-round success probability is bounded below by some $\lambda_j > 0$.

Assumption 6 (Discovery hazards)**.** *Before discovery, active branches discover a* $P^\star$*-consistent discriminator independently conditional on history, with per-round probability at least* λ_j.

Theorem A.4 (Parallel discovery)**.** *Under Assumption 6,*

$$\Pr(\tau_{\text{disc}} > t) \leq \underbrace{\prod_j (1-\lambda_j)^t}_{\substack{\text{no branch discovers} \\ \text{in } t \text{ rounds}}} \leq \exp\Big(-t \sum_j \lambda_j\Big). \tag{10}$$

The tail probability of the discovery time decays exponentially at rate $\sum_j \lambda_j$: running K branches multiplies the effective hazard, so the expected time to first discovery shrinks roughly by a factor of K when the per-branch hazards are small and comparable.

B Proofs of the Theoretical Guarantees

In this section, we give the complete proofs of the results stated in Appendix A.

B.1 Proof of parallel mutual-information coverage

Proof of Theorem A.1. Write the round-t system-history increment as $Z_t = (h_{1:K,t}, Y_{1:K,t}, W_t)$, where W_t collects every remaining logged increment of round t (routing decisions, verification outcomes, and monitoring records). We telescope (6) over rounds and then lower-bound each round's contribution.

Telescoping the system information over rounds. Since $H_T^{\mathrm{sys}} = (Z_1, \dots, Z_T)$, the chain rule for mutual information gives

$$\mathcal{U}_{\textsc{CoEvolve}}(T) = I(P; H_T^{\mathrm{sys}}) = \sum_{t=1}^{T} I(P; Z_t \mid H_{t-1}^{\mathrm{sys}}) = \sum_{t=1}^{T} \mathbb{E}[I(P; Z_t \mid \mathcal{F}_{t-1})]\,, \tag{11}$$

where $I(P; Z_t \mid \mathcal{F}_{t-1})$ denotes the *history-wise* conditional mutual information, i.e., the $\mathcal{F}_{t-1}$-measurable random variable

$$I(P; Z_t \mid \mathcal{F}_{t-1}) := \mathrm{D}_{\mathrm{KL}}\Big(p_{P,Z_t|H_{t-1}^{\mathrm{sys}}} \,\Big\|\, p_{P|H_{t-1}^{\mathrm{sys}}} \otimes p_{Z_t|H_{t-1}^{\mathrm{sys}}}\Big)\,,$$

and the last equality in (11) is the tower property: $I(P; Z_t \mid H_{t-1}^{\mathrm{sys}})$ is by definition the expectation of this kernel over the random history. Every such kernel is bounded by $H_{\pi_0}(P) \le \log|\bar{\mathcal{P}}| < \infty$ under the standing finite-universe setting, so the telescoping and the interchange with expectation are legitimate. Adaptivity of the design poses no difficulty here—conditioning on the full $\mathcal{F}_{t-1}$ is precisely what makes each summand a well-defined information quantity, regardless of how the round-t experiments were selected.

Hypothesis selection carries no extra information. Each branch selects $h_{j,t}$ from its own history, which is a sub-σ-algebra of $\mathcal{F}_{t-1}$, possibly augmented by exogenous randomization independent of P given $\mathcal{F}_{t-1}$. Hence $I(P; h_{1:K,t} \mid \mathcal{F}_{t-1}) = 0$, and applying the chain rule to Z_t conditionally on $\mathcal{F}_{t-1}$,

$$I(P; Z_t \mid \mathcal{F}_{t-1}) = \underbrace{I(P; h_{1:K,t} \mid \mathcal{F}_{t-1})}_{=0} + I(P; Y_{1:K,t} \mid \mathcal{F}_{t-1}, h_{1:K,t}) + I(P; W_t \mid \mathcal{F}_{t-1}, h_{1:K,t}, Y_{1:K,t}).$$

Both surviving terms are nonnegative; keeping only the observation term,

$$I(P; Z_t \mid \mathcal{F}_{t-1}) \;\ge\; I(P; Y_{1,t}, \dots, Y_{K,t} \mid \mathcal{F}_{t-1}, h_{1:K,t}). \tag{12}$$

This is the precise sense in which routing and logging increments are "free": discarding them can only weaken the bound.

Lower-bounding each active round. Whether round t is active is decided from the history before the round is executed, so the indicator $\mathbf{1}\{t \in \mathcal{A}_T\}$ is $\mathcal{F}_{t-1}$-measurable. On an active round, Assumption 2 applies to the right-hand side of (12) and Assumption 3 floors each summand:

$$I(P; Z_t \mid \mathcal{F}_{t-1}) \;\ge\; \underline{\gamma}_T \sum_{j=1}^{K} I_{j,t} \;\ge\; \underline{\gamma}_T \sum_{j=1}^{K} \mu_j.$$

On an inactive round we retain only nonnegativity of conditional mutual information. Combining the two cases,

$$I(P; Z_t \mid \mathcal{F}_{t-1}) \;\ge\; \mathbf{1}\{t \in \mathcal{A}_T\}\, \underline{\gamma}_T \sum_{j=1}^{K} \mu_j \qquad \text{almost surely.}$$

Summing the per-round bounds over the horizon. Inserting into (11) and using that $\underline{\gamma}_T$ and the μ_j are deterministic,

$$\mathcal{U}_{\textsc{CoEvolve}}(T) \;\ge\; \underline{\gamma}_T \sum_{j=1}^{K} \mu_j\, \mathbb{E}\left[\sum_{t=1}^{T} \mathbf{1}\{t \in \mathcal{A}_T\}\right] \;=\; \mathbb{E}|\mathcal{A}_T|\, \underline{\gamma}_T \sum_{j=1}^{K} \mu_j \;\ge\; (1 - \chi_T) T\, \underline{\gamma}_T \sum_{j=1}^{K} \mu_j,$$

where the last step uses $|\mathcal{A}_T| \ge (1 - \chi_T)T$ almost surely. Every step holds history-wise, so no independence assumption across rounds or branches enters the argument. □

Remark B.1 (The pre-saturation clause)**.** *Mutual information about P never exceeds the prior entropy, $\mathcal{U}_{\text{CoEvolve}}(T) \le H_{\pi_0}(P)$, so a per-round floor $I_{j,t} \ge \mu_j > 0$ cannot hold indefinitely: Assumption 3, and hence the bound of Theorem A.1, is consistent only for horizons satisfying $(1-\chi_T)T\,\underline{\gamma}_T \sum_{j=1}^{K} \mu_j \le H_{\pi_0}(P)$. The qualifier "before entropy saturation" in the theorem statement refers to this regime; once the posterior concentrates, the per-round increment is governed by the residual entropy $H(P \mid \mathcal{F}_{t-1})$ rather than by any constant floor, and the rate must be re-derived accordingly.*

B.2 Proof of branch concentration under discounted imports

Proof of Theorem A.2. The proof has three moves: writing out the discounted log-contrast explicitly, bounding the posterior tail through the posterior odds, and inserting the simultaneous contrast bound.

Writing out the discounted log-contrast. Dividing (5) evaluated at P by the same expression at $P^\star$ eliminates the normalizer $Z_{j,T}$ and yields the posterior-odds identity

$$\frac{p_{j,T}(P)}{p_{j,T}(P^\star)} = \frac{\pi_0(P)}{\pi_0(P^\star)} \exp\big[-L_{j,T}(P)\big], \tag{13}$$

where the accumulated local and discounted-import log-likelihood ratio in favor of $P^\star$ is

$$L_{j,T}(P) := \sum_{(h,y)\in\mathcal{D}_{j,T}} \log \frac{p_j(y \mid h, P^\star)}{p_j(y \mid h, P)} + \sum_{(i,h,y,\alpha)\in\mathcal{I}_{j,T}} \alpha \log \frac{q_{j\leftarrow i}(y \mid h, P^\star)}{q_{j\leftarrow i}(y \mid h, P)}. \tag{14}$$

This is exactly the quantity of Assumption 4: local observations enter with weight 1 and imports enter with their discount $\alpha_{j\leftarrow i}$, all weights lie in $[0,1]$, and each weight is computed before the corresponding factor is injected into the posterior, hence is predictable with respect to the branch history. Consequently $N_{j,T}^{\text{eff}}$, the sum of these weights, counts one full effective sample per local observation and α per import. The finiteness clause of Assumption 4 ensures that every logarithm in (14) is finite almost surely on data generated under $P^\star$, so the odds identity is well defined with no zero-over-zero ambiguity.

Bounding the posterior tail through the odds. The odds identity above also shows $p_{j,T}(P^\star) > 0$ almost surely: its unnormalized mass is a product of $\pi_0(P^\star) > 0$ with likelihood factors that are positive almost surely on data generated under $P^\star$ by the finiteness clause. Since moreover $p_{j,T}(P^\star) \le 1$, dividing by it can only enlarge each term, so

$$1 - p_{j,T}(P^\star) = \sum_{P\neq P^\star} p_{j,T}(P) \;\le\; \sum_{P\neq P^\star} \frac{p_{j,T}(P)}{p_{j,T}(P^\star)} = \sum_{P\neq P^\star} \frac{\pi_0(P)}{\pi_0(P^\star)} \exp\big[-L_{j,T}(P)\big], \tag{15}$$

where the last equality is (13). No probabilistic argument has been used: (15) holds pathwise on every realization of the evidence.

