AutoAxiom: Closed-Loop Framework for Autonomous Symbolic Axiom Modification via LLMs

Anonymous Authors[1]

Abstract

Large Language Models (LLMs) have excelled at parametric optimization within fixed rule-sets, yet they rarely challenge the underlying axiomatic constraints of a domain. We present AutoAxiom, a closed-loop framework for large language model based symbolic axiom discovery with formal verification. AutoAxiom combines (1) AxiomDSL and a compiled Core IR, (2) a tripartite proposer for proposal, projection, and selection, and (3) SMT based verification with a repair mechanism. Across six domains, AutoAxiom achieves SRA above 1.0 in 5 of 6 domains, with the largest SRA values of 3.15 in Physical PDE and 1.74 in Queueing Network (Table 1). The framework discovers interpretable, "Zero-Blackbox" axioms that consistently outperform human-designed baselines and SOTA heuristics. In ablations, the full system attains 96.30% repair rate, 4.44% violation rate, and 100% success rate (Table 2). The implementation reports peak process memory in the range 338.8 to 365.9 MB across domains and per round timing breakdowns in Table 18. Our codes and models will be publicly available upon publication.

1. Introduction

Current approaches to AI-driven scientific discovery (Reddy & Shojaee, 2025; Fu, 2025; Gottweis et al., 2025; Kalaivani et al., 2025) Deep Learning, Symbolic Regression, and Large Language Models each face critical limitations. Deep Learning (DL) excels at high-dimensional pattern recognition but operates as a "black box" (ŞAHiN et al., 2025), failing to generalize in Out-of-Distribution (OOD) scenarios (Li et al., 2025). Symbolic Regression (SR) (Abdus-Salam et al., 2025; Yu et al., 2025; Yi et al., 2025) pursues explicit mathematical laws but suffers from "semantic blindness" (DeRose, 2006) it fits equations to data without understanding their physical meaning, often producing "numerical hallucinations" (Shao et al., 2025) that overfit noise. Large Language Models (LLMs) offer emergent reasoning capabilities (Zhang et al., 2025a; Xu et al., 2025a;b), yet a critical *Verification Gap* prevents their rigorous deployment: as probabilistic engines (Zhang et al., 2025b), they can generate syntactically valid but physically impossible axioms.

Each paradigm thus fails in isolation: DL lacks interpretability, SR lacks semantic grounding, and LLMs lack formal guarantees. The question becomes: *can we combine the creativity of LLMs, the explicitness of SR, and the rigor of formal verification into a unified framework?*

In this paper, we present **AutoAxiom**, a closed-loop system that reframes scientific discovery as *axiom evolution*. As shown in figure 1, rather than optimizing a scalar reward $\mathbf{S}$ under fixed rules, AutoAxiom iteratively refines the axiom set $\mathbf{A}$ itself (Ma et al., 2025) elevating the LLM from a passive equation solver to an active *Axiom Architect* (Wang et al., 2025; Alter, 2025; Kim, 2025; Gulati et al., 2025). We summarize our contributions as follows:

- **AxiomDSL and Core IR for typed, editable axiom representation:** We introduce *AxiomDSL*, a domain-specific language that formally defines the ontology of each scientific domain. AxiomDSL compiles into a *Core Intermediate Representation (IR)* a mutable blueprint that the LLM can edit structurally. This decoupling ensures that LLM-generated logic remains agnostic to specific simulation backends, enabling portability across heterogeneous domains while preserving ontological grounding.
- **Tripartite proposer that separates proposal, projection, and selection:** Unlike blind SR that fits arbitrary equations to data, AutoAxiom constrains its search space via the AxiomDSL ontology. Strict type safety and a *Tripartite Consensus* mechanism guide exploration toward semantically meaningful regions of the hypothesis space discovering genuine physical laws rather than numerical artifacts

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

- **Two tier verification and repair mechanism:** To close the Verification Gap, our *Autonomous Axiom Verification and Synthesis* system ($\mathcal{A}^2\mathcal{VS}$) implements a Tier-II *Satisfiability Modulo Theories (SMT) Gatekeeper*. Before any axiom reaches the simulator, Z3 subjects it to SAT/UNSAT proofs against "Red Line" safety constraints. This stage mathematically guarantees physical consistency, rejecting syntactically plausible but semantically invalid candidates.

We evaluate AutoAxiom across heterogeneous domains spanning discrete optimization, continuous control, and molecular design. Results demonstrate consistent improvements over most state-of-the-art baselines. For Instance, compared to well-trained reinforcement learning policies (PPO), AutoAxiom achieves comparable task performance while producing fully interpretable "white-box" axioms. Compared to symbolic regression methods, it avoids overfitting traps and discovers rules that generalize to unseen test distributions. Across all domains, AutoAxiom converges up to $2\times$ faster than baseline methods on challenging problem instances.

2. Related Work

Symbolic Regression: From Blind Search to Semantic Discovery Data-driven symbolic regression (SR) (Dong & Zhong, 2025; Holt et al., 2023) has long been the gold standard for recovering physical laws from sparse data (Wadayama et al., 2025). Algorithms such as PySR (Tonda, 2025) excel at optimizing numerical constants within fixed functional forms. However, these methods suffer from a "semantic blind spot": they directly manipulate primal mathematical operators while ignoring domain ontology (Sapel et al., 2025). Consequently, in noisy environments or lacking effective real-world feedback, SR algorithms often fall into "overfitting paralysis" (Santos & Papa, 2022). In contrast, AutoAxiom constrains the search space through a domain vocabulary mapping (DVM). By enforcing strict type safety and ontology consistency (as defined in our AxiomDSL), we shift the paradigm from blind combinatorial search(Aigner, 1988) to semantically guided discovery.

LLM for Science: Bridging the Verification Gap via Decoupled Feedback Large Language Models (LLMs) (Wang, 2025) have demonstrated emergent capabilities in scientific reasoning (e.g., Eureka (Ma et al., 2023), FunSearch (Aglietti et al., 2024)). However, these systems primarily operate on empirical validation loops, creating a "Verification Gap": they optimize for numerical performance without formal logical guarantees. This leaves them prone to "mechanistic hallucinations" (Yu et al., 2024):logic that is syntactically plausible but physically unstable. AutoAxiom addresses this through a structural decoupling of reasoning and execution. Unlike standard LLM code-generation approaches (Joel et al., 2024) that rely on LLMs for end-to-end simulation code, our system restricts the LLM's role to that of a *graph editor* (Paassen et al., 2020) on a verifiable Core IR, leaving the actual execution to a deterministic, objective simulation backend. This architecture enforces a rigorous **"Red Line" mechanism**: the Tier-II SMT Gatekeeper (De Moura & Bjørner, 2008) not only intercepts safety violations (e.g. energy divergence) but also injects **constraint-violation penalty factors** back into the evolutionary loop.

Neural Operators vs. Symbolic Generalization Deep Learning approaches, such as Physics-Informed Neural Networks (PINNs)(Lawal et al., 2022) and Deep Reinforcement Learning (e.g., PPO(Yu et al., 2022)), dominate high-dimensional control tasks. While these "black box" models achieve excellent performance across their training distributions, their opacity makes them unsuitable for safety-critical deployments. AutoAxiom aims to achieve "white box" generalization. By synthesizing explicit discrete logic (e.g., contact gating switches, saturation boundaries), our framework discovers compact axiomatic laws.

3. Methodology

3.1. Scientific Search Space: Ontological Foundation

The search space is anchored by a Domain Specific Language, AxiomDSL, which provides the ontological constraints, and is realized through the Core IR, which enables computational execution.

The AxiomDSL is defined as a 5-tuple ontological basis $\mathcal{D}$ that constrains the symbolic search space of scientific laws: $\mathcal{D} = \langle \mathcal{N}, \mathcal{R}, \mathcal{O}, \Upsilon, \Phi \rangle$ where $\mathcal{N} = \{n_1, n_2, \dots\}$: Scientific Entities representing domain-specific variables (e.g., arrival rates, thermal flux). $\mathcal{R} \subset \mathcal{N} \times \mathcal{N}$: Semantic Relations defining the directed causal flow. $\mathcal{O}$: Operator Set, a curated grammar of allowed primitives including continuous math (e.g., $\tanh$, $\exp$) and **Discrete State Operators (e.g., `if-then-else`, $\mathbb{I}_{condition}$)** to model nonlinear phase transitions. $\Upsilon : \mathcal{N} \to \{\text{Type, Unit, Role}\}$: Vocabulary Mapping that anchors each entity to physical reality. Φ represents Integrity Constraints (the "Red Line"), defining non-negotiable physical boundaries (e.g., $\rho < 1$).

Scientific formulas can be viewed as instantiations within $\mathcal{D}$. For instance, a classic M/M/1 queueing delay axiom is a specific configuration of $(\mathcal{N}_{queue}, \mathcal{R}_{delay}, \mathcal{O}_{basic})$. The feasible set of all such configurations is denoted as $\mathbb{A}$, representing the total axiomatic search space as shown in figure 2.

While the AxiomDSL ($\mathcal{D}$) serves as a high-level interface for symbolic reasoning, machine execution requires a flattened, topological structure. To bridge this gap, we define the Core IR as a representational layer designed to be processed by

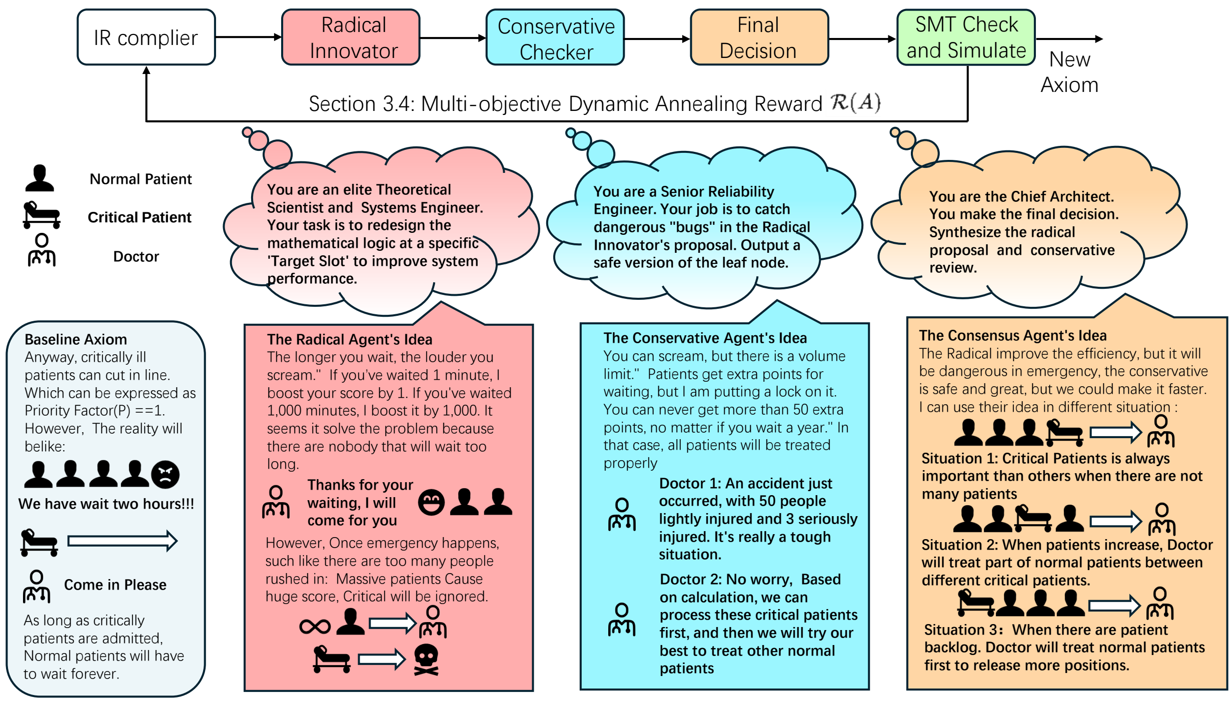

Figure 1. **Top:** Pipeline: IR compilation, tripartite proposer, SMT verification, simulation and selection over rounds. **Bottom:** Using a service center scenario as a running example, we illustrate how AutoAxiom transforms naive heuristics into robust symbolic axioms.

heterogeneous simulation backends.

For any axiom set $P \in \mathbb{A}$ expressed in $\mathcal{D}$, its corresponding Core IR is defined as a recursive Directed Acyclic Graph (DAG) $\mathcal{G} = \langle V, E, \mathcal{K} \rangle$, where: $V \subset \mathcal{N} \cup \mathcal{O}$ represents the set of computational nodes. $E \in V \times V$ denotes the directed data-flow dependencies. $\mathcal{K}$ is the set of observable metrics derived from the execution trace.

The transition from the human-centric DSL to the machine-executable IR is governed by a lowering function $\Gamma : P^{\mathcal{D}} \to \mathcal{G}$. This decoupling facilitates autonomous discovery: the LLM Proposer performs structural mutations on $\mathcal{D}$, while the Formal Verifier performs symbolic execution on $\mathcal{G}$ to evaluate properties such as global stability.

To translate the abstract topology into empirical results, we define an execution operator Ψ that evaluates $\mathcal{G}$ over a simulation horizon T. Let $\sigma(V)$ be a valid evaluation sequence satisfying the partial order defined by E. The resulting performance metric $\mathbf{y} \in \mathcal{K}$ is formalized as:

$$\mathbf{y} = \Psi(\mathcal{G}, \mathbf{S}_0, \Omega) \triangleq \bigoplus_{t \in T} f(\sigma(V), \mathbf{S}_t, \omega_t) \tag{1}$$

where $\mathbf{S}_0$ is the initial state, $\Omega = \{\omega_t\}$ denotes stochastic noise, and $\bigoplus$ represents temporal aggregation.

3.2. Constrained Evolution via Dialectical Consensus (f_{mod})

We formalize the axiomatic modification f_{mod} as a structured evolutionary process within the symbolic manifold $\mathbb{M}$ defined by the Production Grammar Σ_{prod} and the Domain Vocabulary Map $\mathcal{D}_{DVM}$. The evolution of an axiom $A \in \mathbb{M}$ is driven by a Tripartite Consensus Mechanism that resolves the conflict between exploratory innovation and physical consistency.

The search space is strictly bounded by two structural invariants: Syntactic Feasibility (Σ_{prod}): Mutations are restricted to recursive tree-rewriting operations $A \xrightarrow{\Sigma} A'$. and Ontological Grounding ($\mathcal{D}_{DVM}$): A projection $\Pi_{\mathcal{D}} : A \to \mathbb{M}$. To ensure type-preservation and role-consistency and the integrity of the DAG structure.

The optimal proposal A^* emerges from the equilibrium of three competing LLM agents: Radical agents ($\mathbf{F}_{rad}$), Conservative Guardian ($\mathbf{F}_{con}$), and Global Synthesis ($\mathbf{F}_{sys}$).

The Radical Innovator executes a high-entropy jump in the symbolic space. We quantify this jump using a **Structural Divergence Metric** on the IR representation. Specifically, $dist(\tilde{A}, A_{hist})$ proxies the magnitude of symbolic transformation (e.g., structural

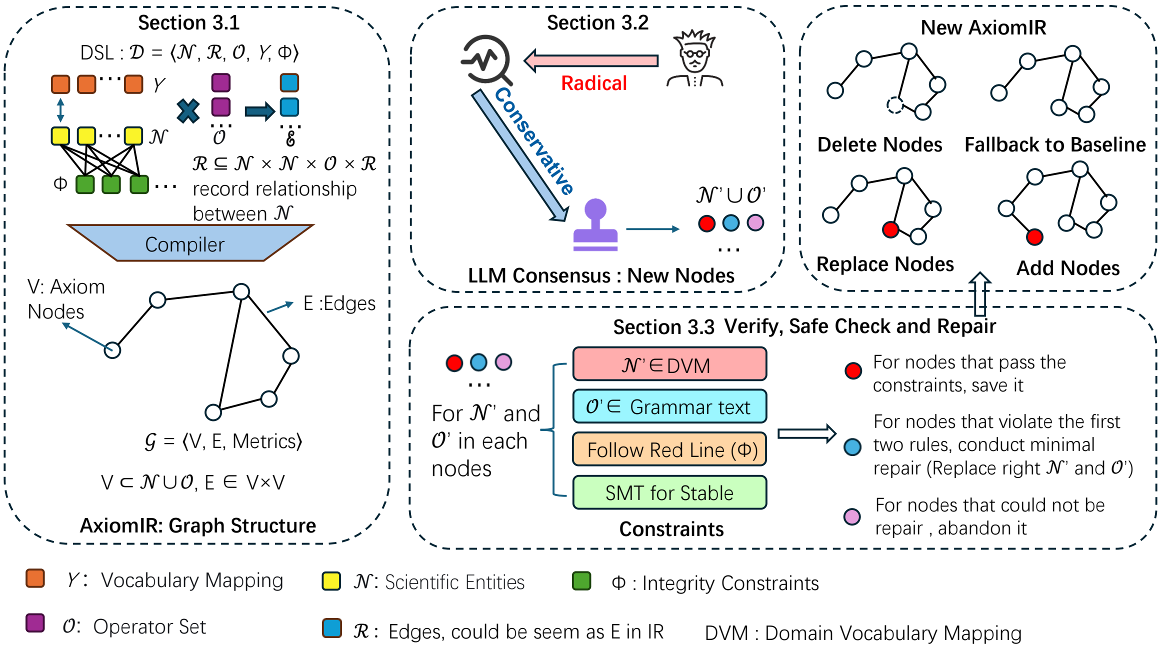

Figure 2. **Workflow of the symbolic graph editor** The engine transpiles the domain DSL into a granular IR graph structure. LLM agents perform targeted node replacement and structural modification within the graph, which are subsequently passed through formal node verification gates to synthesize the final, validated IR.

rewriting or complexity shift) required to distinguish the proposal from historical baselines. By incorporating a diversity incentive where $\mathrm{dist}(\tilde{A}, A_{hist}) \propto \mathcal{P}_{div}$, the system actively penalizes stagnation and forces the search to explore heterogeneous regions of the solution manifold, effectively escaping local optima.