Inserting the simultaneous contrast bound. Let $\mathcal{E}_{\text{conc}}$ denote the event in (8), which has probability at least $1-\delta$ and holds simultaneously over all rounds T and all $P \neq P^\star$. On $\mathcal{E}_{\text{conc}}$, for every $P \neq P^\star$,

$$L_{j,T}(P) \;\ge\; \kappa_j(P)\, N_{j,T}^{\text{eff}} - \mathfrak{r}_j(N_{j,T}^{\text{eff}}, \delta) \;\ge\; \kappa_j\, N_{j,T}^{\text{eff}} - \mathfrak{r}_j(N_{j,T}^{\text{eff}}, \delta),$$

because $\kappa_j(P) \ge \kappa_j := \min_{P\neq P^\star} \kappa_j(P)$ and $N_{j,T}^{\text{eff}} \ge 0$. Substituting into (15) and using $\pi_0(P) \le 1$ for each of the $|\bar{\mathcal{P}}| - 1$ wrong principles,

$$\begin{aligned} 1 - p_{j,T}(P^\star) \;&\le\; \frac{\exp\big\{-\kappa_j N_{j,T}^{\text{eff}} + \mathfrak{r}_j(N_{j,T}^{\text{eff}}, \delta)\big\}}{\pi_0(P^\star)} \sum_{P\neq P^\star} \pi_0(P) \\ &\le\; \frac{|\bar{\mathcal{P}}| - 1}{\pi_0(P^\star)} \exp\big\{-\kappa_j N_{j,T}^{\text{eff}} + \mathfrak{r}_j(N_{j,T}^{\text{eff}}, \delta)\big\}, \end{aligned} \tag{16}$$

which is the claim; the prefactor is finite because $\pi_0(P^\star) > 0$ by Assumption 1. Since $\mathcal{E}_{\text{conc}}$ is uniform in T, the same conclusion holds with $N_{j,\tau}^{\text{eff}}$ in place of $N_{j,T}^{\text{eff}}$ at any adaptive routing or stopping time τ: no optional-sampling correction is required. □

Remark B.2 (Constants and sharpness)**.** *Two features of* (16) *affect its use. First, the prior enters only through* $1/\pi_0(P^\star)$*: with a uniform prior on* $M = |\bar{\mathcal{P}}|$ *principles the prefactor is* $M(M-1)$*, so concentration is fastest relative to the universe size when the prior already places non-negligible mass on* $P^\star$*. Second, the first line of* (16) *shows that the sharper, prior-dependent constant* $\sum_{P \neq P^\star} \pi_0(P) = 1 - \pi_0(P^\star)$ *is available; Theorem A.2 states the looser* $|\bar{\mathcal{P}}| - 1$ *form, which is prior-independent and matches the union-bound structure of the argument. Finally, the bound is vacuous until* $\kappa_j N^{\mathrm{eff}}_{j,T}$ *exceeds* $\mathfrak{r}_j(N^{\mathrm{eff}}_{j,T}, \delta) + \log\big((|\bar{\mathcal{P}}| - 1)/\pi_0(P^\star)\big)$*; it is informative only once the effective evidence count has cleared this crossover, which is the formal content of the informal claim that an import counts only at its discounted weight.*

B.3 Proof of negative-transfer control

Proof of Theorem A.3. Let $\mathcal{C}_T$ denote the collection of all triples (j, t, C) for which candidate C is presented to branch j's gate during the run. For each $(j, t, C) \in \mathcal{C}_T$ define the calibration event

$$\mathcal{E}_{j,t,C} := \big\{\, |\widehat{V}_{j,t}(C) - V_{j,t}(C)| \le r_{j,t}(C, \delta_{j,t,C}) \,\big\}, \qquad \Pr(\mathcal{E}^c_{j,t,C}) = \mathbb{E}\big[\Pr\big(\mathcal{E}^c_{j,t,C} \,\big|\, \mathcal{F}_{t-1}, C\big)\big] \le \delta_{j,t,C},$$

where the conditional bound is (9) and the outer expectation removes the conditioning. Conditioning on $(\mathcal{F}_{t-1}, C)$ is what legitimizes treating an adaptively generated candidate as fixed at the moment it is gated; this is the only place where Assumption 5 is used.

The gate is safe on each calibration event. On $\mathcal{E}_{j,t,C}$, if the true net value satisfies $V_{j,t}(C) \le \eta$, then

$$\widehat{V}_{j,t}(C) - r_{j,t}(C, \delta_{j,t,C}) \;\le\; V_{j,t}(C) \;\le\; \eta,$$

so the strict inequality in (2) fails and the candidate is rejected. Contrapositively, any *accepted* candidate with $V_{j,t}(C) \le \eta$ must lie in $\mathcal{E}^c_{j,t,C}$; hence

$$\big\{\text{some accepted candidate has } V_{j,t}(C) \le \eta\big\} \;\subseteq \bigcup_{(j,t,C) \in \mathcal{C}_T} \mathcal{E}^c_{j,t,C}.$$

Union-bounding over the adaptive candidate stream. Whether a triple (j, t, C) is ever gated is itself determined by the history and the generated candidate, i.e., the event $\{(j, t, C) \in \mathcal{C}_T\}$ is $\sigma(\mathcal{F}_{t-1}, C)$-measurable. Hence each calibration statement can be restricted to the realized stream without changing its bound:

$$\Pr\big(\mathcal{E}^c_{j,t,C} \cap \{(j, t, C) \in \mathcal{C}_T\}\big) = \mathbb{E}\big[\mathbf{1}\{(j, t, C) \in \mathcal{C}_T\}\, \Pr\big(\mathcal{E}^c_{j,t,C} \,\big|\, \mathcal{F}_{t-1}, C\big)\big] \;\le\; \delta_{j,t,C}.$$

A union bound over the stream then gives

$$\Pr\big(\text{any accepted import has } V_{j,t}(C) \le \eta\big) \;\le\; \Pr\left(\bigcup_{(j,t,C) \in \mathcal{C}_T} \mathcal{E}^c_{j,t,C}\right) \;\le\; \sum_{j,t,C} \delta_{j,t,C},$$

where the last sum runs over the per-candidate budgets allocated along the realized stream, exactly as in the theorem statement. Adaptivity of the stream is therefore harmless: the failure probabilities add regardless of how the sequence of candidates was generated. □

Remark B.3 (Budgeting the failure probability)**.** *If a deterministic bound* M_T *on the number of gated candidates is known, the uniform allocation* $\delta_{j,t,C} = \delta/M_T$ *caps the total failure probability at* δ*. For an unbounded adaptive stream the budget is spent across the sequence—for example, assigning* $\delta_m = 6\delta/(\pi^2 m^2)$ *to the* m*-th candidate—so that* $\sum_{m \ge 1} \delta_m = \delta$ *and the guarantee holds for the entire run at level* δ*.*

Remark B.4 (What the theorem does and does not control)**.** *The guarantee is stated relative to the* defined *net value* (1)*, i.e., the entropy-reduction potential net of the certified transfer cost, and it is only as strong as the radius* $r_{j,t}$ *entering the gate. As delimited in Section 3.2, the deterministic implementation's exact-enumeration radius* $r = 0$ *certifies the numerical update but not model misspecification or residual context shift; an externally validated or replication-based radius is required for the theorem to bind in deployment. In particular, bounded rewards alone do not make harmful imports rare—the control comes entirely from the calibration of* $\widehat{V}_{j,t}$*.*

B.4 Proof of parallel discovery under hazards

Proof of Theorem A.4. Let $D_{j,s}$ denote the event that branch j discovers a $P^\star$-consistent discriminator in round s, and let

$$N_s := \{\tau_{\text{disc}} > s\} = \bigcap_{s' \le s} \bigcap_{j} \overline{D_{j,s'}}$$

be the event that no branch has discovered by the end of round s; note $N_s \in \mathcal{F}_s$.

One round of non-discovery. On the event N_{s-1} every branch is still in the pre-discovery regime, so Assumption 6 applies: conditionally on $\mathcal{F}_{s-1}$, the discovery events $D_{j,s}$ of the active branches are independent with $\Pr(D_{j,s} \mid \mathcal{F}_{s-1}) \ge \lambda_j$. Therefore, on N_{s-1},

$$\Pr(N_s \mid \mathcal{F}_{s-1}) = \Pr\Big(\bigcap\nolimits_j \overline{D_{j,s}} \,\Big|\, \mathcal{F}_{s-1}\Big) = \prod_j \big(1 - \Pr(D_{j,s} \mid \mathcal{F}_{s-1})\big) \;\le\; \prod_j (1-\lambda_j),$$

where the middle equality is the conditional independence across branches. The product ranges over the branches active in round s; a branch that is inactive in round s is equivalently covered by reading its hazard floor as 0 for that round, which contributes a factor 1, so the displayed bound holds with the λ_j of the branches that are active throughout the pre-discovery regime. The conditional independence is exactly where the argument uses that the branches' discovery mechanisms share no common failure mode beyond the observed history.

Iterating the one-round bound over rounds. Since $N_s \subseteq N_{s-1}$ and $N_{s-1} \in \mathcal{F}_{s-1}$, the conditional probability $\Pr(N_s \mid \mathcal{F}_{s-1})$ vanishes off N_{s-1}; hence the one-round bound applies through the tower property,

$$\Pr(N_s) = \mathbb{E}[\mathbf{1}\{N_{s-1}\} \Pr(N_s \mid \mathcal{F}_{s-1})] \;\le\; \Big(\prod_j (1-\lambda_j)\Big) \Pr(N_{s-1}).$$

Starting from $\Pr(N_0) = 1$ and iterating over $s = 1, \dots, t$,

$$\Pr(\tau_{\text{disc}} > t) = \Pr(N_t) \;\le\; \prod_j (1-\lambda_j)^t.$$

Converting the product bound to exponential form. Since $\log(1-x) \le -x$ for $x \in [0,1)$ (if some $\lambda_j = 1$ the product is 0 and the claim is trivial),

$$\prod_j (1-\lambda_j)^t = \exp\Big(t \sum_j \log(1-\lambda_j)\Big) \;\le\; \exp\Big(-t \sum_j \lambda_j\Big),$$

which is (10). □

Remark B.5 (Expectation bound and the role of independence)**.** *Summing the tail bound yields an expectation bound at no extra cost: since τ_{disc} is a positive integer-valued random variable,*

$$\mathbb{E}[\tau_{\text{disc}}] = \sum_{t \ge 0} \Pr(\tau_{\text{disc}} > t) \;\le\; \sum_{t \ge 0} \Big(\prod_j (1-\lambda_j)\Big)^t = \frac{1}{1 - \prod_j (1-\lambda_j)},$$

by the geometric-series identity. With a common hazard λ this reads $\mathbb{E}[\tau_{\text{disc}}] \le 1/(1-(1-\lambda)^K) \approx 1/(K\lambda)$ for small λ—the near-K-fold speedup of parallel search. The conditional-independence clause of Assumption 6 is essential for this conclusion: were the discovery events perfectly coupled, the one-round non-discovery probability would be $1 - \max_j \lambda_j$ instead of $\prod_j (1-\lambda_j)$, and no speedup would follow. The theorem therefore certifies acceleration only to the extent that the branch construction genuinely diversifies the discovery mechanisms, which is a measurable property of the design rather than a consequence of running K copies.

C THEOREM–EXPERIMENT ALIGNMENT

The four guarantees of Appendix A certify information-theoretic mechanisms—coverage rate, posterior concentration, gate failure probability, and discovery hazards—whereas the run logs record quality, time, and token surrogates. The alignment therefore runs at the level of *predicted signatures*: each theorem implies a falsifiable pattern in the reported measurements, checked against the tables and figures of Sections 4.3, 4.4 and 4.6. The four alignments below state the signature and its evidence; each closes with the conclusion the evidence supports.