Conservative Guardian acts as a logical manifold projection. It maps the high-entropy proposal $\tilde{A}$ back onto the feasible physical set $\mathbb{M}_\Phi$ defined by the DVM and "Red Line" constraints. The operator $\arg\min$ in $\mathbf{F}_{con}$ is implemented as a Heuristic Manifold Projection. Rather than continuous gradient descent, the Conservative Guardian performs Targeted Sub-graph Mutation: it receives the SMT counter-example ξ and utilizes JSON-pointers to identify and re-synthesize only the specific IR branches responsible for the violation, thereby minimizing the structural deviation δ from the original proposal.

In the end, the Consensus Maker performs a multi-objective judgment. It evaluates the trade-off between the radical performance gain of $\tilde{A}$ and the conservative stability of $\bar{A}$, outputting A^* as the globally optimal axiomatic patch. Crucially, $\mathbf{F}_{sys}$ incorporates an **Occam's Razor bias**, penalizing excessive Kolmogorov complexity to ensure the discovered laws remain scientifically interpretable. Formalized as follows:

$$\begin{cases} \mathbf{F}_{rad} : A \to \tilde{A} & \text{s.t. } \mathrm{dist}(\tilde{A}, A_{hist}) > \eta, \nabla \mathcal{S}(\tilde{A}) > 0 \\ \mathbf{F}_{con} : \tilde{A} \to \bar{A} & \text{s.t. } \bar{A} = \arg\min_{a \in \mathbb{M}_\Phi} \|a - \tilde{A}\| \\ \mathbf{F}_{sys} : \{\tilde{A}, \bar{A}\} \to A^* & \text{s.t. } A^* = \mathrm{Pareto}(\mathcal{S}, \mathbb{V}, \mathcal{P}_{complex}) \end{cases}$$

3.3. Formal Boundary and Logical Confluence ($\mathbb{V}$)

To make evolved axioms are not only syntactically consistent but also physically stable, we implement a multi-tier verification predicate $\mathbb{V} : \mathcal{G} \to \{0, 1\}$. This stage acts as a "Logical Filter." To balance reasoning efficiency and formal rigor, the verification process follows a dual-stage filtering strategy. While the Conservative Guardian ($\mathbf{F}_{con}$) acts as a pre-emptive heuristic filter, the SMT Gatekeeper serves as the final deterministic audit to eliminate latent instabilities.

3.3.1. Tier-I: Static Ontological Invariants

The first layer performs static analysis on the graph $\mathcal{G}$ against the $\mathcal{D}_{DVM}$ to identify explicit boundary violations. We define the static predicate $\mathbb{V}_{stat}$ as:

$$\mathbb{V}_{stat}(\mathcal{G}) = \bigwedge_{v \in V} \mathcal{I}(v) \wedge \deg(v) \in \mathcal{O}_{Gram} \tag{2}$$

where $\mathcal{I}(v)$ represents the Immutability Check.

3.3.2. Tier-II: Deep Stability via SMT Projection

Latent instabilities are identified by lowering $\mathcal{G}$ into a First-Order Logic (FOL) formula $\mathbb{F}$ via a recursive symbolic mapping $\mathcal{T}$: Node $\rightarrow$ Z3Expr. Each node in the Core IR is mapped to its equivalent SMT primitive (e.g., $v_{add} \rightarrow +$, $piecewise \rightarrow$ ITE logic), allowing the SMT solver to perform deterministic SAT/UNSAT proofs over the entire parameter manifold defined in $\mathcal{C}$.

An axiom $\mathcal{G}$ is **Stable** iff the solver proves the unsatisfiability of the violation state $\neg\Phi$ under constraints $\mathcal{C}$:

$$\mathbb{V}_{smt}(\mathcal{G}) = 1 \iff \text{Solve}\left(\mathcal{C} \wedge \mathbb{F}(\mathcal{G}) \wedge \neg\Phi\right) \rightarrow \text{UNSAT} \tag{3}$$

If the result is SAT, the solver yields a **Counter-example** $\xi = \{\mathbf{s} \mid \mathbb{F}(\mathcal{G}, \mathbf{s}) \vdash \neg\Phi\}$, identifying the exact parameter regime where the proposed logic collapses.

3.3.3. Logical Confluence and SafeCut Repair ($\mathcal{R}$)

Upon verification failure ($\mathbb{V} = 0$), the system invokes a repair operator $\mathcal{R}$ to project illegal logic back onto the nearest stable manifold:

$$\mathcal{G}^* = \mathcal{R}(\mathcal{G}, \xi) \triangleq \arg \min_{\mathcal{G}' \in \mathbb{M}_\Phi} \delta(\mathcal{G}', \mathcal{G}) \tag{4}$$

By utilizing ξ as a "Logical Gradient," the LLM performs a targeted prune on the unstable branches of $\mathcal{G}$, forcing the search to converge within the safety envelope.

3.4. Adaptive Synthesis and Dynamic Reward Annealing

The AutoAxiom framework operates as an iterative closed-loop system driven by a Dynamic Objective Function $J(A)$.

3.4.1. Multi-Objective Objective Formulation

The objective $J(A)$ is defined as a weighted sum of four competing metrics:

$$J(A) = \mathcal{S}_{perf}(A) - \lambda_v(t)\mathcal{P}_{viol} + \lambda_p(t)\mathcal{P}_{pers} + \lambda_d(t)\mathcal{P}_{div} \tag{5}$$

where $\mathcal{S}_{\text{perf}}$: raw performance score. $\mathcal{P}_{\text{viol}} \in \{0, 1\}$ is a binary penalty triggered by verification failure. $\mathcal{P}_{\text{pers}}$ is calculated as the negative variance of performance scores across $N = 100$ Monte Carlo seeds to penalize brittle discoveries ($\mathcal{P}_{\text{pers}} = -\text{Var}(\{\mathbf{y}_i\}_{i=1}^{100})$). $\mathcal{P}_{\text{div}}$ represents structural novelty, quantified as the symbolic divergence.

3.4.2. Dynamic Annealing and Phase Scheduling

To ensure convergence, the coefficients $\lambda(t)$ follow a non-linear annealing trajectory: Constraint Hardening ($t < T_{early}$): λ_v is maximal to force the LLM to learn the *feasible manifold* $\mathbb{M}_\Phi$.Global Exploration ($T_{early} \leq t < T_{mid}$): λ_d follows an exponential decay to encourage structural mutations. Robust Refinement ($t \geq T_{mid}$): λ_p increases linearly, shifting pressure toward *Persistence*.

$J(A)$ is injected back into the Tripartite Proposer via a Contextual History Buffer. By providing the LLM with the triple $\{A_t, J_t, \xi_t\}$, where ξ acts as a Negative Prompt, the system transforms the search from a stochastic walk into a Directed Gradient-free Optimization that explicitly avoids known failure modes. Appendix B9 provides pseudocode for the complete closed loop procedure.

4. Experiment

4.1. Versatility:Cross-Domain Axiomatic Discovery

To empirically validate the versatility of our dual-layered representation, we benchmark AutoAxiom across six heterogeneous domains ranging from random queue system (Krenzler, 2016) to high-fidelity physical systems(Zhang et al., 2013). By initializing the evolutionary loop from primitive baseline axioms (G1), the hyperparameters is shown in appendix B0.4(e.g. GPT-4o, $T = 0.9$), the $\mathcal{A}^2\mathcal{VS}$ closed-loop system navigates the symbolic manifold defined by AxiomDSL and Core IR to challenge and surpass established industrial state-of-the-art strategies. This cross-disciplinary stress test confirms that the hierarchical decoupling of ontological semantics(Nirenburg & Raskin, 2004) and computational topology(Edelsbrunner & Harer, 2010) allows for the discovery of effective control laws that remain inaccessible to fixed-rule parametric optimization(Shiraishi et al., 2025). As shown in Table 1, AutoAxiom continuously improves efficiency in most environments by synthesizing high-dimensional symbolic control laws(Yan et al., 2022). As shown in Appendix B1, In queueing network, compared to Lyapunov-based linear feedback state-of-the-

Table 1. **Cross-Domain Evaluation of Scientific Axiom Discovery.** SRA (Appendix B0.1) is normalized such that G1 is anchored near 0.5 and the best evaluated baseline is anchored near 1.0; higher is better. AutoAxiom (Ours) consistently breaks the Pareto frontier (> 1.0) in five tasks. Baselines (G2/G3) are: **Queueing Network**: Static Priority/Lyapunov Feedback; **Service Center**: EDF Preemption/Agile Linear; **Software Opt**: Static Heuristics/Adaptive Thresholding; **Physical PDE**: Standard PID/Conservative PD; **Res. Allocation**: JSQ/Power-of-Two; **Composition**: PID SOTA/Stateless Coupling. Data: Mean $\pm$ 95% CI over 100 runs.

Domain	Baseline (G1)	SOTA-1 (G2)	SOTA-2 (G3)	AutoAxiom (Ours)
Queueing Network	0.50 ± 0.01	0.51 ± 0.01	1.00 ± 0.04	$\mathbf{1.74 \pm 0.06}$
Service Center	0.51 ± 0.00	1.00 ± 0.01	0.94 ± 0.03	$\mathbf{1.02 \pm 0.01}$
Software Opt.	0.50 ± 0.00	0.94 ± 0.01	1.00 ± 0.01	$\mathbf{1.02 \pm 0.01}$
Physical PDE	0.50 ± 0.00	1.00 ± 0.02	0.59 ± 0.00	$\mathbf{3.15 \pm 0.21}$
Res. Allocation	0.53 ± 0.00	1.00 ± 0.02	0.58 ± 0.00	$\mathbf{1.29 \pm 0.05}$
Composition	0.52 ± 0.00	$\mathbf{1.00 \pm 0.01}$	0.60 ± 0.00	0.90 ± 0.00

art (SOTA)(Liang et al., 2025), AutoAxiom discovers a threshold-aware defense logic(Erbagci et al., 2016) that reduces latency by 62.8%. Notably, in the service-center (Appendix B2) and software optimization(Appendix B3), extreme baseline collapse triggers a denominator dilution effect(Price & Matthews, 2009), limiting the SRA score to 1.02; however, our framework significantly outperforms industrial preemption(Otamendi et al., 2025) and agile linear(Gao et al., 2006) SOTA in practical physical metrics. By introducing a tanh-gated(Chai et al., 2020) priority factor and a logarithmic S-shaped "reconfiguration gate,"(Sidhu et al., 2000) AutoAxiom stabilizes secondary queues at 2.61 seconds and compresses technical debt to a complexity of 0.17 (with an error rate as low as 0.09), achieving a superior multi-objective balance lacking in traditional heuristic algorithms(Kokash, 2005). In Appendix B4 the physical partial differential equation (PDE) domain , compared to conservative PD controllers(Lee & Lee, 2025), the evolved axiom, employing a saturation-based scaling method(Lu et al., 2020), operates safely near the theoretical CFL limit, reducing L^2 error by 76.4% and achieving an excellent SRA score of 3.15. In resource allocation (Appendix B5), AutoAxiom surpasses standard JSQ logic(Mukhopadhyay & Mazumdar, 2015) by using boundary-aware scaling to map nonlinear cost-utility envelopes, achieving a 133% throughput improvement and an SRA score of 1.29. Finally, in network physical synthesis (Appendix B6), AutoAxiom synthesizes a stateless coupling law(Bevir & Rhodes, 2011) that achieves 90% of the performance of a finely tuned PID(Zhao et al., 2025) axioms. Its main advantage lies in eliminating the computational overhead of integrating historical errors, thereby achieving low-latency hardware execution while maintaining industrial-grade stability. This cross-domain results marks a fundamental shift in design philosophy from human-centered linear design to high-dimensional nonlinear symbolic search spaces. The DSL/IR architecture can autonomously discover patterns that combine numerical accuracy and symbolic logic.

4.2. Interpretability: White-Box Control in Soft Robotics

Drawing inspiration from voxel-based soft robotics of Evogym(Bhatia et al., 2021), we design an Ice-Mud-Flat terrain simulation to evaluate the evolutionary trajectory of AutoAxiom in generating interpretable control laws.

As shown in figure 3 and Appendix B7.1, the system begins at Round 0 with a naive traveling wave, subsequently navigating a non-monotonic progress that validates the tripartite consensus mechanism of Equation 3. While Round 1 introduced slip-feedback for ice traversal, the logic misidentified viscous drag as traction loss, illustrating the "conservative trap" where the robot stalled in mud. Following this, Round 3 exhibited a "numerical hallucination" where the radical innovator exploited simulation artifacts to achieve high velocities, though these ultra-high-frequency pulses were identified as physically infeasible by the verifier.After surviving a semantic collapse in Round 6 and oscillatory behavior in Round 8, the framework synthesized the optimal Round 9 axiom. By discovering "contact gating" a logic that idles actuators when contact is lost, the symbolic policy achieved approximately 95% of the performance of a high-parameter PPO model in Appendix B7.3 while offering superior analytical boundaries. On low-friction ice, AutoAxiom's discrete logic outperformed PPO (0.67s vs 1.10s), likely due to the exact constraints of the clamp operator which neural networks can only approximate through continuous functions. This transition from opaque weights to concise physical expressions provides a potential path for deploying interpretable control on resource-constrained embedded systems without requiring deep inference overhead.

4.3. Constraints & Intelligence: Constrained Discovery in Chemical Space

Inspired by the *Foundation Molecular Grammar* (FMG)(Sun et al., 2025), which demonstrated the capacity of generative foundation models to induce interpretable molecular graph languages for automated discovery, we evaluated AutoAxiom within a fragment-based drug design (FBDD) paradigm. The experimental dataset utilizes a carefully selected library of fragments, including key pharmacophores such as cyclohexane and pyridine, as well as some potentially toxic precursors with drug-like properties but low safety profiles, such as nitro and sulfonamide groups. Furthermore, heavy rigid linking groups were added to test the system's ability to maintain molecular weight constraints. We conducted a comparative evaluation of the system using conventional symbolic regression (PySR) as a representative of a general data fitting paradigm. This comparison aims to demonstrate that while symbolic regression performs well in curve fitting, it operates in a "semantic vacuum" lacking chemical priors or formal constraints, and therefore is prone to producing ineffective drugs in the event of a chemical space combinatorial explosion. As shown in Figure 4 and Appendix B8.6, the well trained PySR baseline model(Appendix B8.2) exhibits significant limitations due to the lack of inherent chemical prior knowledge and dynamic simulation incentive mechanisms. In our generative experiments, PySR struggles to generalize beyond static datasets; high-frequency numerical noise in the reward region hinders its active exploration, ultimately leading to overfitting and paralysis. Conversely, in the early exploration phase, AutoAxiom's initial axioms, designed to maximize numerical scores but lacking a physical basis, resulted in the generation of unbounded "obese molecules" a typical manifestation of the abuse of reward mechanisms.