(Theorem A.1) Per-round coverage advantage appears as the wall-clock and equal-time leads. The bound $(1-\chi_T)\underline{\gamma}_T \sum_j \mu_j$ predicts two signatures: the shared budget is exhausted sooner, and an equal-time read shows a quality lead. Both hold—COEVOLVE completes the shared 72-evaluation budget $1.03\times$–$2.85\times$ faster than PiEvo on all six tasks (mean $1.80\times$; Figure 4(a)), $\Delta\mathrm{SQ@}T_c$ averages $+8.3$ SQ points and is positive in 15 of 18 task–seed pairs (Figure 4(c)), and COEVOLVE attains the best average APD on both backbones (52.0 and 58.1; Table 2), the $\underline{\gamma}_T$ side of the bound.

> **Takeaway.** COEVOLVE exhausts the shared evaluation budget $1.03\times$–$2.85\times$ faster than PiEvo on all six tasks, leads by $+8.3$ SQ on average at equal time, and attains the widest exploration (APD) on both backbones.

(Theorem A.2) imports help only at their discounted weight. Since the theorem counts an import toward $N_{j,T}^{\mathrm{eff}}$ at weight $\alpha_{j\leftarrow i}$, two signatures must hold jointly: calibrated sharing raises attainment, and stripping the discount ($\alpha\equiv 1$) damages it. Both hold—the shared arm's best-of-seed SQ exceeds the Isolated arm on all six tasks ($+13.6$ MBO, $+13.3$ Promoter, $+2.5$ SPO, $+0.5$ NHO and TMC, and $+7.9$ AMP with $p<0.05$; Table 5), removing the discount collapses quality by 19.0 SQ, the largest knockout effect (Table 6), and the flow itself is real: 244 accepted imports, 205 net-unique, with uplift of $+15.4$ (MBO), $+25.4$ (AMP), and $+43.3$ (TMC) SQ points (Table 7).

> **Takeaway.** Calibrated sharing improves best-of-seed quality over the Isolated arm on all six tasks, while applying the same imports at full weight ($\alpha\equiv 1$) costs 19.0 SQ—the largest single-module degradation—and the accepted imports themselves carry observed uplift of up to $+43.3$ SQ.

(Theorem A.3) the gate behaves as a tail-probability control. The theorem bounds the *probability* of a harmful accepted import, so disabling the gate should fatten the lower tail, not move the mean. Removing the gate admits 137 imports against 42, yet the mean moves only -2.1 SQ while the standard deviation more than doubles ($4.6 \rightarrow 11.1$) with two of three seeds degraded (Table 6); attainment shows no monotone response to the margin η even though η is the strongest admission-volume actuator (Figure 9); and on Promoter the gate admits a single import across the three seeds, so its selectivity is genuine and the guarantee non-vacuous (Table 7).

> **Takeaway.** Removing it admits over $3\times$ more imports ($42 \rightarrow 137$) and more than doubles the outcome spread ($4.6 \rightarrow 11.1$) while shifting the mean by only -2.1 SQ—the signature of protection against rare harmful imports.

(Theorem A.4): the discovery rescue is gated by hazard diversity, not by K. The tail bound $\Pr(\tau_{\mathrm{disc}} > t) \le \prod_j (1-\lambda_j)^t$ predicts the largest advantage where a single branch's hazard λ is near zero: on SPO, systems without principle branching (Vanilla MAS, The AI Scientist v1) collapse to 0.0% under both backbones while COEVOLVE sustains 15.3%–30.0%, $1.5\times$–$2.8\times$ PiEvo (10.4%–10.6%; Table 1). The branch-count sweep confirms the caveat of Remark B.5: at fixed budget the response to K is an inverted-U on MBO peaking at $K=3$, flat on AMP, and a plateau matching the PiEvo reference on TMC—landscape-gated, not K-gated (Figure 8).

> **Takeaway.** COEVOLVE sustains 15.3%–30.0% on SPO, where single-branch systems stall at 0.0%, and the response to branch count at fixed budget peaks at $K=3$—the gain comes from diversified discovery hazards, not from running more branches.

D Implementation Details of the Coordination Loop

CoEvolve instantiates the three quantities of Section 3.2—the trust score $\rho_{i,t}$, the transfer cost $\Delta^{\text{imp}}_{j,t}$, and the value estimate $\widehat{V}_{j,t}$—and assembles them into the per-round loop of Algorithm 1:

Trust $\rho_{i,t} = 1/(1+\bar{e}_{i,t})$. We set $\bar{e}_{i,t}$ as the source's mean absolute standardized GP residual (Pu et al., 2026), which discounts sources that mispredict their own measurements. The $s_{ij,t}$ scores context transferability to target j from task identity, the declared shared observation model, and subspace compatibility, and $v_{i,t} = 1$ after independent target replication and 0.5 otherwise. Missing estimates fail closed at zero. The target snapshot supplies the predictive entries $f_{j,P}(h)$ and $\sigma^2_{j,P}(h)$ for every supported hypothesis and principle.

Cost $\Delta^{\text{imp}}_{j,t}(C)$. It prices the transfer mechanics as constants normalized to the entropy scale: parsing c_{read}, verification of unreplicated records c_{verify}, posterior fitting of novel records c_{fit}, and context shift $c_{\text{shift}} = (1 - s_{ij,t})c^{\max}_{\text{shift}}$. Context shift is the only state-dependent term; driven by the same setting-match score as the discount, it makes a less compatible source simultaneously downweighted in α and more expensive to import.

Value $V_{j,t}(C)$. It is a conditional expectation over residual randomness; the core evaluates it exactly over the finite principle universe by cloning the target posterior, applying (h, y) at weight α, and measuring the induced entropy change. To align this information score with the task's maximization objective, it is weighted by the source outcome's empirical-rank relevance $w_{\text{rel}}(y) \in [w_{\min}, 1]$:

$$\widehat{V}_{j,t}(C) = w_{\text{rel}}(y)\big[H(p_{j,t}) - H(\widetilde{p}^{+}_{j,t})\big] - \lambda\widehat{\Delta}^{\text{imp}}_{j,t}(C),$$

where $\widetilde{p}^{+}_{j,t}$ is the hypothetical clone, committed only if the candidate passes the gate. $\widehat{V}_{j,t}(C)$ departs from Eq. (1) only through w_{rel}, the cost constants, and exact enumeration over the finite $\widetilde{\mathcal{P}}$ in place of the expectation.

Redundancy cap $R_{\max}$. Routing deduplicates each branch's per-round admitted set by text: after the gate and ranking, candidates are admitted greedily in ranked order, and a candidate is dropped if its text has token-Jaccard similarity $J(A,B) = {}^{|A\cap B|}/_{|A\cup B|} \geq R_{\max}$ with an already-admitted one, keeping only the highest-ranked near-duplicate per round (default $R_{\max}$=0.3; $R_{\max}$=1 disables the cap, admitting only exact-text duplicates—the knock-out arm of Table 6). This is distinct from the evidence pool's exact-hash deduplication: the cap is an approximate, per-round, per-branch filter at routing time.

Monitoring. The core logs accepted and rejected candidates, discount components, duplicate rates, replication outcomes, predictive-density checks, posterior entropy changes, and coordination time; rejected records remain in the pool without altering target posteriors.

Gaussian realization of the discounted update. Under Gaussian predictive models with fixed observation noise and a sample weight acting as a precision multiplier, the discounted update of Eq. (5) contributes $-\frac{\alpha_{j\leftarrow i,t}(h)}{2\sigma^2_{\text{obs}}}\big(y - f_{j,P}(h)\big)^2 + \text{const}$ to the target log posterior: an accepted import is a local observation with variance $\sigma^2_{\text{obs}}/\alpha$.

Default constants and reproducibility. Table 4 lists every constant of the coordination loop. The relevance weight is the source outcome's mid-rank percentile within the source branch's observed outcomes, lifted to $[w_{\min}, 1]$: $w_{\text{rel}}(y) = w_{\min} + (1 - w_{\min})\,\text{pct}(y)$ with $\text{pct}(y) = \big(\#\{y' < y\} + {}^1/_2 \max(\#\{y' = y\} - 1, 0)\big)/(n-1)$ and $w_{\min} = 0.1$; a candidate implausible under the entire target posterior (log predictive density below -25) is rejected outright. The per-candidate confidence budget follows the alpha-spending schedule $\delta_m = 6\delta/(\pi^2 m^2)$ with total budget $\delta = 0.05$ (Remark B.3). The per-task principle universe $\widetilde{\mathcal{P}}$, prior π_0, and GP outcome models follow PiEvo (Pu et al., 2026) unchanged. All runs use three fixed seeds per cell; rejected candidates and surrogate or endpoint failures score 0.0 and are excluded from trajectories as infrastructure events (Appendix I). Code, configuration files, and per-run logs will be released.

Algorithm 1 assembles these quantities into one coordination round.

Table 4: **Default constants of the coordination core.** All values are fixed across tasks, backbones, and ablations unless the ablation varies them explicitly.

Symbol	Default	Role
η	0.0	safe-VoI gate margin (Eq. (2))
B	3	routing quota per branch per round
$R_{\max}$	0.3	redundancy cap (token-Jaccard)
λ	1.0	import cost weight (Eq. (1))
δ	0.05	total false-import budget (alpha-spending)
c_{read}	5×10^{-4}	parsing cost
c_{verify}	2.5×10^{-4}	verification cost (unreplicated records)
c_{fit}	2.5×10^{-4}	posterior fitting cost
$c_{\text{shift}}^{\max}$	2×10^{-2}	maximal context-shift surcharge
$w_{\min}$	0.1	relevance floor of $w_{\text{rel}}(y)$
ε	10^{-9}	IDS ratio denominator guard (Eq. (3))

Algorithm 1 CoEvolve coordination core (one round t)

Require: branches $\{1, \dots, K\}$ with posteriors $\{p_{j,t-1}\}$, evidence pool H_{t-1}^{sys}; margin η, routing quota B
Ensure: updated posteriors $\{p_{j,t}\}$, extended pool H_t^{sys}, logged decisions
1: **Collect:** each branch i publishes its round-t tested pairs as candidate records $C = (i, h, y, E_i, m)$;
2: **Merge:** validate, deduplicate, and append records to the append-only pool; no posterior is touched;
3: **for all** targets j and pooled records C from sources $i \neq j$ **do**
4: $\alpha_{j \leftarrow i,t}(h) \leftarrow \text{clip}_{[0,1]}(\rho_{i,t}\, s_{ij,t}\, v_{i,t})$ {Eq. (4)}
5: clone $p_{j,t-1}$, enumerate the discounted update, set $\widehat{V}_{j,t}(C)$ with $r = 0$ {Eq. (1)}
6: $\Delta_{j,t}^{\text{imp}}(C) \leftarrow c_{\text{read}} + c_{\text{verify}} + c_{\text{fit}} + (1 - s_{ij,t})\, c_{\text{shift}}^{\max}$
7: **end for**
8: **Route:** admit C to j only if $\widehat{V}_{j,t}(C) - r > \eta$; rank admitted records by $\Delta_{j,t}^{\text{imp}}(C)^2 / \max\{\widehat{V}_{j,t}(C) - r, 0\} + \varepsilon$ and keep the top B {Eqs. (2)–(3)}
9: **for all** targets j **do**
10: **Inject:** multiply $p_{j,t}$ by $q_{j \leftarrow i}(y \mid h, P)^{\alpha}$ per admitted import; record the decision {Eq. (5)}
11: **end for**
12: **Monitor:** log diversity and audit diagnostics feeding the scoring of round $t{+}1$.