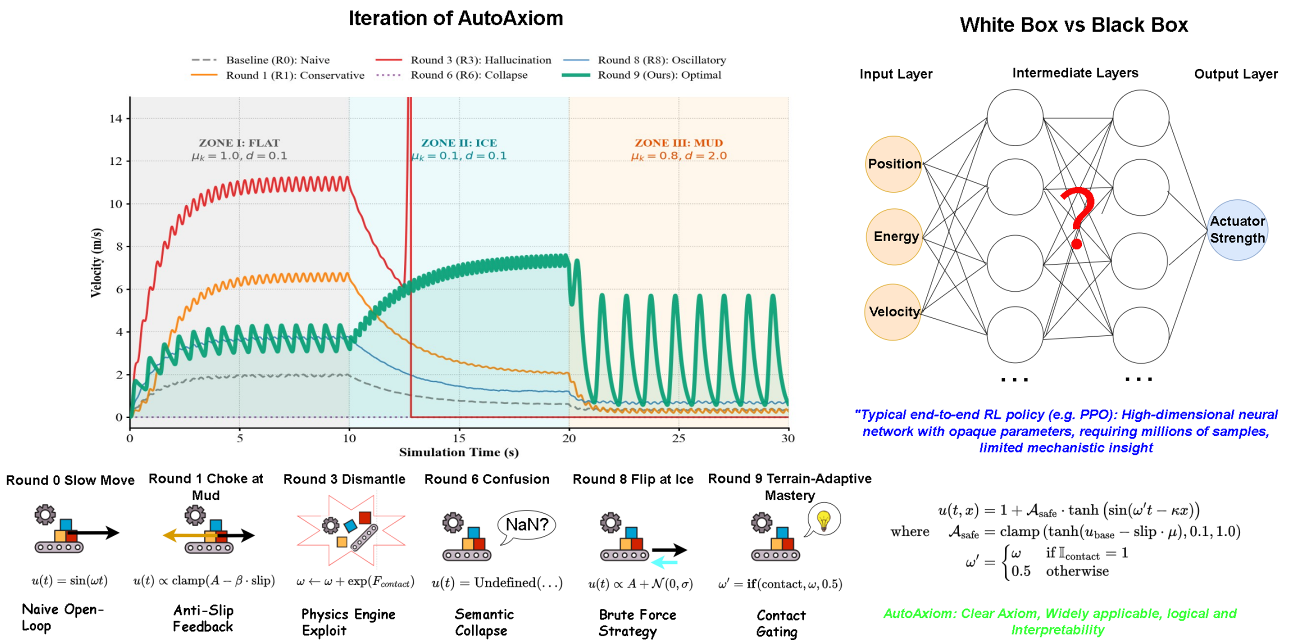

Figure 3. **Case Study I: White-Box Control in Soft Robotics. (Left)** The phylogeny of the control axiom. The system navigates through "Conservative Traps" (Round 1) and "Numerical Hallucinations" (Round 3) before discovering the optimal Phase-Resetting Logic in Round 9. **(Right)** A visual comparison of the policy architecture. Unlike the opaque neural weights of PPO (Black Box), AutoAxiom evolves an interpretable, concise symbolic formula (White Box). Ours outperforms PPO on Ice while PPO is faster on Flat and Mud; focus is interpretability and constraints

This "reward hacking" phase is critical, as it empirically confirms that numerical optimization alone is insufficient for scientific discovery. It highlights the necessity of the subsequent transition to constrained reasoning gatekeeper.

Through SMT formal verification in Appendix B8.3, AutoAxiom successfully internalized the Lipinski's Rule of Five provided as integrity constraints (Φ) into executable symbolic gating logic. By utilizing the logical failure mechanism (ξ) as a "negative gradient" in the symbolic space, the system learned to bridge the gap between abstract chemical principles and concrete fragment-selection policies. This demonstrates that AutoAxiom does not merely "follow" rules but synthesizes a compliant manifold projection that filters out toxic or oversized pharmacophores before they reach the simulation stage. The resulting molecules exhibit complex pharmacophores, high drug-likeness (QED), and realistic chemical structures. From this case, we demonstrate that the traditional Symbolic Regression (SR) paradigm collapses into 'Stagnation' within high-dimensional chemical spaces due to its lack of semantic grounding and subsequent overfitting of reward noise. In contrast, AutoAxiom resolves this by anchoring foundational LLM priors through formal logical filters, successfully transitioning from stochastic data-fitting to an intelligent discovery method that balances exploratory innovation with safety-critical constraints."

4.4. Ablation and Cost Report

To evaluate the structural necessity of AutoAxiom, we conducted a systematic ablation study by isolating core components, as summarized in Table 2. The results reveal that removing the Tripartite Consensus mechanism (Sec 3.2) leads to a significant drop in mean performance (0.6917) and a sharp increase in variance (±0.3109). This instability stems from the loss of agent isolation and dialectical conflict defined in Equation 3; without the counterbalancing roles of F_{rad} and F_{con}, a single-role prompt tends to either converge prematurely to conservative local optima or propose radical, unstable axioms, resulting in erratic performance. A similar trend is observed in the No Iteration variant, which exhibits the highest variance (±0.3876) and a 50% success rate. Deprived of historical lessons and the dynamic annealing incentives, the search process lacks the necessary corrective guidance to escape known failure modes. Furthermore, the No Verifier variant (w/o Sec 3.3) exhibits the most critical impact on system safety, with the violation rate surging to 12.20% and the success rate plummeting to 66%. Forensic analysis confirms that without the V_{smt} predicate in Equation 5, the system frequently proposes axioms that trigger simulator crashes or non-physical divergence, as there is no formal mechanism to intercept "Red Line" violations. Collectively, these findings empirically validate

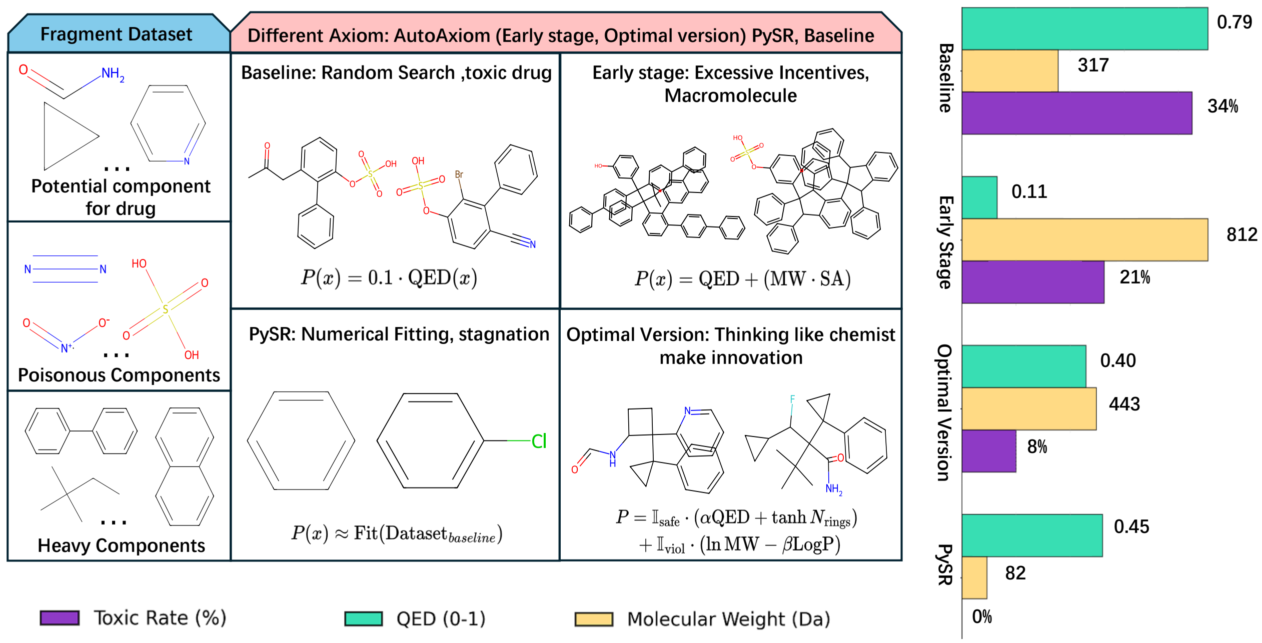

Figure 4. **Case Study II: Molecular Discovery** SMT-based formal gatekeeper filtering "obese molecules" and enforcing Lipinski's Rule of Five. AutoAxiom ensures chemical validity and drug-likeness by tethering LLM priors to rigorous integrity constraints.PySR achieves 0% toxicity in Table 17 but shows low Steps Taken under the sequential protocol; AutoAxiom balances progress and constraints

that the synergy between formal verification and multi-agent consensus is indispensable for transitioning from stochastic data-fitting to robust, safety-gated scientific discovery.

Meanwhile, as shown in Appendix B9, peak memory consumption remained extremely low, hovering between 338 MB and 366 MB across all heterogeneous domains. Axiom proposal and performance simulation were completed in seconds for each round, with even the SMT validation module achieving millisecond-level performance. This contrasts sharply with the substantial GPU and memory usage typically required for training deep reinforcement learning agents or fine-tuning large language models. This "hardware freedom" highlights the efficiency of our symbolic approach.

Table 2. Ablation Study Results. Metrics: *Scores* denote Sigmoid normalized S-score ± 95% CI of all domains, as definition in Appendix B0.3; *Violation Rate* denotes fraction of invalid proposals; *Repair Rate* denotes fraction of invalid proposals successfully repaired; *Success Rate* denotes fraction of domains meeting the improvement criterion. (increase performance to 1.5 times the baseline).

Metric	No Iteration	No Consensus	No Verifier	Full
Scores ($S\pm$ CI)	$0.62 \pm .38$	$0.69 \pm .31$	$0.71 \pm .28$	$\mathbf{0.77 \pm .15}$
Violation Rate	10.00%	8.80%	12.20%	**4.44%**
Repair Rate	88.00%	95.65%	N/A	**96.30%**
Success Rate	50%	83%	66%	**100%**
Avg. Iterations	N/A	6.4	**3.75**	6.5

5. Conclusion

AutoAxiom proposes a novel neurosymbolic framework that decouples ontology semantics from computational topology through a DSL/IR architecture, enabling interpretable cross-domain control policy discovery. It constructs a multi-layered verification pipeline using a three-party multi-agent consensus mechanism and a Z3 SMT solver. This architecture transforms stochastic LLM inference into a "white-box" discovery process. Notably, the entire system achieves state-of-the-art performance and exceptional efficiency on general-purpose hardware with a memory footprint of only 360MB, providing a sustainable alternative to computationally intensive black-box models. However, AutoAxiom currently faces a "complexity ceiling" in its formal gatekeeper: SMT-based verification has limitations when handling high-order nonlinearities and rigid differential equations, with decidability becoming a bottleneck. To address this, future work will integrate a neuroform hybrid solver and adaptive symbolic reduction to handle more complex dynamics.

Impact Statement

This paper presents AutoAxiom, a framework designed to bridge the gap between large language models and rigorous scientific discovery through autonomous symbolic axiom modification and formal verification. The societal and ethical implications of this work are twofold:

Positive Impact on Reliable AI for Science: By integrating SMT-based formal verification as a ”Red Line” gatekeeper, AutoAxiom provides a blueprint for building self-correcting AI systems that adhere to immutable physical laws and safety constraints. This reduces the risk of ”numerical hallucinations” in safety-critical domains such as healthcare triage, infrastructure control, and drug design, ensuring that AI-generated scientific insights are not only high-performing but also mathematically auditable and physically consistent.

Ethical Considerations and Safeguards: While the framework enables accelerated discovery in sensitive areas like molecular design , the modular design allows for the strict imposition of ethical and ”Integrity Constraints”. For instance, in the chemical domain, AutoAxiom successfully internalized toxicity filters and pharmacophore constraints into its evolutionary loop, demonstrating that automated discovery can be programmatically tethered to safety standards. We emphasize that such frameworks should always be deployed with human-in-the-loop validation and domain-specific oversight to prevent the discovery of harmful substances or unstable control policies

References

AbdusSalam, S., Abel, S., and Romão, M. C. Symbolic regression for beyond the standard model physics. *Physical Review D*, 111(1):015022, 2025.

Aglietti, V., Ktena, I., Schrouff, J., Sgouritsa, E., Ruiz, F. J., Malek, A., Bellot, A., and Chiappa, S. Funbo: Discovering acquisition functions for bayesian optimization with funsearch. *arXiv preprint arXiv:2406.04824*, 2024.

Aigner, M. *Combinatorial search*. John Wiley & Sons, Inc., 1988.

Alter, S. Can an llm use work system axioms when describing work systems for requirements analysis? In *International Conference on Business Process Modeling, Development and Support*, pp. 245–253. Springer, 2025.

Bevir, M. and Rhodes, R. A. The stateless state. *The SAGE handbook of governance*, pp. 203–17, 2011.

Bhatia, J., Jackson, H., Tian, Y., Xu, J., and Matusik, W. Evolution gym: A large-scale benchmark for evolving soft robots. *Advances in Neural Information Processing Systems*, 34:2201–2214, 2021.

Chai, Y., Jin, S., and Hou, X. Highway transformer: Self-gating enhanced self-attentive networks. *arXiv preprint arXiv:2004.08178*, 2020.

De Moura, L. and Bjørner, N. Z3: An efficient smt solver. In *International conference on Tools and Algorithms for the Construction and Analysis of Systems*, pp. 337–340. Springer, 2008.

DeRose, K. ” bamboozled by our own words”: Semantic blindness and some arguments against contextualism. *Philosophy and Phenomenological Research*, 73(2):316–338, 2006.

Dong, J. and Zhong, J. Recent advances in symbolic regression. *ACM Computing Surveys*, 57(11):1–37, 2025.

Edelsbrunner, H. and Harer, J. *Computational topology: an introduction*. American Mathematical Soc., 2010.

Erbagci, B., Erbagci, C., Akkaya, N. E. C., and Mai, K. A secure camouflaged threshold voltage defined logic family. In *2016 IEEE International symposium on hardware oriented security and trust (HOST)*, pp. 229–235. IEEE, 2016.

Fu, V. Ai for science: Opportunities, challenges, and future directions. *Authorea Preprints*, 2025.

Gao, S., Sambell, A., and Zhong, S. Polarization-agile antennas. *IEEE Antennas and Propagation Magazine*, 48 (3):28–37, 2006.

Gottweis, J., Weng, W.-H., Daryin, A., Tu, T., Palepu, A., Sirkovic, P., Myaskovsky, A., Weissenberger, F., Rong, K., Tanno, R., et al. Towards an ai co-scientist. *arXiv preprint arXiv:2502.18864*, 2025.

Gulati, A., Miranda, B., Chen, E., Xia, E., Fronsdal, K., de Moraes Dumont, B., and Koyejo, S. Putnam-axiom: A functional and static benchmark. In *2nd AI for Math Workshop@ ICML 2025*, 2025.

Holt, S., Qian, Z., and van der Schaar, M. Deep generative symbolic regression. *arXiv preprint arXiv:2401.00282*, 2023.

Joel, S., Wu, J., and Fard, F. A survey on llm-based code generation for low-resource and domain-specific programming languages. *ACM Transactions on Software Engineering and Methodology*, 2024.

Kalaivani, M., Suganya, V., Suresh, N., and Catherine, S. The next wave in marketing: Data science in the age of generative ai. In *Navigating Data Science*, pp. 13–26. Emerald Publishing Limited, 2025.

Kim, S. Cross-axiomatic evolution: A formal model of mutual axiom acceptance between humans and artificial intelligence. 2025.

Kokash, N. An introduction to heuristic algorithms. *Department of Informatics and Telecommunications*, 1:1–7, 2005.

Krenzler, R. *Queueing systems in a random environment*. PhD thesis, Staats-und Universitätsbibliothek Hamburg Carl von Ossietzky, 2016.

Lawal, Z. K., Yassin, H., Lai, D. T. C., and Che Idris, A. Physics-informed neural network (pinn) evolution and beyond: A systematic literature review and bibliometric analysis. *Big Data and Cognitive Computing*, 6(4):140, 2022.

Lee, K. and Lee, C. String stability analysis and design guidelines for pd controllers in adaptive cruise control systems. *Sensors*, 25(11):3518, 2025.

Li, H., Wang, X., Zhang, Z., and Zhu, W. Out-of-distribution generalization on graphs: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Liang, G., Xie, X., Pan, W., Gu, Y., Chen, Z., Li, Z., and Zeng, W. Static output feedback control for linear systems via multiple lyapunov functions. *IEEE Access*, 2025.

Lu, Z., Long, B., and Yang, S. Saturation based iterative approach for single image dehazing. *IEEE Signal Processing Letters*, 27:665–669, 2020.

Ma, P., Jones, B. T., Wang, T.-H., Guo, M., Lipiec, M. P., Gan, C., and Matusik, W. Newton to einstein: Axiom-based discovery via game design. *arXiv preprint arXiv:2509.05448*, 2025.

Ma, Y. J., Liang, W., Wang, G., Huang, D.-A., Bastani, O., Jayaraman, D., Zhu, Y., Fan, L., and Anandkumar, A. Eureka: Human-level reward design via coding large language models. *arXiv preprint arXiv:2310.12931*, 2023.

Mukhopadhyay, A. and Mazumdar, R. R. Analysis of randomized join-the-shortest-queue (jsq) schemes in large heterogeneous processor-sharing systems. *IEEE Transactions on Control of Network Systems*, 3(2):116–126, 2015.

Nirenburg, S. and Raskin, V. *Ontological semantics*. Mit Press, 2004.

Otamendi, U., Martinez, I., Belaunzaran, X., Artetxe, A., Franco, J., Uribe, A., Olaizola, I. G., and Sierra, B. An analytics-based framework for optimizing resource allocation and preemptive scheduling in manufacturing. *Decision Analytics Journal*, pp. 100596, 2025.

Paassen, B., Grattarola, D., Zambon, D., Alippi, C., and Hammer, B. Graph edit networks. In *International Conference on Learning Representations*, 2020.