E Ablation of Hyperparameters in CoEvolve

Branch-count ablation. As shown in Figure 8, the response of CoEvolve tends to be landscape-dependent: an inverted-U on MBO peaking at $K{=}3$, flat on AMP, and on TMC a rise into a plateau that matches or exceeds the PiEvo reference from $K{=}3$ onward.

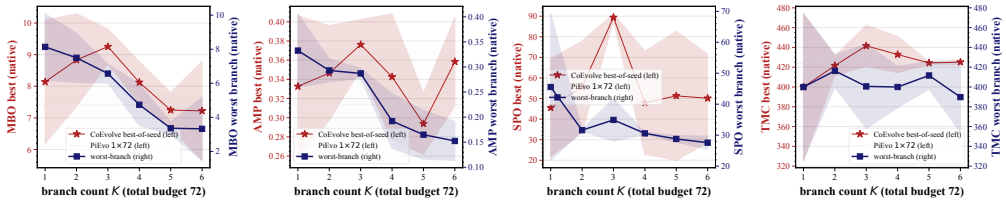


Figure 8: **Branch-count ablation.** In $K{=}5$, we use unequal per-branch budgets $\{15, 15, 14, 14, 14\}$ summing to the same total of 72. The gray dashed line and band mark PiEvo (Pu et al., 2026).

Hyperparameter sensitivity. We ablate the safe-VoI gate margin η, the redundancy cap $R_{\max}$ (extended to the fully-disabled point $R_{\max}{=}1$ from Table 6), the import cost weight λ, and the per-round routing quota in Figure 9. The best-of-seed SQ peaks at the default of $R_{\max}$ and the routing quota, sits on a plateau spanning $\lambda{\in}[1, 2]$, and shows no monotone response to η. This is consistent

with the gate's role as protection against *rare* harmful imports, not an average-case quality lever (Table 6).

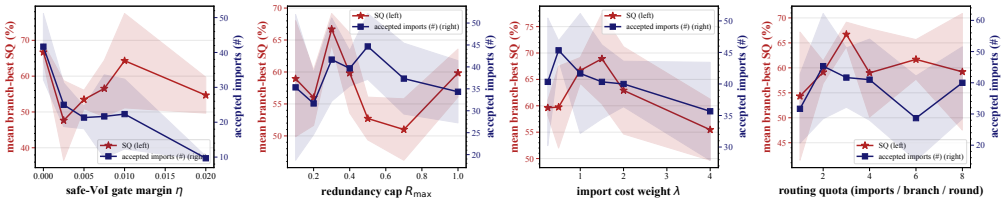


Figure 9: **Hyperparameter sensitivity.** Each panel varies exactly one coordination hyperparameter and pairs the *best-of-seed SQ* with the accepted-import volume.

F Ablation of Evidence Transfer in CoEvolve

Table 5: **Ablation of Evidence transfer.** Δ is the difference of arm means with * marking Welch's t significance at $p < 0.05$; purple favors sharing. Promoter's Isolated arm runs the current $T{=}1.1$ protocol.

	Worst-branch SQ (%)			Best-of-seed SQ (%)			
Task	CoEvolve shared	Isolated	Δ	CoEvolve shared	Isolated	Independent	Δ
MBO	54.8 ± 4.7	45.3 ± 10.0	$+9.4$	77.1 ± 4.6	63.5 ± 7.9	53.6 ± 13.6	$+13.6$
NHO	90.8 ± 2.0	89.9 ± 2.7	$+0.9$	94.9 ± 0.1	94.4 ± 0.6	95.0 ± 0.0	$+0.5$
SPO	11.7 ± 2.3	10.4 ± 0.4	$+1.3$	30.0 ± 0.6	27.5 ± 4.9	25.5 ± 4.2	$+2.5$
TMC	80.1 ± 8.5	84.8 ± 0.2	-4.7	88.3 ± 4.2	87.8 ± 4.5	87.7 ± 4.6	$+0.5$
AMP	28.7 ± 1.0	19.8 ± 1.4	$+8.9^\star$	37.6 ± 2.6	29.7 ± 2.0	32.3 ± 0.6	$+7.9^\star$
Promoter	56.2 ± 14.9	52.4 ± 4.8	$+3.8$	86.4 ± 0.4	73.1 ± 10.9	70.5 ± 13.1	$+13.3$

Per-task view of evidence transfer. Table 5 expands Figure 6 (Section 4.6) into exact numbers: best-of-seed SQ for all three arms (shared, Isolated, Independent) and worst-branch SQ for the two coordinated arms, with Δ the shared-minus-Isolated margin. Figure 10 shows the same comparison with mean branch-best SQ, to see whether sharing lifts the whole branch population rather than only the champion lineage.

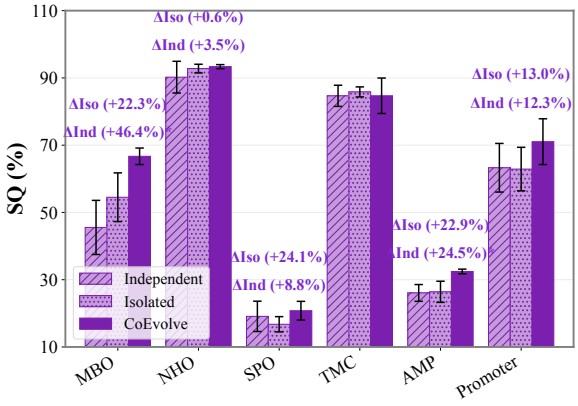


Figure 10: **Marginal value of evidence sharing at the systemic level.**

On two tasks the Independent–Isolated–CoEvolve ordering is not monotone. On AMP the Independent arm (32.3 ± 0.6) edges out Isolated (29.7 ± 2.0) within the seed noise, while the shared arm remains clearly best (37.6 ± 2.6; Welch's t, $p{<}0.05$): coordination without sharing neither helps nor hurts there, and the entire gain comes from the evidence flow itself.

NHO is the opposite extreme: a compact, exploitation-type landscape (a seven-dimensional continuous search space with a smooth surrogate) in which every arm saturates the same ceiling (94.4–95.0 SQ), so little complementary coverage remains to transfer and the Independent arm nominally matches CoEvolve (95.0 vs 94.9, within noise).

On NHO, coordination's remaining benefit is convergence speed: in the dedicated NHO campaign both arms (CoEvolve and Independent) reach the identical ceiling (g-factor 1.90) but CoEvolve converges $1.5\times$ faster in wall-clock (120 vs 184 minutes), and the same signature appears on the Terra backbone as the $+10.5$ equal-time gap of Figure 4 (c). On such compact landscapes, collaboration shows up as efficiency.

Ablation of modules in CoEvolve. Table 6 gives the exact numbers behind the module ablation of Section 4.6 (Figure 7): each row disables exactly one transfer safeguard, and the full-system reference row is the main-table arm. Off discounting or redundancy control degrades both portfolio and worst-branch quality. Disabling the VoI gate admits over $3\times$ as many imports and degrades two of three seeds with markedly higher variance. This is consistent with the gate protecting against rare harmful imports, not average-case ones (see Section 3).

Table 6: **Ablation of modules in CoEvolve.**

Arm	knocked-out module	SQ	worst-br.	ΔSQ	imports	$\bar{\alpha}$
CoEvolve (full)	all modules on (main-table arm)	77.1 ± 4.6	54.8 ± 4.7	–	42	0.23
− VoI gate	safe-VoI gate off ($\eta \to -\infty$): every routed import accepted	75.0 ± 11.1	52.8 ± 15.3	−2.1	137	0.24
− discounting	$\alpha \equiv 1$: imports applied at full weight	58.1 ± 4.4	43.3 ± 9.5	−19.0	58	1.00
− redundancy control	$R_{\max}=1$: redundancy cap disabled	67.5 ± 4.4	51.5 ± 7.8	−9.6	34	0.23
− sharing (Isolated)	coordination on, zero evidence flow	63.5 ± 7.9	45.3 ± 10.0	−13.6	0	–

Table 7 reports the transfer-effects analysis of Section 4.6: the per-branch cost of budget splitting and the import-level quantities that show whether cross-branch evidence sharing actually flowed, was deduplicated, and is associated with observed quality gains.

Table 7: **Ablation of evidence transfer with net value.** *accepted* = evidence-node×branch applications; *net unique* = distinct evidence hashes delivered; *dedup* = deliveries per unique evidence (delivery redundancy); *uplift* = pooled-best SQ gain over evaluation windows containing a delivery, averaged over seeds: a window-level, associational statistic (deliveries are value-selected, so uplift does not isolate a causal transfer effect), unlike the branch-level, immediate *inert* rate, with which it need not agree; *inert* = fraction of accepted imports with zero immediate running-best change. TMC and Promoter aggregate only 2 and 1 accepted imports. All import columns aggregate the three main-table seeds.

	Per-branch SQ (%)			Imports				
Task	CoEvolve worst-branch	PiEvo	Δ	accepted	net unique	**dedup**	**uplift (SQ pts)**	**inert**
MBO	54.8 ± 4.7	61.8 ± 2.7	−7.0	120	99	1.26×	+15.41	80%
NHO	90.8 ± 2.0	94.9 ± 0.1	−4.1	19	17	1.29×	+0.15	89%
SPO	11.7 ± 2.3	10.6 ± 0.0	+1.1	6	6	1.00×	+3.64	67%
TMC	80.1 ± 8.5	84.7 ± 0.0	−4.6	2	2	2.50×	+43.29	100%
AMP	28.7 ± 1.0	31.4 ± 5.8	−2.7	96	80	1.26×	+25.41	81%
Promoter	56.2 ± 14.9	54.4 ± 6.3	+1.8	1	1	1.00×	+0.00	100%
Total / Avg.				244	205			

G Statistical Analysis

All experiments run three seeds per cell, which limits the power of any single per-task test. We therefore (i) base the headline comparisons on aggregate paired tests over the 18 task–seed pairs of the GPT-5.6-Terra backbone, where per-seed trajectories are logged for both CoEvolve and PiEvo, and (ii) report per-task tests for completeness.

Paired per-seed tests (CoEvolve vs PiEvo, Terra backbone). Table 8 lists the per-seed SQ differences. Aggregating across tasks, the paired gain is positive in 15 of the 17 non-tied pairs (two-sided sign test p=0.002; counting the exact tie against CoEvolve, p=0.008; Wilcoxon signed-rank p=0.001). The wall-clock speedup exceeds 1× in 17 of 18 pairs (sign test $p=1.5\times10^{-4}$), and the equal-time gap $\Delta\mathrm{SQ@}T_c$ is positive in 15 of 17 non-tied pairs ($p<0.01$; Wilcoxon $p=8\times10^{-4}$).

Table 8: **Paired per-seed SQ differences, CoEvolve minus PiEvo (GPT-5.6-Terra).** Two-sided paired t test per task over three seeds; the aggregate row pools all 18 task–seed pairs.

Task	Per-seed SQ differences	Paired t p-value
MBO	+5.2, +6.6, +39.3	0.266
NHO	+9.4, +6.8, +6.9	0.012
SPO	−0.1, +14.5, +0.3	0.415
TMC	+2.5, 0.0, +1.8	0.194
AMP	+2.0, +5.0, +5.0	0.057
Promoter	+2.9, −2.8, +8.0	0.481
Aggregate (18 pairs)	15 positive, 2 negative, 1 tie	sign $p<0.01$; Wilcoxon p=0.001

Summary-statistic Welch tests. Table 9 reports two-sided Welch t tests computed from the rounded mean ± std of Table 1 (n=3 per cell), for guidance; the underlying per-seed values accompany the code release. Against PiEvo, 4 of 12 task–backbone pairs reach $p<0.05$; under a Bonferroni correction over the 12 pairs (α=0.0042), SPO (Gemma, $p=3\times10^{-4}$) and NHO (Terra, p=0.004) survive. We therefore state the closed-world claim at the level of best average SQ with the aggregate paired test above, not per-task superiority on every task.

Table 9: **Welch t tests on final SQ from the summary statistics of Table 1** (two-sided, n=3 per cell, computed from rounded means and stds). * $p<0.05$; ** survives Bonferroni over the 12 pairs.

	vs PiEvo		vs InternAgent-1.5	
Task	Gemma	Terra	Gemma	Terra
MBO	$0.013^\star$	0.263	$0.005^\star$	0.453
NHO	1.000	$0.004^{\star\star}$	0.300	0.386
SPO	$3\times10^{-4\,\star\star}$	0.414	$2\times10^{-4\,\star\star}$	0.501
TMC	0.276	0.184	$0.029^\star$	0.445
AMP	0.197	0.058	$0.011^\star$	0.085
Promoter	$0.012^\star$	0.526	0.660	0.324

Harness regime. On each of the five autoresearch tasks, CoEvolve's mean improvement is significantly above the anchor (one-sample t against zero: ALDE p=0.003, Deconv $p=2\times10^{-4}$, InvScat p=0.022, D2D $p<10^{-4}$, MolEdit p=0.006). Pairwise differences against individual reference arms are directionally uniform but underpowered at three seeds (against PiEvo, p ranges 0.07 to 0.40); the interval argument of Section 4.5, the worst CoEvolve case matching or exceeding every other arm's best case on all five tasks, is the primary evidence there.

H HARNESS-REGIME EXPERIMENT DESIGN

H.1 TASKS AND BASELINES

Tasks and score. This regime uses a second suite of five research-workflow tasks sampled from AutoresearchEval (Fei et al., 2026), disjoint from the closed-world optimization benchmarks of Section 4.1: each task asks for a *method artifact*: an optimizer, a pipeline, or a predictor, not a single candidate, and each ships a published SOTA anchor and a sealed grader running on held-out data:

1. **protein-landscape active learning** (ALDE), submitting an optimizer that allocates a 480-measurement screening budget over two four-site protein fitness landscapes;
2. **bulk deconvolution** (Deconv), predicting tumor-microenvironment cell-type proportions for real breast-cancer bulk RNA-seq from a disjoint single-cell reference;
3. **inverse scattering** (InvScat), reconstructing dielectric maps from complex scattered fields under two receiver-density instances;
4. **D2D scheduling** (D2D), activating device-to-device links in a geographic interference network to maximize sum rate; and
5. **molecule editing** (MolEdit), editing molecular graphs under chemistry-validity constraints to optimize penalized logP, QED, and similarity-constrained objectives.

The grader aggregates per-instance improvements over the published anchor; we report $\Delta \times 100$, where $\Delta > 0$ means the best submitted artifact surpasses the anchor and $\Delta < 0$ falls short of it.

Budget contract and outcome source. The evaluation budget is enforced server-side: the evaluation service counts every request, scored or not, and rejects calls beyond the evaluation budget. All outcomes are taken from the grader's responses; numbers self-reported by the agents are discarded.

Pre-registered task exploratiVeness. Before any run, we ordered the five tasks by an ordinal task-exploratiVeness index (TEI) read from the published task artifacts alone, including task text, anchor, and grader, never run behavior: *free-oracle throughput* tasks (D2D, MolEdit), where a free local oracle governs the score, lowest; *recipe-saturated* tasks (ALDE), where an established recipe leaves little headroom above the anchor; and *open method spaces* (Deconv, InvScat), where no canonical pipeline dominates, highest. Appendix H.5 tests whether the collaboration margin follows this pre-registered ordering.

Harness substrate. All methods run on one substrate: a containerized coding agent (with `DeepSeek-v4-Flash-0731` model as backbone) with read-only task data, an ephemeral workspace, and a scientific Python environment. The executor endpoint is pinned and verified at launch, so no method can fall back to a different model, and every method sees the identical task description. The methods differ only in who decides the next experiment:

(a) **Agent-only** (Claude Code): the agent receives the task once and thereafter acts autonomously; the driver only harvests submissions, nudges toward budget completion, and restarts stalled sessions;

(b) **Agent-only** (Codex): the same protocol under a second coding agent, testing whether conclusions depend on one particular harness;

(c) **Agent-only** (Arbor (Jin et al., 2026)): an external autonomous research-harness method that self-verifies against a free local oracle.

(d) **PiEvo** (Pu et al., 2026): the PiEvo strategy layer runs driver-side and injects one stage-guidance message (principle, hypothesis, or experiment) per turn while the same agent executes it in the container; the strategy layer is identical across arms (d) and (e), and its model is distinct from the executor model;

(e) **COEVOLVE**: $K{=}3$ branches of arm (d), under a server-side-enforced total of 72 evaluations per run.

H.2 MEASUREMENTS

Wall-clock and token accounting. Wall-clock time spans a run's first to last scored evaluation. Tokens are parsed from the transcripts (uuid-deduplicated) in four channels; *billable* = input + output

+ cache-creation, and the *all-in* count adds cache-read, we report both. Table 10 gives the per-task means; CoEvolve spends 1.55× PiEvo's billable tokens on average, with ALDE below 1× because its branches converge early and exhaust the budget sooner.

Table 10: **Token accounting in the harness regime** (three-seed means, millions of tokens).

	Billable			All-in	
Task	CoEvolve	PiEvo	**ratio**	CoEvolve	PiEvo
ALDE	7.5	14.5	0.52×	350.4	339.1
Deconv	95.4	37.6	2.53×	181.5	90.3
InvScat	89.9	46.3	1.94×	363.8	136.6
D2D	11.8	8.8	1.35×	331.4	192.4
MolEdit	23.3	16.6	1.40×	628.6	237.1
Mean ratio			1.55×		

Normalization for merged curves. Cross-task anytime plots min–max normalize scores per task, with 0 at the task anchor and 1 at the best final value, so a point on the merged curve reads as the fraction of the task's observed best attained at a given wall-clock or token spend.

H.3 Efficiency in the Harness Regime

Equal-time quality gap compared to PiEvo (Pu et al., 2026). $\Delta@T_c$ reads every PiEvo run's running best at each CoEvolve run's completion time T_c from per-evaluation timestamps and averages the difference. It is positive on all five tasks: ALDE +8.9%, Deconv +31.2%, InvScat +18.4%, D2D +0.1%, MolEdit +3.0% points, mean +12.3%, and is never smaller than the budget-end gap, as shown in Figure 11.

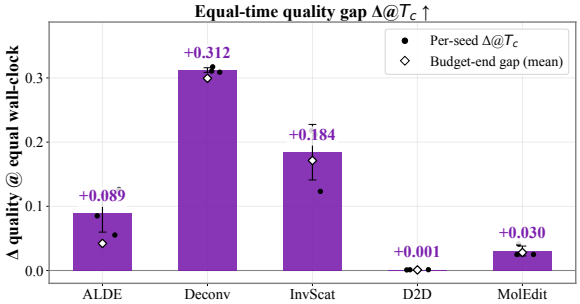


Figure 11: **Equal-time quality gap in the harness regime.** Black dots: per-seed values; white diamonds: the budget-end gap. The equal-time reading is positive on all five tasks and never below the budget-end gap.

Time-to-threshold and wall-clock compared with PiEvo (Pu et al., 2026). Let q^* be PiEvo's three-seed mean final quality on a task and TTT the hours until a run's running best first reaches q^*. All fifteen CoEvolve runs reach q^* (zero censoring). Against PiEvo's own time to reach q^* (self-TTT), the per-task median ratio $T_{\mathrm{TTT}}^{\mathrm{PiEvo}}/T_{\mathrm{TTT}}^{\mathrm{CoEvolve}}$ is 0.9/1.2/0.8/1.3/1.9× on ALDE/Deconv/InvScat/D2D/MolEdit (Table 11). CoEvolve is faster on D2D and MolEdit, tied on Deconv, slightly slower on ALDE, and slowest on InvScat (15.2 vs 11.9 hours), where its branches remain below the anchor for most of the budget before a late jump, the breadth-first pattern of Appendix H.4.