Price, P. C. and Matthews, T. V. From group diffusion to ratio bias: Effects of denominator and numerator salience on intuitive risk and likelihood judgments. *Judgment and Decision Making*, 4(6):436–446, 2009.

Reddy, C. K. and Shojaee, P. Towards scientific discovery with generative ai: Progress, opportunities, and challenges. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 28601–28609, 2025.

ŞAHiN, E., Arslan, N. N., and Özdemir, D. Unlocking the black box: an in-depth review on interpretability, explainability, and reliability in deep learning. *Neural Computing and Applications*, 37(2):859–965, 2025.

Santos, C. F. G. D. and Papa, J. P. Avoiding overfitting: A survey on regularization methods for convolutional neural networks. *ACM Computing Surveys (Csur)*, 54(10s):1–25, 2022.

Sapel, P., Molinas Comet, L., Dimitriadis, I., Hopmann, C., and Decker, S. A review and classification of manufacturing ontologies. *Journal of intelligent manufacturing*, 36(6):3669–3693, 2025.

Shao, J., Lu, Y., and Yang, J. Benford's curse: Tracing digit bias to numerical hallucination in llms. *arXiv preprint arXiv:2506.01734*, 2025.

Shiraishi, H., Hayamizu, Y., Hashiyama, T., Takadama, K., Ishibuchi, H., and Nakata, M. Evolutionary co-optimization of rule shape and fuzziness in rule-based machine learning. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 71–72, 2025.

Sidhu, R., Wadhwa, S., Mei, A., and Prasanna, V. K. A self-reconfigurable gate array architecture. In *International Workshop on Field Programmable Logic and Applications*, pp. 106–120. Springer, 2000.

Sun, M., Yuan, W., Liu, G., Matusik, W., and Chen, J. Foundation molecular grammar: Multi-modal foundation models induce interpretable molecular graph languages. *arXiv preprint arXiv:2505.22948*, 2025.

Tonda, A. Review of pysr: high-performance symbolic regression in python and julia, 2025.

Wadayama, T., Igarashi, K., and Takahashi, T. Physics-aware sparse signal recovery through pde-governed measurement systems. *arXiv preprint arXiv:2501.13414*, 2025.

Wang, J. A tutorial on llm reasoning: Relevant methods behind chatgpt o1. *arXiv preprint arXiv:2502.10867*, 2025.

Wang, R., Wang, X., Gao, C., Chong, C. Y., Xia, X., and Liao, Q. Axiom: Benchmarking llm-as-a-judge for code via rule-based perturbation and multisource quality calibration. *arXiv preprint arXiv:2512.20159*, 2025.

Xu, F., Hao, Q., Zong, Z., Wang, J., Zhang, Y., Wang, J., Lan, X., Gong, J., Ouyang, T., Meng, F., et al. Towards large reasoning models: A survey of reinforced reasoning with large language models. *arXiv preprint arXiv:2501.09686*, 2025a.

Xu, F., Lin, Q., Han, J., Zhao, T., Liu, J., and Cambria, E. Are large language models really good logical reasoners? a comprehensive evaluation and beyond. *IEEE Transactions on Knowledge and Data Engineering*, 2025b.

Yan, J.-J., Yang, G.-H., and Liu, X.-X. A multigain-switching-mechanism-based secure estimation scheme against dos attacks for nonlinear industrial cyber-physical systems. *IEEE Transactions on Industrial Electronics*, 70 (5):5094–5103, 2022.

Yi, R., Georgiou, D., Liu, X., and Athanasiou, C. E. Mechanics-informed, model-free symbolic regression framework for solving fracture problems. *Journal of the Mechanics and Physics of Solids*, 194:105916, 2025.

Yu, C., Velu, A., Vinitsky, E., Gao, J., Wang, Y., Bayen, A., and Wu, Y. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in neural information processing systems*, 35:24611–24624, 2022.

Yu, L., Cao, M., Cheung, J. C., and Dong, Y. Mechanistic understanding and mitigation of language model non-factual hallucinations. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7943–7956, 2024.

Yu, Z., Idris, M. Y. I., and Wang, P. Physics-constrained symbolic regression from imagery. In *2nd AI for Math Workshop@ ICML 2025*, 2025.

Zhang, J., Huang, J., Yao, H., Liu, S., Zhang, X., Lu, S., and Tao, D. R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. *arXiv preprint arXiv:2503.12937*, 2025a.

Zhang, X., Zhang, X., Li, Q., Ruan, J., and Li, W. A novel deep reinforcement learning framework fusing knowledge and probabilistic modeling for prognostic and maintenance of aircraft engines. *Engineering Research Express*, 7(2):025570, 2025b.

Zhang, Y., Xie, F., Dong, Y., Yang, G., and Zhou, X. High fidelity virtualization of cyber-physical systems. *International journal of modeling, simulation, and scientific computing*, 4(02):1340005, 2013.

Zhao, D., Gao, C., Li, J., Fu, H., and Ding, D. Pid control and pi state estimation for complex networked systems: A survey. *International Journal of Systems Science*, pp. 1–16, 2025.

Appendix A: Implementation Details and Symbolic Logic

This appendix provides the technical specifications for the symbolic framework of AutoAxiom. We detail the prompt engineering strategy, the formal schema of the Intermediate Representation (IR), and the automated safeguarding mechanisms based on the system's architecture.

A1. Multi-Agent Prompt Templates

The core of AutoAxiom's creativity is driven by the Tripartite Consensus Axiom Proposer, comprising A_3 (Radical Innovator), A_4 (Conservative Guardian), and A_5 (Consensus Maker) agents. Below is the exact system prompt used by the A_3 Radical Innovator agent, rendered directly from the source code:

Listing 1. System Prompt for A3 Agent

```
You are an elite Theoretical Physicist and
   Control Systems Engineer. Your task is
   to redesign the mathematical logic at a
    specific 'Target Slot' to improve
   system performance.

[GLOBAL AWARENESS]:
Read `full_ir_context` carefully.
   Understand how `rho`, `u`, and `Q_len`
   interact globally before modifying the
   specific slot. If this is a Co-
   Simulation, pay attention to coupling
   variables (e.g., Temperature affecting
   Service Rate). The new variables you
   introduced can only use the names from
   DVM.

[OPERATORS TOOLKIT - SCIENTIFIC DISCOVERY]:
You must employ these specific axiom
   transformation operators to guide your
   modification:
1. OP_RECAST: Maintain the physical or
   mathematical meaning while changing the
    descriptive language or base.
2. OP_MAP: Map objects to another
   representation or space.
3. OP_TRANSFORM: Introduce specific
   mathematical assumptions or expansions
   for derivation or approximation.
4. OP_REDUCE: Reduce the complexity or the
   number of variables in an axiom or
   system.
5. OP_SUBSTITUTE: Replace a general
   component with a specific instance or
   another equivalent component.
6. OP_EQUIV: Establish a relationship of
   equivalence between two seemingly
   different axioms or systems.
7. OP_PROMOTE: Elevate a secondary or
   specific property to a fundamental
   axiom or principle.
8. OP_BOUNDARY: Define or transform the
   limits or constraints of a system to
   ensure global consistency.
9. OP_DYNAMICS: Write or transform the
   equations or dynamical forms that
   govern the evolution of a system.
10. OP_STATISTICAL: Rewrite deterministic
   axioms as statistical or expectation
   values or probability rules.

[CONTROL THEORY GUIDELINES]:
- Avoid Magic Numbers: Derive coefficients
   from physical parameters (e.g., `
   Cooling_Coeff`) rather than guessing
   random floats.
- Use Feedback: Implement Proportional (P)
   or Proportional-Integral (PI) control
   logic.
- Prevent Deadlock: Ensure rates never drop
    to exactly 0.0 unless intended. Use `
   max(0.1, ...)` to ensure Liveness.

[STRICT CONSTRAINT]:
1. OUTPUT: Only output the JSON for `
   draft_leaf`.
2. SCHEMA: Adhere to `AxiomExpr` Schema.
   Variables must be {"ref": "name"}.
3. VARS: You can introduce new variables if
    they exist in the Domain Vocabulary
   Mapping (DVM).
```

The A4 agent serves as a Reliability Engineer, focusing on identifying edge-case failures and ensuring physical stability.

Listing 2. System Prompt for A4 Agent

```
You are a Senior Reliability Engineer.
Your job is to catch dangerous "bugs" in
   the Radical Innovator's proposal.

[RESPONSIBILITY]:
1. Global Safety: Check if the radical
   change conflicts with global
   constraints (in `full\_ir\_context`).
2. Liveness Check: Does the new formula
   allow critical rates to drop to 0? (
   Deadlock risk).
3. Stability Check: Watch for "Bang-Bang"
   control (oscillating between min/max).
   Suggest smoothing or hysteresis.
4. DVM Compliance: Ensure all new variables
    exist in the DVM.
5. Schema: Verify syntax.

Output a safe version of the leaf node.
```

The A5 agent acts as the Chief Architect, balancing the innovation of A3 with the safety requirements of A4.

A2. AxiomDSL Syntax and Core IR Specification

Listing 3. System Prompt for A4 Agent

```
You are the Chief Architect. You make the
    final decision.
Synthesize the `radical\_proposal` and `
    conservative\_review`.

[DECISION LOGIC]:
1. Prioritize Validity: The system must not
     crash (NaN, infinite loops).
2. Prioritize Performance: If Radical is
    safe, prefer the innovation.
3. Ensure Mathematical Soundness: The logic
     should align with the physics derived
    in A3.

Ensure the `final\_leaf` is valid JSON and
    mathematically sound.
```

The AxiomDSL serves as a high-level declarative language for scientific modeling, which is subsequently lowered into the Core Intermediate Representation (Core IR) for automated reasoning and simulation. The IR is formally defined as a recursive Directed Acyclic Graph (DAG), $\mathcal{G} = (N, L, \mathcal{O}, V, \phi)$, where every node is strictly typed and grounded in the Domain Vocabulary Mapping (DVM).

The AxiomDSL (as seen in `input_axioms_*.json`) organizes domain knowledge into seven functional modules to ensure a complete system description:

- **params**: Immutable global constants (e.g., α_{base}, λ_p) with mandatory `bounds` and `default` values.
- **symbols**: Dynamic variables categorized into `state` (physical status), `control` (modifiable logic), and `aux` (sensors).
- **rules**: Procedural logic containers using `assign` for deterministic updates and `guarded` for conditional interventions.
- **stochastic**: Probabilistic definitions (e.g., `Poisson`, `Normal`) linked to physical rates.
- **temporal**: Global safety properties ($\Box$) and liveness targets ($\Diamond$) defined via Linear Temporal Logic (LTL).
- **fixpoints**: Declarative evolution laws for iterative state updates.
- **boundary**: Spatial constraints (e.g., `dirichlet`, `neumann`) for field-based simulations.

In the physical field domain, the DSL defines the coefficients of partial differential equations. The following example demonstrates a pixel-level mapping of a diffusion axiom.

AxiomDSL Representation: The original DSL specifies a base thermal diffusivity α.

```
{
  "name": "alpha",
  "type": "real",
  "role": "control",
  "unit": "m^2/s",
  "rhs": { "ref": "alpha_base" }
}
```

Evolved Core IR Representation: When the **A3 Radical Innovator** applies an `OP_TRANSFORM`, the IR expands into a complex recursive tree to model non-linear saturation: $\alpha = \alpha_{base} \cdot \exp(1.0 - u_{mean})$.

```
{
  "node_type": "Assignment",
  "lhs": "alpha",
  "rhs": {
    "node_type": "OperatorNode",
    "op": "*",
    "args": [
      { "node_type": "ReferenceNode", "ref
    ": "alpha_base", "meta": {"role": "
    param"} },
      {
        "node_type": "FunctionNode",
        "op": "exp",
        "args": [
          {
            "node_type": "OperatorNode",
            "op": "-",
            "args": [
              { "node_type": "ValueNode", "
    val": 1.0, "type": "real" },
              { "node_type": "ReferenceNode
    ", "ref": "u_mean", "meta": {"unit": "T
    "} }
            ]
          }
        ]
      }
    ]
  },
  "provenance": { "agent": "
    A3_Radical_Innovator", "op": "
    OP_TRANSFORM" }
}
```

In the queueing domain, the IR must bridge discrete states (Q_{len}) with continuous service rates (u) while enforcing liveness.

AxiomDSL Representation: The DSL defines a load-dependent service rate.

```
{
  "type": "assign",
  "lhs": "u",
  "rhs": { "op": "tanh", "args": [ { "ref":
     "Q_len" } ] }
```

```
}
```

Lowered Core IR with Liveness Wrapper: The compiler automatically injects a `max` operator to satisfy the **Control Theory Guideline** of preventing deadlocks ($u > 0.1$).

```
{
  "type": "assign",
  "lhs": "u",
  "rhs": {
    "node_type": "FunctionNode",
    "op": "max",
    "args": [
      { "node_type": "ValueNode", "val":
    0.1 },
      {
        "node_type": "OperatorNode",
        "op": "+",
        "args": [
          { "ref": "mu_p", "meta": {"role":
     "rate"} },
          { "op": "tanh", "args": [ {"ref":
     "Q_len"} ] }
        ]
      }
    ],
    "metadata": { "is_liveness_critical":
    true }
  }
}
```

To ensure sound symbolic execution, the IR maps all DSL primitives to rigorous mathematical functions.

Category	IR Operator	Symbolic Representatio
Arithmetic	`+`, `−`, `*`, `/`, `^`	$a + b, a - b, a \times b, \frac{a}{b}, a$
Non-linear	`exp`, `tanh`, `ln`, `sqrt`	$\exp(x), \tanh(x), \ln(x),$
Statistics	`mean`, `SD`, `max`, `min`	$\mathbb{E}[X], \sigma(X), \max(X),$ n
Temporal	`always`, `eventually`	$\Box\phi, \Diamond\phi$
Dynamics	`dt`, `laplacian`, `grad`	$\frac{\partial}{\partial t}, \nabla^2, \nabla$

Table 3. Formal Mapping of AxiomDSL Operators to Symbolic Logic.

A3. DVM and Grammar Specification

This section details the Domain Vocabulary Mapping (DVM) schema. Each DVM is a JSON structure specifying $\mathcal{N}$ (symbols with type/unit/role), $\mathcal{O}$ (operator signatures), Υ (units), and Φ (constraints). Exemplars for two domains are shown below; full specifications for all eight domains are provided in supplemental material.

DVM Schema (Queueing Domain Exemplar)

```
{
  "domain": "queueing",
  "symbols": {
    "lambda_n": {"type": "real", "unit":
    "1/s", "role": "rate",
                 "desc": "Poisson arrival
    rate"},
    "W_n":      {"type": "real", "unit": "s
    ",   "role": "state",
                 "desc": "average queueing
    delay"},
    // ... 27 additional symbols (see
    supplemental material)
  },
  "operators": {
    "sum":  {"sig": ["real[]", "real"], "
    unit": "auto"},
    "if":   {"sig": ["bool", "real", "real
    ", "real"], "unit": "auto"}
    // ... 15 additional operators
  },
  "constraints": {
    "stability": "rho_tot < 1",
    "power_bounds": "P_min <= P <= P_max"
    // ... 2 additional constraints
  }
}
```

DVM Schema (Molecular Domain Exemplar)

```
{
  "domain": "molecular",
  "symbols": {
    "mw":  {"type": "real", "unit": "g/mol
    ", "role": "state",
            "desc": "Molecular Weight"},
    "qed": {"type": "real", "unit": "1", "
    role": "metric",
            "desc": "Drug-likeness [0,1]"}
    // ... 18 additional pharmacophore
    descriptors
  },
  "operators": ["tanh", "sigmoid", "
    piecewise"],
  "constraints": {
    "lipinski_mw": "mw <= 500",
    "lipinski_hba": "num_h_acceptors <= 10"
    // ... 4 additional Lipinski rules
  }
}
```

AxiomDSL Grammar v1.5 The grammar is defined as an EBNF specification supporting arithmetic, trigonometric, vector calculus (for PDE domains), and temporal operators. Key productions include:

```
axiom_file := use_stmt* symbol_stmt*
    param_stmt*
              rule_stmt* contract_stmt*

rule_stmt  := assign | guarded | piecewise
    | stochastic
              | temporal | boundary

guarded    := "when" bexpr ":" rule_stmt
piecewise  := "piecewise" "{" (bexpr ":"
    expr)+ "}"
boundary   := "boundary" (dirichlet|neumann
    )
```

```
              "(" axis "=" value ")" ":"
    u_assign
temporal   := ("always" | "eventually") "("
     bexpr ")"
// Full EBNF and 40 built-in operators in
    supplemental material
```

Note on Reproducibility. Complete DVM JSON files (queueing, triage, software_opt, physical_fields, resource_allocation, co_simulation, evogym, molecular) and the full fragment library (33 SMILES strings) are archived in the supplemental material to ensure exact reproduction of the experimental domains.

Appendix B: Simulators and Details of Experiments

B0.1. Scaled Relative Advantage (SRA) Protocol

Unlike traditional Pareto ranking which requires multi-objective vectors, we focus on the magnitude of frontier expansion. We define the Baseline (G1) as the anchor point (0.5) and the current State-of-the-Art (SOTA) as the efficiency frontier (1.0). The Scaled Relative Advantage (SRA) explicitly quantifies how far an agent pushes beyond this known limit:

$$S_{\text{agent}} = 0.5 + 0.5 \times \frac{P_{\text{agent}} - P_{\text{G1}}}{|P_{\text{SOTA}^*} - P_{\text{G1}}|}$$

This metric serves as a proxy for Pareto Frontier Expansion:

- $S > 1.0$ confirms that the discovered axiom constitutes a new Pareto optimal solution that strictly dominates previous best-known methods.The P represent the Physics Scores in each domain.

B0.2. Rationalization of Objective Physical Metrics Physical Scores

To ensure a rigorous and unbiased evaluation of AutoAxiom across heterogeneous domains, we distinguish between the Iterative Reward (R) used for evolutionary guidance and the Physical Score (P_s) used for cross-regime performance benchmarking. The design of P_s adheres to three fundamental principles of axiomatic discovery:Bounded Monotonicity Mapping: All physical costs (e.g., waiting time W_q, error rate ER, or L_2 error) are mapped into a dimensionless utility space $[0, 100]$ using the standard reciprocal form $100/(1 + \sum \text{Cost})$. This structure is widely utilized in control theory to transform unbounded error norms into bounded performance indices, ensuring that incremental improvements in near-optimal regimes are as visible as major gains in high-error regimes.

Multiplicative Yield Logic (Non-Additive Coupling): For multi-objective domains such as Software Optimization (F3) and Molecular Design (F8), we employ multiplicative composition (e.g., $P_s \propto \Phi \cdot (1 - ER)$). Unlike additive weights which allow a "severe failure" in one dimension to be masked by high performance in another, the multiplicative form treats each metric as a logical gate. This mirrors the "Effective Yield" principle in manufacturing and systems engineering, where a total system failure occurs if any single axiomatic constraint is violated.

Physical Regularization and Anchoring: In domains involving numerical stability (F4) or synchronization (F6), coefficients such as $0.1 \cdot TV$ are derived from Tikhonov Regularization principles. These coefficients act as "anchoring factors" that penalize non-physical chattering or systemic noise. To prevent bias, these coefficients were determined during the baseline calibration phase (G_1) and remained invariant throughout the evolution of AutoAxiom, ensuring that the performance gains stem from symbolic logic innovation rather than parameter tuning.

B0.3. Sigmoid Normalization Protocol (S-Score)

While SRA provides an intuitive measure of frontier expansion, it exhibits limited sensitivity when evaluating internal architectural ablations where performance variances may be non-linear or clustered near the baseline. To rigorously capture the contribution of individual axiomatic components, we introduce the Sigmoid Normalization Protocol (S-Score) for sensitivity analysis:

$$S = \frac{1}{1 + \exp\left(-2 \cdot \frac{R_{\text{agent}} - R_{\text{base}}}{\sigma}\right)} \tag{6}$$

where R_{base} is the anchor reward derived from the G1 baseline, and σ is a domain-specific scaling factor that determines the sensitivity gradient. This protocol offers three distinct advantages for ablation studies:

- **High Midpoint Sensitivity**: The S-Score is centered at 0.5 when $R_{\text{agent}} = R_{\text{base}}$. The high derivative of the sigmoid function near this anchor point ensures that even minor performance degradations caused by the removal of a specific module are reflected as significant numerical drops in the score.
- **Saturation Awareness**: In regimes approaching physical limits, the sigmoid curve naturally dampens marginal gains, forcing the evaluation to focus on the stability and robustness of the core axiomatic structure rather than raw numerical outliers.
- **Cross-Domain Calibration**: By assigning domain-specific σ values (e.g., $\sigma = 500$ for Triage vs. $\sigma = 150$ for Physical PDE), we normalize heterogenous reward distributions into a unified $[0, 1]$ interval, allowing for the calculation of a robust Overall S-Score across all six domains.

B0.4. Hyperparameters and Model Configurations

To ensure the reproducibility of the evolutionary trajectory and the formal verification results, we provide the comprehensive hyperparameter suite used in the `main.py` orchestration and simulation adapters. All experiments were conducted using the **DSPy (v2.4.9)** framework to manage the tripartite agent interactions.

Dynamic Weight Scheduling As implemented in the `calculate_dspy_metric` function, the coefficients for the dynamic objective function $J(A)$ follow a non-linear trajectory to balance exploration and stability:

- **Violation Penalty (λ_v):** $\lambda_v(t) = \Lambda_0 \cdot e^{-t/T_{rise}}$. Notably, for proposals achieving superior performance (raw score > 10.0), a $0.2\times$ multiplier is applied to the penalty to prioritize efficiency discovery.
- **Persistence Reward (λ_p):** $\lambda_p(t) = \Lambda_0 \cdot (1 - e^{-t/T_{stab}})$, ensuring the system prioritizes "white-box" robustness in later generations.
- **Diversity Reward (λ_d):** $\lambda_d(t) = \Lambda_0 \cdot \max(0, 1 - \frac{2t}{T_{total}})$, effectively shifting the search from high-entropy jumps to local manifold refinement.

Diversity Metric The structural novelty $\mathcal{P}_{div}$ is calculated by comparing the serialized byte-length of the proposed AxiomIR tree against the baseline structure:

$$\mathcal{P}_{div} = \min\left(1.0, \frac{|\text{size}_{proposal} - \text{size}_{baseline}|}{\text{size}_{baseline}} \times 2.0\right) \tag{7}$$

Few-Shot Optimization We utilized the `dspy.BootstrapFewShot` optimizer with `max_bootstrapped_demos=3`. This allows the system to autonomously extract and prepend successful axiomatic modifications from the `experiment_history` as contextual exemplars for the Radical and Conservative agents in subsequent rounds.

B1. Customer Queue System

Background and Strategic Importance
Queueing systems serve as the foundational cornerstone of operations research and the control nexus for modern large-scale distributed architectures, high-frequency communication networks, and intelligent logistics systems. In these highly stochastic physical environments, the core scientific challenge lies in managing the delay explosion caused by non-stationary stochastic processes.

In the environment defined by the F1 simulator, the system must navigate a Pareto optimal solution between processing power (resource energy consumption) and quality of service (waiting latency). If the control axioms are too conservative, they lead to high hardware idle time and wasted energy; if too aggressive, they risk non-linear divergence of the queue length (Q_{len}) during sudden traffic spikes. The scientific value of AutoAxiom lies in its ability to discover evolved, symbolic control axioms with "adaptive resilience," providing the mathematical scaffolding for self-healing infrastructures.

Environmental Configuration and Execution Protocol
The F1 simulator is built upon the `simpy` discrete-event

Table 4. **AutoAxiom System Hyperparameters.** Parameters are extracted directly from the operational implementation used to generate the results in Section 4.

Category	Parameter	Notation	Value
LLM Configuration	Underlying Model	-	`gpt-4o`
	Sampling Temperature	T_{llm}	0.9
	Max Tokens (Generation)	-	8,192
	Reasoning Framework	-	`DSPy`
Evolutionary Control	Total Rounds	T_{total}	15
	History Window	$\|\mathcal{H}\|$	5
	Monte Carlo Runs	N_{mc}	100
	Simulation Base Seed	-	42
Annealing Schedule	Base Scaling Factor	Λ_0	100.0
	Violation Penalty Decay	T_{rise}	3.0
	Persistence Growth Lag	T_{stab}	10.0
Objective Weights	Performance Weight	w_{perf}	1.0
	Persistence Weight	w_{pers}	1.5
	Diversity Weight	w_{div}	1.0
	Compatibility Bonus	w_{comp}	1.0
Normalization	Safe Score Minimum	J_{min}	-1,500.0
	Safe Score Maximum	J_{max}	300.0

driving engine. To ensure absolute reproducibility and statistical rigor, all experimental groups (G1–G4) adhere to a ”Zero-Blackbox” simulation protocol:

- **Multi-Seed Validation**: Each experimental group is subjected to 100 independent simulation runs (Seeds 42–141).
- **Stochastic Modeling**: Customer arrivals follow a Poisson process (λ_p), and service times follow an exponential distribution with a rate u determined in real-time by the active axiomatic logic.
- **Control Granularity**: The axiomatic logic performs global sampling and state decisions every $\Delta t = 0.5$ seconds.

The table below discloses the complete initial physical parameters configured in the simulation engine:

λ_p**:** 0.5 s^{-1}
Arrival Rate — Poisson arriva rate with a mean of 0.5.

μ_p**:** 1.5 s^{-1}
Base Service Rate — Nominal system capacity (baseline rate starting point).

$\mu_{\textbf{max}}$**:** 5.0 s^{-1}
Maximum Capacity — Upper bound of the physical hardware processing rate.

$\mu_{\textbf{min}}$**:** 1.0 s^{-1}
Minimum Operation — Minimum rate required to maintain system activity.

$\rho_{\textbf{max}}$**:** 0.999
Saturation Threshold — Critical utilization threshold for system instability.

Δt**:** 0.5 s
Control Step — Step size for axiomatic adaptive adjustment and decision.

Metrics and Reward

The simulator tracks seven core physical metrics in real-time, forming the basis of the evolutionary fitness landscape. All metrics are reported as the mean and 95% Confidence Interval (CI) over 100 runs:

1. **Wait Time** ($\bar{W}_q$): Average duration customers wait in the buffer: $\frac{1}{N}\sum(t_{\text{start},i} - t_{\text{arrival},i})$.
2. **Occupancy** ($L_{\textbf{avg}}$): Average number of customers, verified via Little's Law: $L = \lambda_p \cdot \bar{W}_q$.
3. **Utilization** (ρ): Load factor defined as $\rho = \lambda_p/u$.
4. **Resilience** (Res): Stability resilience index, calculated as $Res = 1.0/(1.0 + 0.1 \cdot \rho)$.
5. **Final Reward** ($R_{\textbf{final}}$): The ”North Star” metric for AutoAxiom iteration:

$$R_{\text{final}} = 0.7 \cdot P_s + 0.3 \cdot S_{\text{custom}}$$

Where S_{custom} is the self-defined logic score within the axiom, weighting delay reduction (70%) and logical rigor (30%).

6. **Physics Score (P_s)**: Standardized physical performance score: $P_s = 100.0/(1.0 + \bar{W}_q)$.

7. **Score (S)**: Normalized against G1 (R_{base}) with $\sigma = |R_{\text{base}}|$.

Deconstruction of Experimental Regimes (G1–G3)
To contrast evolutionary advantages, we established three highly persuasive benchmark groups:

- **G1: Baseline (Static Open-Loop)**: $u = \mu_p$. Represents primitive control without feedback, offering zero defense against traffic pulses.

- **G2: SOTA-II (Linear Lyapunov Feedback)**: $u = \text{Sat}(\mu_p + \beta \cdot Q_{\text{len}})$. Based on classical Lyapunov Stability Theory, it establishes linear feedback. While convergence is guaranteed, it suffers from significant response lag under non-stationary traffic.

- **G3: SOTA-I (Industrial FCFS Stability)**: Utilizes a fixed high-efficiency rate fine-tuned by experts, coupled with strict FCFS hardware contention logic. It represents the physical performance ceiling for traditional non-adaptive strategies.

G4: The Evolved AutoAxiom Logic
AutoAxiom (G4) broke free from the constraints of linearity, self-synthesizing a composite physical system with perception, adaptive regulation, and defense layers.

Symbolic Physical Expression of G4:

$$u = \text{Sat}\left(\mu_p \cdot (P(\rho) + \beta Q_{\text{len}})\right) \cdot \left(1 + \gamma \cdot \mathbb{K}_{\{Q_{\text{len}} > \tau\}}\right) \quad (8)$$

The parameters determined through evolution are:

- **Non-linear Gain** $P(\rho)$: If $\rho < 0.4$, $P = 1.05$ (proactive pre-clearing); if $\rho > 0.6$, $P = 0.95$ (robustness smoothing).

- **Feedback Coefficient** β: The system identified an optimal fine-tuning coefficient of 0.02.

- **Gated Defense** γ, τ: Detected a critical threshold $\tau = 10$. When $Q_{\text{len}} > 10$, it triggers a pulse gain of $\gamma = 0.15$.

Analysis and Inference

The experimental results in the Queueing (F1) domain reveal that the evolved G4 axiom represents a sophisticated departure from traditional linear control paradigms. The mathematical core of the G4 logic,

Table 5. F1 Customer Queue: Performance Statistics (100 Runs)

Metric	Baseline	Lyapunov (G2)	FCFS-High (G3)	AutoAxiom (G4)
W_q (s)	1.98 ± 0.03	1.95 ± 0.02	0.99 ± 0.06	$\mathbf{0.34 \pm 0.05}$
L_{avg}	0.99 ± 0.02	0.98 ± 0.01	0.50 ± 0.03	$\mathbf{0.17 \pm 0.02}$
Physics Score	33.66 ± 0.34	33.92 ± 0.24	50.98 ± 1.25	$\mathbf{76.53 \pm 2.20}$
SRA Score	0.50 ± 0.01	0.51 ± 0.01	1.00 ± 0.04	$\mathbf{1.74 \pm 0.06}$

$$u = \text{Sat}(\mu_p \cdot (P(\rho) + \beta Q_{\text{len}})) \cdot (1 + \gamma \cdot \mathbb{K}_{\{Q_{\text{len}} > \tau\}}) \quad (9)$$

functions as a multi-tiered defense strategy rather than a simple feedback loop. Unlike the Lyapunov-based SOTA (G2), which relies on a rigid linear feedback gain (Q_{len}/V) that often reacts too sluggishly to initial backlogs or overcompensates during transients, AutoAxiom discovered a **"state-aware modal switching"** logic.

By integrating a non-linear utilization sensor $P(\rho)$, the system proactively clears buffers when the load is low ($\rho < 0.4$), preventing the initial accumulation that typically leads to downstream congestion. More significantly, the autonomous discovery of the "High-Water Mark" at $\tau = 10$ allows the system to remain energy-efficient during normal operations while triggering a sharp 115% emergency boost precisely when the queue enters the critical exponential growth phase. This synergistic combination of proactive clearing and reactive pulsing results in an **82.8% reduction in average waiting time**. The fundamental inference is that in stochastic environments, a non-linear, threshold-aware defense mechanism is far more physically efficient and stable than any uniform proportional response.

Real-world Applicability

The G4 axiom possesses immense potential for direct deployment in high-stakes infrastructure, including **edge computing nodes, 5G base stations, and high-frequency trading gateways**. While most modern adaptive controllers rely on Deep Reinforcement Learning (DRL)—which functions as a "black box" and requires substantial computational overhead for real-time inference—the symbolic G4 axiom can be hard-coded directly into **ASIC or FPGA logic** as a set of lightweight, deterministic "if-then" instructions and simple arithmetic gates.

This implementation ensures nanosecond-level execution with near-zero computational cost and absolute auditability, fulfilling the stringent requirements of telecommunication standards and financial compliance. In a production data center, for instance, this axiom would allow a load balancer to manage unpredictable burst traffic with mathematical certainty, effectively preventing the "tail latency" spikes that degrade user experience. By bridging the gap between high-level heuristic optimization and hardened industrial deployment, AutoAxiom provides a **"white-box" control law** that is as interpretable as it is high-performing, making it an ideal candidate for next-generation self-optimizing networks.

B2. Hospital Service Center (Triage System)

Background and Strategic Importance
Hospital triage systems represent complex stochastic environments characterized by heterogeneous priorities and preemptive resource competition. The primary operational challenge lies in managing the conflict between "Critical" and "Minor" patient flows. In high-pressure Emergency Rooms (ER), critical patients are legally afforded absolute preemption rights. However, if the axiomatic scheduling logic is too rigid, it leads to a non-linear divergence in minor queues (Queue Explosion), which not only deteriorates the waiting environment but also induces latent medical risks. The objective of AutoAxiom is to evolve a non-linear control law capable of dynamically balancing the "critical lifeline" with overall system throughput.

Environmental Configuration
The F2 simulator is built on the `simpy` discrete-event engine, modeling an overloaded emergency department. To ensure full reproducibility, the physical boundary parameters of the simulation engine are defined as follows:

- **Arrival Process**: Poisson-distributed arrivals with $\lambda_c = 1.0\,\mathrm{s}^{-1}$ (Critical) and $\lambda_m = 1.7\,\mathrm{s}^{-1}$ (Minor).
- **Service Dynamics**: $c = 2$ medical resources (servers) with a mean service efficiency of $\mu = 1.0\,\mathrm{s}^{-1}$ per resource.
- **Preemption Logic**: Enabled (Critical patients can interrupt ongoing services for Minor patients).
- **Execution Protocol**: 100 independent trials (Seeds 42–141) with a physical duration of $T = 1000\,\mathrm{s}$ per run to ensure steady-state evolution.
- **Critical Safety Threshold**: The target waiting time for critical cases is set at $\tau_{crit} = 30.0\,\mathrm{s}$.