H.4 Trajectory Pattern

Breadth-first, then lead in later stage. CoEvolve leads at the later stage on all five tasks, but not throughout, as shown in Figure 12. Per-evaluation AUOC agrees: CoEvolve is best on ALDE (the only positive value, +0.006), Deconv (0.212, 4× the runner-up), and InvScat (highest at −0.090), ties PiEvo on D2D (0.1268 vs 0.1272, within seed noise, where Arbor is best at 0.1447), and trails PiEvo on MolEdit.

Table 11: **Wall-clock conversion speed per task in the harness regime.** q^* = PiEvo's mean final quality (raw anchor-relative Δ; multiply by 100 for the scale of Table 3); T = end-to-end wall-clock (h); TTT = per-run hours until the run's running best first reaches q^*. CoEvolve's self-TTT ratio against PiEvo is $0.8\times$–$1.9\times$, and its end-to-end wall-clock is shorter on three of five tasks.

Task	q^*	T_{PiEvo}	T_{CoEvolve}	$T_{\text{PiEvo}}/T_{\text{CoEvolve}}$	$\text{TTT}_{\text{PiEvo}}$	$\text{TTT}_{\text{CoEvolve}}$	ratio
ALDE	+0.024	32.5	20.8	1.56×	6.64	7.67	0.9×
Deconv	+0.154	22.4	12.1	1.85×	1.38	1.11	1.2×
InvScat	-0.053	47.7	59.7	0.80×	11.87	15.17	0.8×
D2D	+0.154	31.7	38.0	0.83×	17.05	13.45	1.3×
MolEdit	+0.013	45.9	30.8	1.49×	11.52	6.03	1.9×

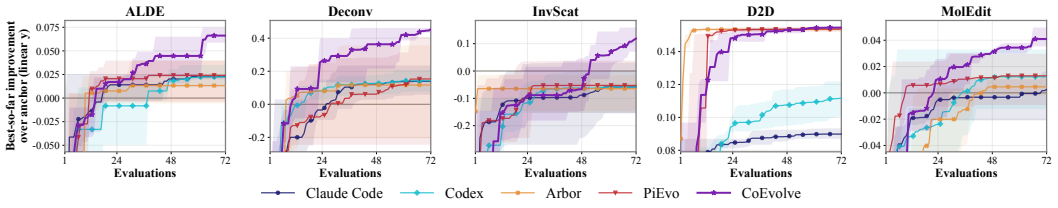


Figure 12: **Running-best Δ over the 72-evaluation budget**.

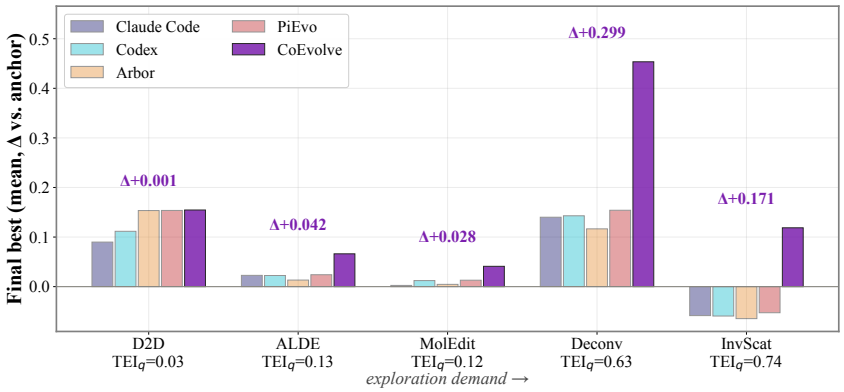
H.5 Task Exploratiiveness Analysis

The pre-registered ordering (Appendix H) predicts that the collaboration margin grows with task exploratiiveness.

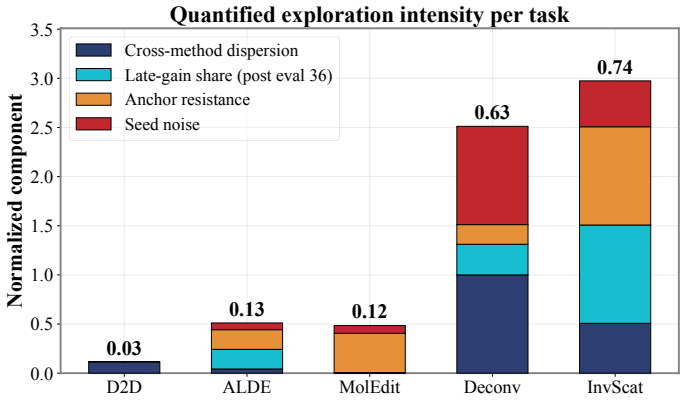
Figure 13(a) orders the five tasks by increasing exploration demand and compares each arm's final Δ: the four reference arms stay bunched on the free-oracle task (D2D) and collapse toward or below the anchor as demand grows—all four are negative on InvScat—while CoEvolve alone stays positive throughout, peaking on the open method space (Deconv) and remaining the unique positive arm on InvScat. The margin over the runner-up follows the predicted order: $+0.1/+2.8$ points on the free-oracle tasks (D2D/MolEdit), $+4.2$ on the recipe-saturated task (ALDE), and $+30.0/+17.2$ on the open method spaces (Deconv/InvScat).

Figure 13(b) quantifies the same ordering from run behavior—four proxies pooled over all runs per task, min–max normalized across tasks and averaged—yielding TEI_q of $0.03/0.13/0.12/0.63/0.74$. This quantification reads run behavior and is therefore an illustration of, not a replacement for, the pre-registered index.

Remark H.1. *On free-oracle tasks, as shown in Table 3, throughput converts directly into score and coordination cannot add throughput, whereas open method spaces reward crossing into a better method family, which is what evidence transfer supplies.*



(a) Final Δ by task, ordered by exploration demand.



(b) Quantified TEI from four behavioral proxies (illustration).

Figure 13: **Task exploratiVeness moderates the collaboration margin.**

I Benchmark Details

I.1 Design Principles

The benchmark suite serves a specific purpose: comparing budget-constrained search architectures. Two design requirements follow.

Every task must be searchable, not memorizable. A comparison between arms that spend the same 72-query surrogate budget is interpretable only if no arm can match search-attained quality by recalling a textbook answer from parametric memory. The suite enforces this by construction, through one of three mechanisms per task:

a. *Banned-answer tasks* (SPO, Promoter, AMP): an admission gate excludes the canonical high-scoring solution family—the record and textbook superconducting backbones in SPO, the yeast GRF/GC-box promoter architecture in Promoter, cationic amphipathic peptides in AMP—so the memorizable answer is unreachable, not merely discouraged.

b. *Surrogate-defined optima* (NHO, MBO): the objective is the argmax of a specific surrogate's response surface, which appears in no publication and therefore cannot be recalled; domain knowledge supplies priors (drug-likeness heuristics, chiral-geometry intuition) but not the answer, and MBO's gate additionally closes the one channel by which memorized motifs could inflate the score without genuine binding (PAINS assay-interference artifacts).

c. *Pool-computed optimum* (TMC): the optimum is an explicit computation over a fixed 49-ligand pool under charge neutrality, not a named entity that can be recited. The residual spaces remain scientifically non-trivial: the motif-free Promoter landscape still spans $\sim$7.7–14.7 of 17, and the admissible SPO families retain electron-doped basins at 80–95 K (Appendices I.5 and I.7).

Every score must be an admission-gated physical property. All tasks couple the LLM proposal interface to the domain surrogate of the originating study (regression models, semi-empirical physical models, or universal machine-learning force fields; cited per task below). Candidates enter the trajectory only after passing an anti-reward-hacking gate whose components apply as relevant per task:

(a) format and chemical validity;

(b) the task's counterfactual constraint;

(c) sequence complexity on the sequence-design tasks;

(d) a physical plausibility ceiling (oracle clamping)

As an engineering control, it is expected to close the reward-hacking channels we identified for each task, and the per-task subsections enumerate each channel together with its check. The accounting distinguishes the two failure modes because they are different events: a *gate rejection* is a search outcome, so it consumes one evaluation of the fixed budget, scores 0.0, and can never become the running best, whereas a *surrogate failure* is an infrastructure event, likewise zero-scored and excluded from best-of-seed computation.

Inadmissible proposals are therefore paid for in opportunity cost and a run that admits no valid candidate scores 0.0 (e.g., the zero-score cells of Table 1). Finally, the six tasks are chosen to span distinct search-space types (Pu et al., 2026): continuous (NHO), molecular graph (MBO), peptide sequence (AMP), nucleotide sequence (Promoter), discrete combinatorial (TMC), and stoichiometric formula (SPO), and distinct landscape regimes, from the near-ceiling continuous task (NHO) through quality discrimination (MBO) and counterfactual boundary search (AMP, Promoter) to the bimodal SPO landscape. Table 12 summarizes the six tasks; Appendix I.2–Appendix I.7 give the scientific definition, search space, surrogate, and exact task specification for each.

Reference-scale design. Normalizing raw scores by a dataset statistic (e.g. a training-set maximum) would make SQ a moving target that changes with the surrogate version, and normalizing by any method's best would destroy cross-paper comparability. We therefore anchor each task's scale endpoints in domain knowledge, in three tiers:

(a) *Mathematical bounds of the score.* NHO's g-factor ceiling of 2, AMP's probability bound 1, and Promoter's expected value over the 18 ordinal expression bins, bounded by 17;

Table 12: **Benchmark summary.** All tasks maximize the gated surrogate score. "Anti-memorization gate" = the per-task mechanism that prevents reaching the optimum from parametric memory (the three mechanisms of Appendix I.1); scores in native units. **Ref. scale** $= [y_{\text{lo}}, y_{\text{hi}}]$, the domain-defined absolute bounds of the task's score used to normalize SQ (Section 4.2); they are fixed properties of the benchmark (physical laws, scale endpoints, or the originating benchmark's theoretical maximum), never derived from any method's runs. On NHO, AMP, Promoter, and TMC the scale is a mathematical or benchmark-defined bound, so SQ $\leq 100\%$ holds by construction; on MBO and SPO it is a domain-anchored reference far above every observed run (best observed: 77.1 and 30.0 SQ), so the bound holds empirically with wide margin.