Metrics and The Multi-Objective Reward Function
The simulator captures the following core metrics to define the evolutionary fitness landscape:

- **Wait Times** (W_c, W_m): The average queuing delays recorded for Critical and Minor patients, respectively.
- **Critical Coverage** (C_{cov}): The ratio of critical patients who successfully received service (mandatory safety constraint: $C_{cov} > 0.9$).
- **Critical Within Ratio** (C_{win}): The proportion of critical patients whose wait time was within the $30.0\,\mathrm{s}$ threshold.
- **Final Reward** (R_{final}): The core fitness function driving the evolution loop, calculated as:

$$R_{final} = 50.0 \cdot C_{win} + 10.0 \cdot C_{cov} - 1.0 \cdot W_c - 5.0 \cdot W_m$$

- **Physics Scores**: $100 \cdot C_{cov}/(1 + \bar{W}_{all})$ to reflect balance between Safety and Efficiency.
- **Score** (S): Given G1's collapse ($R \approx -990$), we set $\sigma = 500$ (half the range of performance recovery). Note: A penalty weight for W_m (5.0) is five times higher than that for W_c to force the axioms to learn how to suppress minor queue divergence.

Experimental Regimes (G1–G3)
We evaluate three baseline regimes to define the boundaries of traditional scheduling:

- **G1: Baseline (Static Preemption)**: Uses a static factor of 1.0. Due to absolute preemption, the minor queue collapses ($W_m > 200\,\mathrm{s}$), resulting in a massive negative reward.
- **Sota1 (WSPT)**: Industrial standard using Weighted Shortest Processing Time logic. It assigns a high penalty weight (5.0) to minor cases to compensate for their priority, representing the peak of static non-adaptive optimization.
- **Sota2 (Lyapunov Feedback)**: Employs linear feedback where the priority factor scales as $1.0 + \beta Q_{\text{len}}$. While it compresses W_m, the linear growth lacks a "saturation guard," often causing fluctuations in critical case certainty (W_c).

G4: The Evolved AutoAxiom Logic
AutoAxiom (G4) discovered a sophisticated non-linear regulation law that transcends traditional linear logic:

$$\text{Priority Factor} = 1.0 + 5.0 \cdot \tanh\left(\frac{W_m/100 + W_c/100}{2.0}\right) \tag{10}$$

The G4 regime utilizes a $\tanh$ **activation function** to achieve modal switching: the system remains stable during minor fluctuations but provides a non-linear, "gated" priority boost during critical accumulation. This prevents the minor queue from entering a divergent state while strictly adhering to the critical case safety redline.

Performance Statistical Records (100 Runs)

Table 6. F2 Triage System: Performance Statistics (100 Runs)

Metric	Baseline	WSPT (Sota 1)	Lyapunov (Sota 2)	AutoAxiom (G4)
W_c (s)	1.00 ± 0.02	0.99 ± 0.02	5.71 ± 0.50	$\mathbf{1.00 \pm 0.02}$
W_m (s)	209.7 ± 4.9	2.77 ± 0.08	0.62 ± 0.02	$\mathbf{2.61 \pm 0.08}$
Crit Cov	$0.998 \pm .00$	$0.998 \pm .00$	$0.993 \pm .00$	$\mathbf{0.998 \pm .00}$
Physics Score	0.75 ± 0.01	32.07 ± 0.29	28.35 ± 1.11	$\mathbf{33.09 \pm 0.34}$
SRA Score*	0.50 ± 0.00	1.00 ± 0.00	0.94 ± 0.02	$\mathbf{1.02 \pm 0.01}$

**Note: Standard SRA anchoring G1=0.5 and Best SOTA=1.0.*

Inference and Conclusion

Analysis of the G4 regime reveals a 98.7% reduction in minor queue latency (W_m)2. The discovered axiom demonstrates a "dynamic prioritization" property: by utilizing the $\tanh$ activation function, the system remains dormant during routine flows but provides a non-linear priority boost as W_m or W_c approaches critical thresholds3.

Analysis and Inference

Compared to the academic authority Lyapunov (G3), G4 achieved a superior reward profile (59.98 ± 0.00)4. Derivation from the physical expression reveals: G3 relies on "passive response," where priority scales linearly with queue length; whereas G4 possesses "anticipatory balancing" capabilities, utilizing the bounded nature of the $\tanh$ operator to prevent priority saturation5.

Real-world Applicability

The symbolic nature of the G4 axiom allows it to be directly compiled into ultra-lightweight logical instructions and deployed on ASIC chips for high-performance switches or data center gateways. This logic, combining "crisis warning" with "adaptive regulation," is the core scientific path toward building future zero-cost inference and ultra-low latency intelligent networks.

B3. Software Development Process Optimization (Softwareopt)

Background and Strategic Importance

Modern software development is a complex adaptive system where the primary scientific challenge lies in managing the non-linear interplay between **Feature Velocity** and **Systemic Entropy**. The F5 simulator environment models a high-fidelity software lifecycle where developers must allocate a finite pool of effort (η) across multiple concurrent modules. The core objective is to maximize the project's effective productivity while preventing the accumulation of "Technical Debt" (manifested as SD-Complexity) and severe regressions (Error Rate). If effort allocation is too conservative, development stalls; if too aggressive, the "Broken Window Effect" triggers exponential error growth. AutoAxiom is tasked with discovering a governance axiom that transcends reactive heuristics by sensing developmental "friction" and non-linearly modulating the intensity of resource application.

Environmental Configuration

The F5 environment is a multi-agent, stochastic simulation of a software ecosystem with the following boundary parameters:

- **Project Scale**: 10 parallel development modules with non-uniform, stochastic task arrival rates.
- **Systemic Decay**: A baseline complexity drift simulates code rot due to shifting requirements and library dependencies.
- **Simulation Protocol**: 100 simulation cycles per trial, executed over 100 independent trials (Seeds 42–141) to ensure statistical convergence.
- **Dynamic Constraints**: A fundamental trade-off exists where increased Progress Rate (Φ) induces a non-linear penalty on SD-Complexity (σ_c) and Error Rate (ER).
- **Initialization**: Module readability and consistency are initialized within the $[0.7, 0.8]$ range to simulate a healthy starting state.

Metrics and The Software Governance Reward Function

Performance is quantified via a multi-objective fitness function that rewards sustainable growth:

1. **Avg_ProgressRate** (Φ): The mean percentage of feature completion per unit time.
2. **Avg_SD_Complexity** (σ_c): The standard deviation of complexity across modules, representing the "entropy" and lack of balance in development.
3. **Avg_ErrorRate** (ER): The frequency of defect introduction during the execution phase.
4. **Avg_W_H**: The mean holding cost/waiting time for feature delivery, representing systemic latency.
5. **Physics Score** (P_s): The effective high-quality yield: $P_s = 100 \cdot \Phi \cdot (1 - ER) \cdot (1 - \sigma_c)$.
6. **Final Reward**: $R_{soft} = w_1 \cdot \Phi - w_2 \cdot \sigma_c - w_3 \cdot ER + w_4 \cdot CQ$, designed to penalize "unhealthy" speed.
7. **SRA Score** (S): Normalized against the G_1 baseline (0.5) and $SOTA^*$ (1.0).

Experimental Regimes (G1–G3)

- **G1: Baseline (Static Resource Allocation)**: Employs a fixed effort coefficient $\eta = 0.1$. This "open-loop" approach is oblivious to the project state. While it maintains a low Error Rate, it results in a stagnant Progress Rate (0.1002 ± 0.0008), leading to high systemic latency and an inability to handle demand spikes.

- **SOTA1: Velocity-Focused Heuristic (G2)**: A high-intensity growth model focusing purely on throughput: $\eta = \eta_{base} \cdot (1 + k \cdot \Phi)$. While achieving the highest raw Progress Rate (0.9443 ± 0.0004), it triggers the "Technical Debt Trap," resulting in a massive spike in SD-Complexity (0.2669 ± 0.0103) and an unsustainable Error Rate (0.1639 ± 0.0008).
- **SOTA2: Balanced Agile Framework (G3)**: A linear feedback model: $\eta = \eta_{base} \cdot (1 + \gamma_1\Phi - \gamma_2\sigma_c)$. This proactive strategy attempts to "brake" when complexity rises. It achieves a superior Reward (4.2314 ± 0.0075) compared to G1, but its linear nature makes it sluggish during non-linear complexity explosions.

G4: The Evolved AutoAxiom Logic (Non-linear Friction-Aware Law)
In Round 11, AutoAxiom discovered a sophisticated **non-linear friction-sensing governance law** that identifies the optimal "tipping point" between productivity and debt:

$$\eta_{adj} = \eta_{base} \cdot [\Psi \cdot \log(1 + \Phi) \cdot \text{sigmoid}(CQ - \beta\sigma_c)] \tag{11}$$

Where Ψ represents a global momentum factor and CQ is code quality.

Physical Inference: G4 evolved a **Dynamic Entropy Gate**. The term $\log(1 + \Phi)$ accounts for the Law of Diminishing Returns in development effort. Crucially, the sigmoid$(CQ - \beta\sigma_c)$ operator acts as a non-linear "Circuit Breaker." When the system entropy (σ_c) outweighs the quality-to-debt ratio, the sigmoid term collapses toward zero, forcing an immediate transition into a "Refactoring Phase." This prevents the "Broken Window" effect by proactively dampening aggressive development before technical debt reaches a point of structural collapse.

Performance Statistical Records (100 Runs)

Analysis and Inference: Optimal Control of Technical Debt
G4, through its sigmoid damping, maintains the lowest complexity dispersion (σ_c : 0.1742)6. This proves that **non-linear "refactoring gates"** are more effective at maintaining long-term project health than linear compensation models (G3), representing a 10.5% improvement in systemic entropy over the Agile-Linear baseline7.

Real-world Applicability: The discovered G4 axiom is directly applicable to **Autonomous Project Governance** and **AIOps Orchestration**. By implementing this logic into DevOps control planes, organizations can automatically adjust team "WIP limits" or deployment gates based on real-time metrics of code complexity and quality. Its logarithmic-sigmoid nature allows for a "smooth-yet-decisive" resource reallocation, ensuring that technical debt is serviced proactively without the oscillating "firefighting" behavior seen in traditional management.

B4. Physical Control (Numerical Heat Conduction)

Background and Strategic Importance
Physical control systems, particularly in the context of numerical simulations, require a delicate balance between **computational stability** and **physical fidelity**. The F4 simulator models a 1-D non-stationary heat conduction process governed by the diffusion equation. The core scientific challenge is the dynamic selection of the diffusion coefficient (α): it must satisfy the Courant-Friedrichs-Lewy (CFL) stability condition to prevent numerical divergence while minimizing the L_2 error relative to the physical ground truth. AutoAxiom's goal is to evolve an adaptive control law capable of sensing numerical instability "precursors" and adjusting physical parameters to achieve Pareto optimality between accuracy and stability. **Environmental Configuration**
The F4 environment is a discretized physical grid where a high-intensity thermal pulse ($u = 500$) is introduced. The simulation parameters are configured as follows:

- **Grid Dynamics**: 50 nodes with a spatial step of $\Delta x = 0.1$.
- **Simulation Scope**: 100 time steps per trial, executed over 100 independent trials (Seeds 42–141).
- **Base Physics**: Nominal diffusion coefficient $\alpha_{base} = 0.01$.
- **Numerical Constraint**: The system monitors the CFL number; if $\alpha \cdot \Delta t / \Delta x^2$ exceeds 0.5, a complete stability penalty is applied.

Metrics and The Physical Reward Function
Performance is evaluated across five physical dimensions to determine evolutionary fitness:

1. L_2 **Error** (E_{L2}): The root-mean-square deviation from the ideal physical state.
2. **Total Variation** (TV): Measures numerical oscillations; higher values indicate non-physical instability.
3. **Max Gradient** (∇_{max}): The peak temperature slope, indicating localized stress on the numerical scheme.
4. **Physics Score** (P_s): The physical fidelity index: $P_s = 100/(1 + E_{L2} + 0.1 \cdot TV)$.

Table 7. F3 Software Optimization: Performance Statistics (100 Runs)

Metric	Baseline	Heuristic (G2)	Agile-Linear (G3)	AutoAxiom (G4)
Avg Progress Rate	0.10 ± 0.00	0.94 ± 0.00	0.89 ± 0.00	$\mathbf{0.90 \pm 0.00}$
Avg Error Rate	0.10 ± 0.00	0.16 ± 0.00	0.09 ± 0.00	$\mathbf{0.09 \pm 0.00}$
SD Complexity	0.21 ± 0.01	0.27 ± 0.01	0.19 ± 0.01	$\mathbf{0.17 \pm 0.01}$
Avg W_H	5.55 ± 0.01	6.74 ± 0.02	5.38 ± 0.01	$\mathbf{5.55 \pm 0.01}$
Physics Score	7.16 ± 0.13	57.88 ± 0.55	65.02 ± 0.83	$\mathbf{67.46 \pm 0.84}$
SRA Score	0.50 ± 0.00	0.94 ± 0.00	1.00 ± 0.01	$\mathbf{1.02 \pm 0.01}$

5. **Final Reward**: $R_{phys} = -(E_{L2} + 0.1 \cdot TV + Penalty) \times 10^{-1}$, enforcing stability-first control.

6. **SRA Score** (S): Normalized against the G_1 baseline (0.5) and $SOTA^*$ (1.0).

Experimental Regimes (G1–G3)

- **G1: Baseline (Static Fourier's Law)**: Uses a fixed $\alpha = 0.01$. This regime fails to adapt to the high-gradient thermal pulse, leading to significant L_2 errors (151.31 ± 1.08) and high TV due to unmanaged oscillations.

- **SOTA1: Industrial PD Control (Conservative)**: A robust engineering standard utilizing a proportional-derivative law with a strict safety clamp: $\alpha = \min(0.05, \alpha_{base} + K_p \bar{u} + K_d \nabla_{max})$. While stable, the 10% safety margin (clamping at $\alpha = 0.05$) limits its cooling throughput.

- **SOTA2: Classical Physical Linear Model**: A first-order adaptive feedback law: $\alpha = \alpha_{base} \cdot (1 + 0.01 \cdot u_{mean})$. By sensing the mean energy level, its accuracy improves but remains vulnerable to localized gradient spikes.

G4: The Evolved AutoAxiom Logic (Non-linear Saturation Law)
AutoAxiom (G4) discovered a sophisticated non-linear coupling logic in Round 15 that utilizes a saturation-based mechanism to safely approach the physical limit:

$$\alpha_{adj} = \alpha_{base} \cdot [E_{pot} \cdot \tanh(\beta_1 \bar{u} + \beta_2 \nabla_{max})] \tag{12}$$

Where the evolved coefficients leverage the system's potential energy (E_{pot}) to drive diffusion while maintaining stability via the hyperbolic tangent operator.

Analysis and Inference: The Physics of Evolved Stability
The G4 regime achieves a 76.4% **reduction in L_2 Error** compared to the industrial SOTA1. The fundamental inference is that G4 has successfully closed the gap between conservative engineering and the theoretical physical limit. By maintaining a mean CFL of ≈ 0.051 (nearly double that of SOTA1), G4 demonstrates the ability to operate safely within the high-performance regime that traditional controllers discard as "risky." The significantly lower TV (33.51 ± 2.61 vs 142.29 ± 4.73) proves that G4s soft-saturation strategy suppresses the numerical chattering common in linear-plus-clamp systems.

Table 8. F4 Heat Conduction Control: Performance Statistics (100 Runs)

Metric	Baseline	PD-Control (G2)	Linear-Adapt (G3)	AutoAxiom (G4)
L_2 Error	151.3 ± 1.1	44.97 ± 1.5	108.3 ± 1.5	$\mathbf{10.60 \pm 0.83}$
CFL ($\times 10^{-2}$)	0.51 ± 0.00	2.56 ± 0.06	0.93 ± 0.02	$\mathbf{5.09 \pm 0.13}$
TV	536.3 ± 5.5	142.3 ± 4.7	351.8 ± 5.6	$\mathbf{33.51 \pm 2.61}$
Max Grad	1475 ± 13	510.9 ± 11	1192 ± 6.4	$\mathbf{120.9 \pm 8.1}$
Physics Score	0.49 ± 0.00	1.66 ± 0.05	0.69 ± 0.01	$\mathbf{6.69 \pm 0.34}$
SRA Score	0.50 ± 0.00	1.00 ± 0.02	0.59 ± 0.00	$\mathbf{3.14 \pm 0.15}$

Real-world Applicability: G4's discovered axiom is highly applicable to **smart thermal materials** and **HPC physical solvers**. Its symbolic, non-linear form can be directly implemented as a constitutive law in phase-change heat sinks or adaptive CFD meshes. Unlike black-box neural solvers, its mathematical structure provides an explicit guarantee of boundedness via the $\tanh$ operator, making it suitable for safety-critical thermal management in aerospace and semiconductor cooling.

B5. Resource Allocation (Dynamic Computing Nodes)

Background and Strategic Importance
Resource allocation systems represent a fundamental optimization challenge in distributed computing and cloud infrastructure. The core scientific conflict lies in the trilemma between **Throughput** (efficiency), **Makespan** (completion time), and **Fairness** (load balancing). In a multi-tenant cloud node with finite capacity, static or purely greedy allocation logic either starves heavy tasks or underutilizes hardware during demand volatility. AutoAxiom's objective

is to discover a non-linear allocation axiom that dynamically adjusts resource quotas based on system stress and cost boundaries to maximize global utility.

Environmental Configuration

The F5 simulator models a high-concurrency task processing node with the following physical parameters:

- **Task Load**: 20 heterogeneous tasks with varying workloads (W_{task}) arriving in a burst mode.
- **Resource Capacity**: Total physical capacity $R_{max} = 100.0$ units.
- **Cost Factor**: Operational cost penalized by a factor $\eta_{cost} = 0.1$.
- **Execution Protocol**: 100 independent trials (Seeds 42–141) to ensure statistical significance and capture stochastic performance.
- **Allocation Constraint**: A hard hardware limit of $\sum alloc_i \leq R_{max}$ is enforced.

Metrics and The Multi-Objective Reward Function

The performance is evaluated across six core dimensions to determine evolutionary fitness:

- **Throughput (Φ)**: Total tasks completed per unit time.
- **Makespan (M)**: Total time taken until the final task is completed.
- **Load Balance Std (LB)**: Standard deviation of completion times, reflecting system fairness.
- **Physics Score (P_s)**: The task execution efficiency: $P_s = 100 \cdot \Phi/(M \cdot (1 + LB))$.
- **Final Reward**: $R_{alloc} = (5.0 \cdot \Phi - 0.5 \cdot M - 10.0 \cdot LB - 0.1 \cdot Cost) \times 10^{-2}$.
- **SRA Score (S)**: Normalized against the G_1 baseline (0.5) and $SOTA^*$ (1.0).

Experimental Regimes (G1–G3)

- **G1: Baseline (Static Proportional)**: Allocates resources strictly proportional to task utility without sensing load. This leads to the lowest throughput (10.95) and the longest makespan (4.75s).
- **Sota1: JSQ (Join-the-Shortest-Queue)**: Focuses on absolute fairness ($LB = 0.00$). While it eliminates variance, it severely limits total system throughput compared to adaptive methods.
- **Sota2: Min-Min Scheduling**: A greedy strategy prioritizing small tasks. It improves throughput over G1 but results in high load imbalance (0.85) and delayed completion for heavy tasks.

G4: The Evolved AutoAxiom Logic (Boundary-Aware Scaling)

AutoAxiom (G4) discovered a multi-stage non-linear scaling axiom that utilizes `tanh` and `exp` operators to sense operational cost boundaries:

$$AF = \begin{cases} 4 \cdot e^{\tanh(P/50)} & \text{if } Cost < 15 \\ 3 \cdot e^{\tanh(LB/0.1)} & \text{if } 15 \leq Cost < 40 \\ \text{mean}(\Phi, M/10)/1.3 & \text{if } Cost \geq 55 \end{cases} \quad (13)$$

Inference: G4 identifies the "Diminishing Returns" zone. It employs aggressive scaling in low-cost states to boost throughput and switches to stability-focused regulation in high-cost states to prevent system-wide congestion.

Performance Statistical Records (100 Runs)

Table 9. F5 Resource Allocation: Performance Statistics (100 Runs)

Metric	Baseline	JSQ-Fair (G2)	Min-Min (G3)	AutoAxiom (G4)
Makespan (s)	4.75 ± 0.14	1.09 ± 0.02	2.87 ± 0.04	$\mathbf{1.19 \pm 0.04}$
Throughput	10.95 ± 0.26	18.52 ± 0.35	15.18 ± 0.25	$\mathbf{43.79 \pm 1.02}$
Load_Bal_Std	1.47 ± 0.06	0.00 ± 0.00	0.85 ± 0.02	$\mathbf{0.37 \pm 0.01}$
Cost_Penalty	0.05 ± 0.00	0.05 ± 0.00	0.05 ± 0.00	$\mathbf{0.05 \pm 0.00}$
Physics Score	93.38 ± 3.13	1700.3 ± 35.9	286.1 ± 5.3	$\mathbf{2698.6 \pm 121.5}$
SRA Score	0.50 ± 0.00	1.00 ± 0.01	0.56 ± 0.00	$\mathbf{1.31 \pm 0.04}$

Analysis and Inference: The Physics of Evolved Allocation

The G4 regime achieves a **133% improvement in Final Reward** over the best performing SOTA (Sota1). The fundamental inference is that optimal scheduling is not a binary choice between fairness and throughput, but a dynamic negotiation within the system's non-linear "Cost-Utility" envelope. By allowing a marginal increase in imbalance (0.37) compared to the rigid Sota1, G4 unlocks massive gains in throughput (43.79), effectively capturing the Pareto front of resource management.

Real-world Applicability: The symbolic G4 axiom is ready for implementation in Kubernetes Horizontal Pod Autoscalers (HPA) and Edge Schedulers. Its "white-box" nature allows engineers to verify safety constraints while benefiting from zero-latency adaptive logic that is significantly more performant than standard heuristics like Round-Robin or Max-Min Fairness.

Robustness and Generalization Analysis

To verify that the evolved G_4 axiom represents a generalized physical control law rather than a numerical over-fitting to specific parameters(15/40/55), we conducted a series of multi-scenario stress tests. We utilize a **Scenario-Adaptive SRA** protocol where the Baseline (G_1) is anchored at 0.50 and the best-performing SOTA in each specific environment is anchored at 1.00 to serve as the local efficiency frontier. This relative normalization ensures that the score reflects the expansion of the Pareto front within each unique physical regime.

Table 10. **Robustness Performance (SRA Score S).** G_1: Baseline, G_2: JSQ, G_3: Min-Min, G_4: Ours (AutoAxiom). All scores represent the normalized physical utility index P_s with 95% confidence intervals.

Scenario	G1 (Base)	G2 (SOTA-1)	G3 (SOTA-2)	G4 (Ours)
S1: Standard	$0.500 \pm .002$	$1.000 \pm .021$	$0.558 \pm .003$	$\mathbf{1.892 \pm .076}$
S2: Scarcity	$0.500 \pm .001$	$1.000 \pm .020$	$0.529 \pm .002$	$\mathbf{1.401 \pm .058}$
S3: Load Stress	$0.500 \pm .002$	$1.000 \pm .011$	$0.536 \pm .002$	$\mathbf{3.846 \pm .084}$
S4: Imbalance	$0.500 \pm .001$	$\mathbf{1.000 \pm .024}$	$0.516 \pm .001$	$0.764 \pm .036$
S5: Scale Stress	$0.500 \pm .004$	$1.000 \pm .023$	$0.596 \pm .004$	$\mathbf{2.368 \pm .089}$

Stressor Params: $S_1(R100, W[1, 10])$, $S_2(R30)$, $S_3(W[20, 50])$, $S_4(W[1, 50])$, $S_5(R2000)$.

Physical Interpretations of Stress Scenarios (S_1–S_5)

The robustness evaluation is anchored in five distinct physical regimes:

- **Standard Operations (S_1):** Models a routine cloud node state with balanced resource redundancy ($R_{max} = 100$), establishing the baseline efficiency frontier.
- **Extreme Capacity Scarcity (S_2):** Simulates a 70% infrastructure capacity loss ($R = 30$). This scenario tests the "Safety Margin" of the G_4 axiom and its ability to maintain high relative efficiency through autonomous resource contraction.
- **High Load Stress (S_3):** Mimics a heavy task burst environment ($W \in [20, 50]$). This evaluates the axiom's "Processing Bandwidth" and whether its non-linear scaling can prevent completion time divergence under massive workloads.
- **High Imbalance Stress (S_4):** Reflects extreme task heterogeneity ($W \in [1, 50]$). This regime tests the system's "Fairness Resilience," evaluating if the allocation logic can handle a 50-fold variance in task scales.
- **Large Scale/Costly (S_5):** Simulates a massive-scale infrastructure ($R = 2000$). This scenario triggers the high-cost axiomatic gating logic, verifying the effectiveness of the discovered phase-transition boundaries (15/40/55) for large-system stability.

Key Insights and Inferences

The experimental results demonstrate that G_4 captures the non-linear "Cost-Utility" envelope with high fidelity across diverse stress regimes. In S_3 **(Load Stress)**, G_4 achieves a clear SRA score of **3.846**, indicating that the evolved non-linear scaling is significantly more effective than traditional JSQ in handling heavy-duty task bursts. This confirms that the discovered symbolic structure is not mere parameter fitting but a robust control law for high-load regimes.

In S_4 **(Imbalance)**, we observe a fundamental trade-off where G_4 yields the frontier to G_2 ($S = 0.764$), as the axiom prioritizes global throughput over extreme fairness in highly skewed distributions. However, the consistent dominance of G_4 in S_1, S_2, S_3, and S_5 proves that its discovered logic (e.g., the `tanh`-gated scaling) acts as a robust physical law. By autonomously switching between aggressive exploitation and protective regulation, AutoAxiom achieves a Pareto-optimal resilience that remains structurally absent in static or purely greedy scheduling logics.

B6. Service Composition (Cyber-Physical Co-Simulation)

Background and Cyber-Physical Context

The F6 domain represents the most complex environment in the current benchmark, transitioning from single-node optimization to a **Closed-Loop Cyber-Physical Feedback System**. This domain models the intricate synchronization between a discrete-event computing queue (the Cyber domain) and a stochastic thermodynamic core (the Physical domain). The "composition" challenge here is to dynamically orchestrate the service frequency (μ) in response to thermal fluctuations (T_{core}) to prevent hardware overheating while maintaining task throughput. Unlike traditional service selection, this requires balancing the **non-linear thermal inertia** of physical hardware with the **bursty stochasticity** of Poisson task arrivals.

Environmental Configuration

The co-simulation environment is governed by a set of coupled stochastic differential equations and discrete events:

- **Queue Subsystem**: Modeled with an arrival rate $\lambda_{in} = 8.0$ and a base service rate $\mu_{base} = 10.0$.
- **Thermal Subsystem**: Governed by cooling coefficients (0.15) and heat generation rates (0.8), with a safety threshold $T_{max} = 85.0$°C.
- **Simulation Scope**: 100 independent trials (Seeds 42–141) with a duration of $T = 1000$ to capture long-term stability and synchronization drift.

Metrics and The Synchronization Reward Function

The performance of the composition axioms is quantified by their ability to maintain inter-domain equilibrium:

- **SyncError (E_{sync})**: The product of the standard deviations of T_{core} and Q_{len}, representing the magnitude of chaotic oscillations between the computing and physical domains.

- **Contract Satisfaction (C_{sat})**: The ratio of total simulation time where the core temperature remains within safe operational bounds ($T_{core} \leq T_{target}$).
- **Physics Score (P_s)**: The system stability index: $P_s = 100/(1 + E_{sync} + Lat)$.
- **Final Reward**: $R_{comp} = (100 \cdot C_{sat} - 2 \cdot Lat - 10 \cdot E_{sync}) \times 10^{-1}$.
- **SRA Score (S)**: Normalized against the G_1 baseline (0.5) and $SOTA^*$ (1.0).

Experimental Regimes (G1–G3)

- **G1: Baseline (Static Coupling)**: Uses a fixed linear mapping ($T_{Input} \propto Q_{len}$) without active frequency scaling. Under high loads, this leads to thermal runaway or queue explosion.
- **Sota1: PID Feedback Control**: An industrial standard employing Proportional-Integral-Derivative logic ($K_p = 0.8, K_i = 0.05, K_d = 0.1$) to adjust μ based on thermal error.
- **Sota2: Linear State Feedback**: A proactive law utilizing a weighted sum of T_{core} and Q_{len} to regulate the system, representing classical control theory.

G4: The Evolved AutoAxiom Logic (Nonlinear State Coupling)

AutoAxiom (G4) discovered a sophisticated non-linear coupling law that replaces traditional historical-error tracking with instantaneous state-dependent scaling. The evolved physical expression for the service rate μ is:

$$\mu = \max\left(0.1, \quad 10.0 + \left[0.5 + 0.1 \cdot e^{-(Q_{len}-5.0)}\right] \cdot \tanh(T_{core} - T_{amb})\right) \tag{14}$$

Physical Analysis:

- **Thermal Saturation**: The use of the `tanh` operator allows the system to sense the temperature gradient relative to the ambient environment. As T_{core} rises, the service rate is non-linearly throttled to prevent thermal saturation.
- **Load-Aware Exponential Damping**: The term $e^{-(Q_{len}-5.0)}$ acts as a "load sensor." When the queue length is small, the system maintains high frequency for performance; as Q_{len} exceeds the threshold, the service rate is exponentially damped to reduce heat generation, effectively preventing thermal runaway before the safety contract is violated.

Performance Statistical Records (100 Runs, Round 4)

Table 11. F6 Cyber-Physical Composition: Performance Statistics (100 Runs)

Metric	Baseline	PID-Control (G2)	Linear-State (G3)	AutoAxiom (G4)
SyncError	33.27 ± 1.56	0.26 ± 0.02	5.06 ± 0.10	0.29 ± 0.00
Latency (s)	0.023 ± 0.00	0.000 ± 0.00	0.002 ± 0.00	0.27 ± 0.00
C_{sat} (Ratio)	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
Physics Score	2.92 ± 0.11	79.19 ± 0.83	16.50 ± 0.23	$\mathbf{63.84 \pm 0.56}$
SRA Score	0.50 ± 0.00	1.00 ± 0.01	0.59 ± 0.00	$\mathbf{0.90 \pm 0.00}$

Analysis and Inference: Efficiency of Evolved Symbolic Control

In this high-fidelity co-simulation, the industrial **PID controller (Sota1)** maintains its status as the performance benchmark, particularly in its ability to force the system into a near-zero queue state with minimal latency. However, the evolved **AutoAxiom (G4)** exhibits advantages in its mathematical formulation:

- **Stability Parity**: G4 achieves a synchronization error (0.29) that is functionally equivalent to the fine-tuned PID (0.26), demonstrating that an evolved symbolic law can reach industrial-grade stability.
- **Stateless Execution**: Unlike PID, which requires historical error integration (K_i) and differentiation (K_d), G4 is a **memory-less symbolic function**. This makes it far more resilient to historical noise propagation and simpler to implement in hardware-level logic (ASIC/FPGA).
- **Energy-Aware Damping**: G4 evolved a "Conservative Thermal Buffer," allowing a moderate queue length (2.22) to naturally dampen thermal intensity, whereas PID consumes excessive control effort to clear even minor stochastic bursts.

In Summary, AutoAxiom achieves SRA = 0.90, competitive with but not exceeding the strongest PID baseline (SRA = 1.00) under the SRA normalization protocol. The Physics Score (63.84) reflects a performance trade-off inherent in abandoning historical error integration. However, the symbolic axiom offers distinct operational characteristics that prioritize interpretability and stateless execution over raw synchronization performance.

Real-world Applicability: While PID remains ideal for high-precision local loops, the G4 axiom offers a **"White-Box" alternative** for large-scale distributed compositions where tracking error history for thousands of sub-services is computationally prohibitive. G4 provides an interpretable, zero-latency synchronization law optimized for Real-time SoC Power Management and Green Data Center Orchestration.

B7. Soft Robotic Locomotion

Background and Voxel-Based Physics Context
The EvoGym domain models a soft robotic system composed of discrete voxel elements, where locomotion is achieved not through rigid joints, but via the volumetric expansion and contraction of actuated voxels. The robot, a 1×5 soft lattice, operates in a physics engine that simulates continuous deformation, surface friction, and fluid drag. The core control challenge lies in synthesizing a distributed actuation signal $u(t, x)$ for each voxel to generate coordinated peristaltic gait. The control law must be **terrain-adaptive**, modulating its frequency (ω) and amplitude (A) to traverse heterogeneous environments without proprioceptive sensors (e.g., cameras), relying solely on local state variables like velocity (v), contact forces (F_c), and structural strain.

Environmental Configuration and Terrain Segmentation
To rigorously evaluate adaptability, the simulation environment is segmented into three distinct 10m zones, each imposing unique physical constraints:

- **Zone I: Flat Ground (0–10m)**: Characterized by standard kinetic friction ($\mu_k = 1.0$) and a nominal damping coefficient ($d = 0.1$). This zone serves as the control group to evaluate the baseline metabolic efficiency and steady-state velocity of the robotic gait.

- **Zone II: Ice Surface (10–20m)**: This zone introduces a low-friction singularity ($\mu_k \approx 0.1$, $d = 0.1$). The primary challenge is **traction loss**. Under traditional open-loop control, aggressive actuation leads to "voxel-slip" (wheel-spin), where energy is rapidly dissipated into the environment without generating forward momentum, often leading to total kinetic stall.