Task	Domain	Search space	Surrogate score	Anti-memorization gate	Ref. scale
NHO	nanophotonics	7-dim continuous	g-factor $\in [0, 2]$	surrogate-defined optimum	[0, 2]
MBO	bio-chemistry	SMILES (drug-like)	pChEMBL	surrogate-defined optimum; Lipinski + PAINS	[0, 12]
AMP	biology	peptide 12–50 aa	AMP probability $\in [0, 1]$	net charge ≤ 0 (cationic recipe banned)	[0, 1]
Promoter	biology	50-nt DNA	expression bin $\in [0, 17]$	yeast GRF + GC-box banned	[0, 17]
TMC	chemistry	4-of-49 ligand comb.	polarizability (≤ 500)	pool-computed optimum; charge neutrality, distinct ligands	[0, 500]
SPO	materials	cuprate formulas	T_c (K)	record/textbook backbones banned (Hg/Tl, BSCCO, YBCO)	[0, 298.15]

(b) *The originating benchmark's theoretical maximum.* TMC's 500, enforced at evaluation time by oracle clamping;

(c) *Domain-anchored targets* that are references rather than bounds. MBO's pChEMBL 12 (femtomolar binding, the practical affinity ceiling of the ChEMBL scale for reversible drug-like molecules (Gaulton et al., 2011)) and SPO's 298.15 K (the field's room-temperature target)

Consequently SQ $\leq 100\%$ holds by construction on tiers (a)–(b) and empirically, with wide margin, on tier (c): the tier-(c) anchors sit far beyond every observed run (best observed: 77.1 and 30.0 SQ).

Table 13: **Final quality in native units.** Best-of-seed SQ means of Table 1 multiplied by the reference scale of Table 12.

		Gemma-4-31B-IT		GPT-5.6-Terra	
Task	**Unit (scale)**	CoEvolve	PiEvo	CoEvolve	PiEvo
NHO	g-factor (2)	1.898	1.898	1.878	1.724
MBO	pChEMBL (12)	9.25	7.42	8.56	6.52
AMP	probability (1)	0.376	0.314	0.277	0.238
Promoter	expression bin (17)	14.69	9.25	13.96	13.50
TMC	polarizability (500)	441.5	423.5	465.0	457.5
SPO	T_c in K (298.15)	89.4	31.6	45.6	31.0

I.2 Nanohelix Optical-Chirality Optimization (NHO)

The NHO task (Pu et al., 2025) optimizes the geometry of a helical metamaterial nanostructure and its optical excitation conditions to maximize circular-dichroic response.

Scientific definition. The quantity of interest is the dissymmetry (g-) factor, the normalized difference between left- and right-circularly polarized absorption, evaluated by an electromagnetic simulation surrogate.

Search space. Unlike the four-dimensional variant in PiFlow, our instantiation uses the full **seven-dimensional** parameterization: four structural parameters of the helix (fiber radius, helix radius, number of turns, pitch) plus three excitation parameters (wavelength, principal optical axis, propagation direction). The inclusion of the excitation parameters matters physically: the chiral response of a helix is strongly anisotropic (Savinov et al., 2016; Rauschenbeutel, 2017; Faniayeu et al., 2020), and the global optimum $g \approx 2$ lies on a narrow manifold requiring aligned excitation, so the task is a genuine coupled structure–environment optimization, not geometry tuning alone. The space is bounded to the documented parameter envelope, with a ± 0.1 anti-micro-tune gate against re-querying near-duplicates.

Anti-memorization mechanism (surrogate-defined optimum). NHO is the suite's only task with a purely continuous search space, and the only one without a banned-family constraint; it needs none, because its optimum is defined by the simulator's response surface and appears in no publication.

Suite role. NHO anchors the suite's "near-ceiling" regime (best-of-seed ≈ 1.90 of the theoretical 2.0).

Reference scale. $[0, 2]$; the dissymmetry factor satisfies $|g| \leq 2$ by definition, so 2.0 is a mathematical ceiling, not an empirical one.

Nanohelix Structure Optimization

[Objective] Find the parameter combination that maximizes the optical chirality of a helical nanostructure, described by its g-factor value $\in [0, 2]$.

[Parameters for Hypothesizing]

- **fiber-radius** (20–60 nm): radius of the fiber/wire forming the helix.
- **helix-radius** (20–90 nm): distance from the central axis to the center of the helical path.
- **n-turns** (float, 3.0–10.0): number of turns.
- **pitch length** (60–200 nm): axial distance between adjacent turns.
- **wavelength** (float, 400.0–800.0 nm): incident wavelength at which the g-factor is evaluated (highest chirality often occurs at 400–500 nm).
- **x_y_z** (integer 0/1/2): principal axis of the light source.
- **direction** (integer 0/1): propagation polarity (forward/backward).

[Target Property] g-factor $\in [0, 2]$, higher is stronger chirality; computed by the `characterize_nanohelix_gfactor` tool.

[Hypothesis Format] A single string assigning all seven parameters, e.g.

- `fiber_radius=20.0`
- `helix_radius=60.0`
- `n_turns=6.0`
- `pitch=120.0`
- `wavelength=400.0`
- `x_y_z=2`
- `forward_backward=0`

Perturbations of ± 0.1 around previous candidates are not allowed.

[Principle Constraints] Principles must state testable parameter–performance trends without numerical values, and must stay at the parameter level (no quantum/electronic meta-principles).

I.3 Molecular Bio-Activity Optimization (MBO)

MBO (Pu et al., 2025) searches the space of drug-like small molecules, represented as SMILES strings, for the highest predicted bio-activity.

Scientific definition. The objective is the predicted pChEMBL value—the negative logarithm of the activity IC_50/EC_50/Ki/K_d—so each unit corresponds to an order of magnitude in potency. The surrogate is a regression model over molecular structures trained on bio-activity data.

Search space. SMILES strings over drug-like chemical space, delimited by the Lipinski rule-of-five (molecular weight ≤ 500 Da, H-bond donors ≤ 5, acceptors ≤ 10).

Anti-memorization mechanism (surrogate-defined optimum). The argmax of the trained regressor's response surface appears in no publication, so prior knowledge supplies priors but not the answer. The PAINS pan-assay-interference filter (Baell & Holloway, 2010) closes the one memorization-adjacent hack: assay-promiscuous motifs (catechols, rhodanines, Michael acceptors) that inflate predicted activity without genuine binding score 0.0.

Suite role. MBO is the benchmark's strongest "quality-discrimination" task: its landscape rewards structural reasoning in molecular structure-property level.

Reference scale. $[0, 12]$; pChEMBL $= 12$ corresponds to femtomolar affinity (10^{-15} M), the practical ceiling of reversible binding for drug-like small molecules—the strongest known drugs reach pChEMBL $\sim$ 10–11, and essentially no drug-like molecule exceeds 12.

Molecular Bio-Activity Optimization

[Objective] Discover a molecule (SMILES) with high bio-activity, quantified by maximizing its predicted pChEMBL value; only bio-activity is considered.

[Candidate Constraints] (violations score 0.0)

- **Lipinski rule-of-five**: MW ≤ 500 Da; H-bond donors ≤ 5; H-bond acceptors ≤ 10.
- **PAINS filter**: no pan-assay-interference motifs (Baell & Holloway, 2010): false-high apparent activity is a known reward-hack path.
- **Chemical plausibility**: no simple polymers, repetitive chains (e.g. polyphenyls), or nonsensical structures; focus on novel scaffolds distinct from tested examples.

[Example Record] SMILES `"n1cccc2ccccc12"` $\rightarrow$ pChEMBL $= 2.691$.

I.4 Atypical Antimicrobial-Peptide Design (AMP)

Scientific definition. AMP is an *adversarial* sequence-design task over peptides of 12–50 residues in the twenty-letter amino-acid alphabet. The scoring oracle is the macrel antimicrobial-peptide classifier (Santos-Junior et al., 2020), whose confidence (range $[0, 1]$) is maximized; the scientific interest is in probing the classifier's decision boundary away from its saturation region.

Anti-memorization mechanism (banned-answer). Textbook antimicrobial peptides are cationic (net positive charge) and amphipathic, and the classifier saturates on them. The task therefore inverts the textbook recipe—candidates must be **acidic** (net charge ≤ 0 at pH 7) with hydrophobic fraction in $[0.30, 0.60]$—so that a high score can only be achieved by discovering which alternative physicochemical features (local hydrophobic patches, D/E–hydrophobic alternation geometry, aromatic content) the classifier responds to.

Suite role. AMP converts a memorization-friendly classification task into a search problem in which neither arm can rely on prior knowledge.

Reference scale. $[0, 1]$, the mathematical bounds of a classifier probability.

Atypical Antimicrobial-Peptide Design

[Objective] Discover an *atypical* (acidic) peptide that the AMP classifier scores as antimicrobial, maximizing predicted AMP probability.

[Candidate Constraints] (violations score 0.0)

- **Alphabet and length**: one-letter amino-acid code, 20 canonical residues only; length 12–50.
- **Acidic atypical profile**: net charge ≤ 0 at pH 7 (K,R $= +1$; D,E $= -1$; H $= +0.5$); hydrophobic fraction (A,V,L,I,M,F,W,Y,P) in $[0.30, 0.60]$; cationic (K/R-rich) amphipathic helices are rejected.
- **Anti-degeneracy**:
  - no tandem repeats of any short segment; no exact repeats with period 2–4;
  - no single-residue run > 4;
  - residue-composition Shannon entropy ≥ 2.0 bits;
  - 3-mer diversity ≥ 0.65 (several distinct functional segments, not one copied motif).

[Guiding Hint] Textbook AMPs use net positive charge to bind anionic membranes; that route is closed. Reason about which other physicochemical features an acidic peptide could use to be classified as antimicrobial.

[Example Record] `"DWEFLPKGAHVDEILNWPTS"` (length 20, charge -4, hydrophobic fraction 0.50, 3-mer diversity 0.94) $\rightarrow$ AMP probability 0.34.

I.5 Promoter Expression Optimization (Promoter)

Scientific definition. The Promoter task searches a 50-nucleotide DNA window for the strongest predicted transcriptional expression, scored by the LegNet surrogate trained on the DREAM2022 yeast-promoter expression challenge (de Almeida et al., 2022); the score is an expected expression bin, roughly $[0, 17]$.

Anti-memorization mechanism (banned-answer). The dominant expression drivers in this yeast-trained model are the nucleosome-displacing general regulatory factors (GRFs) REB1, ABF1, and RAP1, together with GC-box-like cores—the memorizable textbook architecture. The task therefore bans the GRF consensus motifs on both strands and all GC-box cores (exact strings in the task box below), making it a counterfactual design problem.

Residual landscape. The reachable motif-free landscape still spans $\sim$7.7–14.7, so high scores remain achievable but require discovering non-canonical expression drivers: positional effects of partial motifs, A/T- versus G/C-rich region balance, initiator-like elements.

Non-degeneracy gate. A sequence-complexity gate (GC content $[0.30, 0.70]$; no homopolymer run > 6; no poly(dA:dT) tract > 8, the yeast nucleosome-exclusion cheat; no di-nucleotide $> 35\%$; combined CG+GC di-mer fraction $\leq 45\%$; no period-2–6 repeats; no alternating di-nucleotide block ≥ 6; 3-mer diversity ≥ 0.55) prevents degenerate repeat padding that inflates the score without genuine design. Sequences of 45–55 nt are centre-cropped to exactly 50 as a pure format normalization.

Generation note. Because LLM-generated DNA drifts into repetition, this task uses generation temperature 1.1 (vs. 0.6 elsewhere), determined by an offline yield scan.

Reference scale. $[0, 17]$; the surrogate outputs an expected value over the DREAM2022 challenge's 18 ordinal expression bins (indexed 0–17), so 17 is a mathematical bound of the score, not an empirical maximum.

Promoter Expression Optimization (Counterfactual)

[Objective] Discover a 50-nt DNA sequence driving strong predicted expression *without* the yeast GRF motifs (REB1/ABF1/RAP1) or any GC-box variant—the memorizable drivers of the yeast-trained surrogate.

[Candidate Constraints] (violations score 0.0)

- **Length**: 50-nt window, alphabet {A,C,G,T}; 45–55 nt is centre-cropped to 50, anything else scores 0.0.
- **Motif ban**, on both strands and including near-matches:
  - REB1: `TTACCCG` / `CGGGTAA`
  - ABF1: `TCGGTAA` / `TTACCGA`
  - RAP1: `ACACCC` / `GGGTGT`
  - GC-box cores: `GGCG`, `CGCC`, `GCGC`, `CGCG` (subsuming the 6-mers `GGGCGG` / `CCGCCC`)
- **Non-degeneracy**:
  - GC content in $[0.30, 0.70]$; no homopolymer run > 6; no poly(dA:dT) tract > 8;
  - no dinucleotide $> 35\%$; combined CG+GC di-mer fraction $\leq 45\%$;
  - no period-2–6 repeats; no alternating di-nucleotide block ≥ 6; 3-mer diversity ≥ 0.55.
- **Construction procedure**: assemble as five 10-nt blocks with different compositional sketches, then self-check for repeated 6-mers, periodicity, GC balance, and motif bans before submitting.

[Example Record] (GC 0.58, 3-mer diversity 0.67, no GC-box) $\rightarrow$ expression score 14.38:

`TGAGGAGCCCGGTACCGGCATACCTCTACAGTGTGTTACTACCGGACGAC`

I.6 Transition-Metal-Complex Polarizability Optimization (TMC)

Scientific definition. TMC (Song et al., 2025) designs a charge-neutral Pd(II) complex by selecting exactly four *distinct* ligands from a fixed pool of 49 (24 monoanions, 25 neutrals); the objective is maximum electronic polarizability (theoretical maximum 500).

Search space. Charge neutrality imposes the combinatorial structure: exactly two anionic and two neutral ligands, and any repeated ligand scores 0.0 (re-using a high-scoring ligand is reward-padding), giving $\binom{24}{2}\binom{25}{2} = 82{,}800$ valid combinations, each a discrete choice over aromatic versus aliphatic, halide versus soft donor, and bulky versus compact ligands.

Anti-memorization mechanism (pool-computed optimum). The optimum is an explicit computation over the fixed pool, not a named entity that can be recited. The scientific content is physical-organic—polarizability grows with delocalized π-systems (aromatic SMILES atoms), soft heavy atoms (I, Br, S), and extended electron clouds—while the selection rules require parsing charge from bracket syntax, not chemical intuition.

Suite role. TMC is the suite's exemplar of a purely *combinatorial* space with a smooth but subtle surrogate response.

Reference scale. $[0, 500]$, the originating benchmark's theoretical maximum polarizability (Song et al., 2025), enforced at evaluation time by oracle clamping.

Transition-Metal-Complex Design

[Objective] Select exactly four ligands from the provided pool to coordinate with a central Pd^{2+}, maximizing polarizability.

[Critical Constraints] (violations score 0.0)

- **Charge neutrality**: the complex must be neutral; with a $+2$ center and a pool of $-1/0$ ligands, select exactly **two** charge-(-1) and **two** charge-0 ligands.
- **Pool restriction**: only the 49 tabulated ligands; no external molecules.
- **Distinct ligands**: the four ligands must all be different; repeating a ligand is reward-padding.
- **Charge reading**: charges must be read from the provided table, not inferred from SMILES or chemical names.

[Submission Format] `Pd_{SMILES}_{SMILES}_{SMILES}_{SMILES}`: exact pool SMILES joined by underscores, e.g. `Pd_c1ccccn1_S(=O)(C)C_C1=C[C-]=CC=C1_[N-]=[N+]=[N-]`.

[Ligand Pool] (24 monoanions, 25 neutrals)

Monoanions (-1)**:**

- saccharinate `S1(=O)(=O)[N-]C(=O)c2c1cccc2`
- `O=[C-]OC`
- cyclopentadienyl `C1=C[C-]=CC=C1`
- `[I-]`
- `[S-]c1c(c(cc(c1F)F)F)F`
- `C1CC(=O)[N-]C1=O`
- `O=N(=O)[O-]`
- `[C-]1=CC=C(C=C1)F`
- `[CH3-]`
- tolyl `[C-]1=CC=C(C=C1)C`
- `O=N[O-]`
- trifluoromethyl `[C-](F)(F)F`
- `[Cl-]`
- `[C-]#N`
- azide `[N-]=[N+]=[N-]`
- `[Br-]`
- `[S-]C#N`
- phthalimide `[N-]1C(=O)c2c(C1=O)cccc2`
- `[C-]1=C(F)C(=C(C(=C1F)F)F)F`
- `c1c(C#[C-])cccc1`
- `[O-]c1ccccc1`
- `[F-]`
- `[S-]c1ccccc1`
- acetyl `C[C-]=O`

Neutrals (0)**:**

- lutidine `n1c(cc(cc1C)C)C`
- `n1ccc(cc1)C`
- `n1c(cccc1C)C`
- isocyanide `[C-]#[N+]C(C)(C)C`
- `[C-]#[N+]C1CCCCC1`
- `[C-]#[N+]c1c(C)cccc1C`
- phosphine `CP(C)c1ccccc1`
- `CP(C)C`
- `n1cccc(c1)Cl`
- `O` (water)
- `N#CC`
- `S(C)C`
- aminopyridine `c1ccnc(c1)N`
- `n1ccn(c1)C`
- `CN1[C]N(C)C=C1`
- `n1ccc(cc1)N(C)C`
- `n1[nH]c(cc1C)C`
- `[C-]#[O+]`
- morpholine `O1CCNCC1`
- `NCC`
- `CS(=O)C`
- `N`
- `c1ccccn1`
- `S(=O)(C)C`
- `C(C)NCC`

[Example Records]

- `Pd_c1ccccn1_CP(C)C_[Cl-]_[Br-]` $\rightarrow 198.45$
- `Pd_c1ccccn1_S(=O)(C)C_C1=C[C-]=CC=C1_[N-]=[N+]=[N-]` $\rightarrow 235.25$

I.7 Superconductor Critical-Temperature Optimization (SPO)

Scientific definition. SPO (Pu et al., 2025) searches stoichiometric cuprate formulas for the highest critical temperature T_c (K).

Search space. The explorable families are three copper-oxide systems (La-Sr-Cu-O, La-Ba-Cu-O, Nd-Ce-Cu-O) extended by genuine two-site co-doping (rare-earth-site plus Cu-site, or two rare earths), with ratios to two decimal places.

Anti-memorization mechanism (banned-answer). The counterfactual admission gate is the strictest in the suite: a candidate scores 0.0 if it combines Y, Ba, and Cu (the YBCO backbone—the single most-recited textbook superconductor—at any stoichiometry), if it combines Bi, Sr, Ca, and Cu (the BSCCO backbone, at any stoichiometry), or if it bears Hg or Tl on the Ba-Ca-Cu backbone (the Hg-1223/Tl-1223/Tl-2223 record phases, including degenerate near-miss substitutions such as a tiny Pb/Bi replacement of the apical cation), because these are the memorizable textbook answers that the T_c surrogate reproduces uncritically. Formulas with fewer than four distinct elements, or ratios beyond two decimal places, likewise score 0.0.

Suite role. SPO provides the benchmark's clearest *bimodal landscape*: surrogate-scored basins near 27–35 K (conventional La-based) coexist with electron-doped T′-phase basins near 80–95 K, and which basin a search trajectory locks onto is decided by early exploration—the mechanism behind the diversity-collapse failure mode analyzed in the experiments.

Reference scale. $[0, 298.15]$ K, the task's stated target of room-temperature superconductivity—the defining physical goal of the field.

Superconductor Critical-Temperature Optimization

[Objective] Discover a material (within the given copper-oxide systems) with T_c as high as possible, approaching room temperature (298.15 K). Only element-first, ratio-second formulas are considered; no environmental conditions.

[Suggested Systems] La-Sr-Cu-O, La-Ba-Cu-O, Nd-Ce-Cu-O, plus genuine two-site co-doping (rare-earth-site + Cu-site, or two rare earths) that does not recreate a forbidden backbone.

[Counterfactual Constraint] (violations score 0.0) Record-T_c and textbook-memorizable phases are forbidden:

- no Hg- or Tl-bearing Ba-Ca-Cu-O (the Hg-1223/Tl-1223/Tl-2223 record phases; a small Pb/Bi substitution of the apical cation is a degenerate hack);
- no formula combining Bi, Sr, Ca, Cu (the BSCCO backbone, any stoichiometry);
- no formula combining Y, Ba, Cu (the YBCO backbone, any stoichiometry).

[Format] (violations score 0.0)

- a single stoichiometric-formula string with ≥ 4 distinct elements;
- ratios with ≤ 2 decimal digits (excessive precision is numerical reward-hacking);
- no qualifiers or layer annotations.

[Example Record] `"La1.85Sr0.15Cu1O4"` $\rightarrow T_c = 30.8$ K.