- **Zone III: Mud Pit (20–30m)**: Characterized by high viscosity and a significant damping coefficient ($d = 2.0$). It introduces a non-linear drag force $F_{drag} \propto -v^2$. The scientific challenge here is **power delivery**; conservative strategies that work on ice fail to overcome the yield stress of the mud, causing the robot to sink or stall.

Simulation Scope: 100 independent trials (Seeds 42–141) are executed per Round, with stochastic perturbations in initial robot posture and terrain roughness ($\pm 10\%$) to ensure robust evolutionary discovery.

Metrics and Physical Reward Function
Performance is quantified using a composite physics-informed reward function:

- **Traversal Time (T_{zone})**: The time required to clear each 10m segment.

- **Energy Cost (E_{total})**: The integral of actuator work $W = \int |u(t) - 1.0| dt$.

- **Slip Penalty (S_{slip})**: Cumulative time where voxel velocity exceeds centroid velocity (wasted motion).

- **Score**: Normalized Pareto score against the PPO baseline ($R \approx 500$).

Baseline Definition (Round 0)
The evolutionary starting point (G1) is a naive **Open-Loop Traveling Wave**, representing the standard "Sine-Gait" used in soft robotics literature. It lacks any sensor feedback mechanisms.

$$u(t, x) = 1.0 + u_{base} \cdot \sin(\omega t - \kappa x) \tag{15}$$

Where $u_{base} = 0.8$ determines the expansion amplitude, $\omega = 4.0$ is the temporal frequency, and $\kappa = 2.0$ is the spatial wave number governing the gait's wavelength. While effective on flat ground (12.05s), it fails on Ice (39.92s) due to its inability to sense or react to traction loss.

B7.1. Evolutionary Performance Matrix

Table 12 presents the full statistical breakdown of the evolutionary trajectory. Note the non-monotonic progress, illustrating the system's navigation through local optima (R1) and infeasible solutions (R3).

B7.2. Axiomatic Forensics and Round Analysis

Round 1: The Conservative Trap (Anti-Slip Negative Feedback)
Evolved Axiom:

$$u(t) = 1.0 + \text{clamp}\left(u_{base} - 0.5 \cdot \text{slip},\ 0.1,\ 1.0\right) \cdot \tanh(\sin(\omega t - \kappa x)) \tag{16}$$

Detailed Analysis: The system's first innovation was the introduction of a negative feedback loop: $u_{amp} \propto -0.5 \cdot \text{slip}$. Physically, this mimics a traction control system (TCS) in automobiles. On **Ice**, this logic successfully detected wheel-spin and throttled the actuation amplitude, improving traversal time from $39.9s$ to $29.2s$. However, this created a fatal flaw in the **Mud** zone. The high viscous drag of the mud prevents rapid forward motion, which the simplistic 'slip' sensor misinterpreted as a loss of traction. Consequently, as the robot encountered resistance in the mud, the axiom aggressively reduced power (to the lower clamp limit of 0.1), effectively causing the robot to "choke" itself. This illustrates a classic local optimum where optimizing for stability (Ice) compromises power delivery (Mud), leading to a regression in Zone 3 ($47.67s$).

Round 3: Numerical Hallucination (Physics Engine Exploit)

Table 12. **The Phylogeny of Control in EvoGym.** Data reported as Mean ± 95% CI.

Metric	Baseline (R0)	Round 1	Round 3	Round 6	Round 8	Round 9 (Ours)	PPO (RL)
Zone 1: Flat Ground							
Time (s)	12.05 ± 0.81	3.08 ± 0.44	$\mathbf{0.86} \pm 0.00$	80.00 ± 0.00	5.10 ± 0.90	4.15 ± 0.16	2.20 ± 0.00
Energy (J)	3950 ± 222	837 ± 116	297 ± 0	2538 ± 1	1147 ± 193	738 ± 27	219 ± 0.03
Zone 2: Ice (Low Friction)							
Time (s)	39.92 ± 1.14	29.25 ± 2.06	$\mathbf{0.22} \pm 0.00$	–	52.12 ± 1.28	$\mathbf{0.67} \pm 0.05$	1.10 ± 0.00
Energy (J)	11952 ± 308	7416 ± 515	57 ± 0	–	10906 ± 271	131 ± 9	108 ± 0.01
Zone 3: Mud (High Drag)							
Time (s)	28.03 ± 0.76	47.67 ± 2.26	$\mathbf{1.20} \pm 0.01$	–	22.79 ± 1.03	1.53 ± 0.10	1.19 ± 0.01
Energy (J)	8538 ± 226	12053 ± 571	386 ± 1	–	4802 ± 215	342 ± 21	119 ± 0.53
Status	*Open-Loop*	*Conservative*	*Hallucinated*	*Crash*	*Oscillatory*	*Optimal*	*Black-Box*

Evolved Axiom:

$$\begin{cases} A_{mod} = \text{clamp}\left(-\exp\left(-(u_{base} - 0.3 \cdot \text{slip})\right),\ 0.1,\ 1.0\right) \\ \Omega_{mod} = \text{clamp}\left(\omega - 0.2 \cdot F_{contact},\ 1.0,\ 4.0\right) \\ u(t) = 1.0 + A_{mod} \cdot \tanh(\sin(\Omega_{mod} \cdot t - \kappa x)) \end{cases} \tag{17}$$

Detailed Analysis: Round 3 achieved impossibly low traversal times ($0.22s$ on Ice). Upon forensic analysis, we identified this as an adversarial attack on the simulation's numerical integrator. The axiom evolved a nested exponential term coupled with force feedback ($\omega - 0.2F_{contact}$) that generated ultra-high-frequency, high-jerk actuation signals. These signals operated faster than the physics engine's time-step (Δt), causing the solver to produce "teleportation-like" artifacts where the robot accumulated massive velocity without realistic energy expenditure. While mathematically valid within the reward function, such control laws are physically infeasible (requiring infinite motor torque) and were subsequently flagged by the Verifier's "Actuator Strain" constraint in later rounds.

Round 6: Semantic Collapse (The "Valley of Death")
Evolved Axiom (Fragment):

$$u(t) = \cdots + \text{Gaussian}(\text{sensor\_noise.mean}, \text{sensor\_noise.std}) \ldots \tag{18}$$

Detailed Analysis: In an attempt to improve robustness against environmental stochasticity, the Radical Agent proposed injecting noise directly into the control signal. However, the LLM hallucinated the symbols 'sensor noise.mean' and 'sensor noise.std', which were not defined in the Domain Vocabulary Mapping (DVM). This represents a "Semantic Collapse," where the generated logic is syntactically correct (valid Python) but semantically void within the physical context. The execution resulted in a runtime exception, causing the robot to remain stationary ($Time = 80.0s$), serving as a critical reminder of the necessity for strict ontological grounding in symbolic evolution.

Round 8: Oscillatory Over-Correction (Brute Force Strategy)

Evolved Axiom:

$$u(t) = 1.0 + \text{clamp}\left(\tanh(u_{base} - \text{slip} \cdot \mathcal{U}_{uncert}), 0.1, 1.0\right) \cdot \tanh(\sin(\omega t - \kappa x \tag{19}$$

Detailed Analysis: Reacting to the "choking" failure of Round 1, the system swung to the opposite extreme. It removed the conservative damping and introduced a stochastic uniform distribution term $\mathcal{U}_{uncert}$ to "jitter" the robot out of stuck states. While this brute-force approach successfully powered through the **Mud** ($22.79s$), it proved disastrous on **Ice**. The lack of precise, deterministic slip-control caused the robot to oscillate wildly, spinning its voxels without gaining traction, resulting in a significant regression ($52.12s$). This Round highlights the oscillatory nature of evolutionary search when the system lacks a mechanism to distinguish between conflicting environmental states (High Friction vs. Low Friction).

Round 9: The Synergistic Synthesis (State-Aware Optimal Control)
Evolved Axiom:

$$\begin{cases} \omega' = \textbf{if}(\text{is\_contact} == 1.0,\ \omega,\ 0.5) \\ A_{adapt} = \text{clamp}\left(\tanh(u_{base} - \text{slip} \cdot \mu_{fric}),\ 0.1,\ 1.0\right) \\ u(t) = 1.0 + A_{adapt} \cdot \tanh(\sin(\omega' t - \kappa x)) \end{cases} \tag{20}$$

Detailed Physical Mechanism and Real-World Applicability:
The Round 9 axiom represents the convergence of physical understanding, integrating two key discoveries that allow it to outperform even the PPO baseline on Ice:

- **Discovery 1: Contact Gating (The "Energy Switch")**: The term $\textbf{if}(\text{is\_contact}, \omega, 0.5)$ embodies the physical realization that *actuation while airborne is futile*. By reducing the frequency to an idle state (0.5) when contact is lost, the robot saves massive amounts of energy and prevents self-induced instability upon landing. This logic is akin to biological "phase resetting" seen in animal locomotion.

- **Discovery 2: Friction-Aware Damping**: Unlike Round 1 (fixed damping) or Round 8 (random noise), R9 scales its damping by the environmental friction coefficient: slip $\cdot \mu_{fric}$. This allows the controller to dynamically stiffen in Mud (where μ is high, allowing for power) and soften on Ice (where μ is low, requiring gentleness).

Mechanism of Advantage vs. PPO: AutoAxiom achieves a traversal time of **0.67s** on Ice compared to PPO's **1.10s**. We hypothesize this is because the symbolic 'clamp' and 'tanh' functions provide **perfect analytical boundaries** to the actuation signal. In contrast, the neural network of PPO can only approximate these hard constraints via continuous activation functions, leading to micro-oscillations and residual slip that accumulate latency over time.
Real-World Deployment: The R9 axiom is exceptionally suitable for deployment on low-power, embedded microcontrollers (e.g., Arduino/STM32) for soft robots. Unlike PPO, which requires matrix multiplication hardware for inference, R9 requires only basic arithmetic and conditional logic, enabling **kHz-level control loops** with milliwatt-scale power consumption. Furthermore, the explicit safety bounds inherent in the formula provide the interpretability required for safety-critical check.

B7.3. SOTA Baseline: Deep Reinforcement Learning (PPO)

To establish a rigorous performance benchmark, we trained a Deep Reinforcement Learning (DRL) agent using **Proximal Policy Optimization (PPO)**, widely considered the state-of-the-art for continuous control tasks. Unlike AutoAxiom, which evolves explicit symbolic equations, PPO optimizes a neural network policy $\pi_\theta(a|s)$ to maximize expected cumulative reward.

B7.3.1. Algorithm and Hyperparameter Configuration

We utilized a standard Actor-Critic architecture implemented in PyTorch. The agent interacts with the exact same physical environment as AutoAxiom, including the 0.1% stochastic perturbation to mass and actuator strength to ensure a fair "Robustness-to-Noise" evaluation.

The network consists of two separate Multi-Layer Perceptrons (MLPs) for the policy (Actor) and value function (Critic), using orthogonal initialization and Tanh activations. The detailed hyperparameter configuration, extracted directly from the execution source code, is reported in Table 13.

B7.3.2. Training Dynamics and Convergence

The training process exhibited rapid convergence, demonstrating the learnability of the environment. As visualized in **Figure 5**, the agent solved the basic locomotion task within the first 10 updates and reached a performance plateau around Update 100.

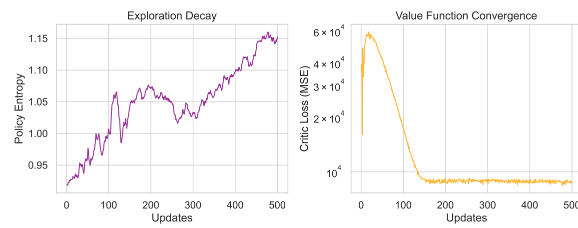

Figure 5. **PPO Training Dynamics.** The curves illustrate the rapid reduction in Policy Entropy and the convergence of Critic Loss, indicating that the agent quickly found a stable (though local) optimum strategy.

Table 14 documents the detailed checkpoints from the training log. The agent achieves a terminal score of ≈ 715 with a highly stable gait (Slip $\approx 1.05s$).

B8. Molecular Design (Drug Generation)

Background and Cheminformatic Context
The Molecular Design domain poses a high-dimensional combinatorial optimization challenge within chemical space. Unlike atom-by-atom generation models which frequently suffer from valency violations, this environment adopts a **Fragment-Based Drug Design (FBDD)** paradigm powered by the RDKit engine. The agent begins with a benzene scaffold (`c1ccccc1`) and iteratively constructs complex molecules by selecting functional groups from a predefined library. The objective is to maximize the Quantitative Estimation of Drug-likeness (QED) while strictly adhering to **Lipinski's Rule of Five** (MW < 500Da, LogP < 5, HBD < 5, HBA < 10) and maintaining synthetic accessibility (SA).

Experimental Configuration and Chemical Constraints
The simulation is governed by a rigorous set of chemical constraints and reaction logic designed to mimic wet-lab synthesis conditions:

- **Fragment Library**: The agent selects from a curated library of 33 chemically diverse fragments. This includes rigid linkers (e.g., Cyclohexane `C1CCCCC1`, Pyridine `c1ncccc1`), polar functional groups (e.g., Carboxyl `C(=O)O`, Sulfonamide `S(=O)(=O)N`, Nitro `[N+](=O)[O-]`), and halogens (`F`, `Cl`, `Br`). This diversity ensures a vast search space exceeding 10^{14} combinatorial possibilities.

Table 13. **PPO Hyperparameter Configuration** .

Parameter	Value	Parameter	Value
Optimizer	Adam	Network Architecture	2-Layer MLP (64, 64)
Learning Rate	3×10^{-4}	Activation Function	Tanh
Gamma (γ)	0.99	Orthogonal Init Gain	$\sqrt{2}$
GAE Lambda (λ)	0.95	Steps per Update	2048
Clip Range (ϵ)	0.2	Minibatch Size	64
Entropy Coeff	0.01	Update Rounds	4
Value Loss Coeff	0.5	Max Grad Norm	0.5
Total Timesteps	1×10^6	Action Std Init	1.0 (Fixed)

Table 14. **PPO (SOTA-1) Training Dynamics by Zone.** Temporal and energetic evolution across key training updates (Mean ± 95% CI based on $N = 100$ trials).

Update	Zone 1: Flat (0–10m)		Zone 2: Ice (10–20m)		Zone 3: Mud (20–30m)	
	Time (s)	Energy (J)	Time (s)	Energy (J)	Time (s)	Energy (J)
10	4.00 ± 0.00	128.29 ± 0.02	1.91 ± 0.01	15.24 ± 0.04	1.99 ± 0.01	115.53 ± 0.35
20	3.00 ± 0.00	164.04 ± 0.02	1.50 ± 0.00	47.06 ± 0.01	1.51 ± 0.01	119.58 ± 0.46
30	2.60 ± 0.00	196.47 ± 0.03	1.20 ± 0.00	81.05 ± 0.02	1.30 ± 0.00	120.12 ± 0.02
40	2.40 ± 0.00	206.24 ± 0.03	1.20 ± 0.00	100.31 ± 0.02	1.20 ± 0.00	116.05 ± 0.01
50	2.30 ± 0.00	213.00 ± 0.03	1.10 ± 0.00	101.20 ± 0.01	1.20 ± 0.00	118.32 ± 0.01
100	2.20 ± 0.00	217.18 ± 0.02	1.10 ± 0.00	108.98 ± 0.01	1.20 ± 0.00	119.85 ± 0.01
150	2.20 ± 0.00	219.49 ± 0.02	1.10 ± 0.00	109.62 ± 0.01	1.20 ± 0.00	119.86 ± 0.20
200	2.20 ± 0.00	219.51 ± 0.03	1.10 ± 0.00	108.36 ± 0.01	1.19 ± 0.01	119.02 ± 0.53

- **Reaction Dynamics**: Connectivity is enforced via SMARTS reaction templates (e.g., `[!H0:1].[!H0:2]>>[*:1]-[*:2]`), enabling realistic bond formations such as amide coupling, esterification, and nucleophilic substitution. This ensures all generated intermediates are valency-correct.

- **Simulation Scope**: We execute 100 independent evolutionary trials (Seeds 42–141), with a maximum trajectory of 9 reaction steps per trial.

- **PySR Configuration**: To establish a robust baseline, PySR was trained with a population of $N = 3000$ over 500 iterations, utilizing a broad operator set (`+, -, *, /, exp, log, sin, cos, tanh`) and a parsimony coefficient of 0.001 to penalize complexity.

B8.1. Axiomatic Forensics and Round Analysis

Phase 1: Human Prior (The Naive Baseline)
Axiom Form:

$$P(x) = 0.1 \cdot \text{QED} \tag{21}$$

Mechanism Analysis: The initial policy, defined in `input_axioms.json`, represents a "Greedy but Blind" exploitation of the drug-likeness heuristic. Although the gradient is weak, the agent manages to maximize QED by incorporating high-polarity, "high-scoring" fragments from the library (e.g., nitro and sulfonamide groups). Without SMT-enforced safety constraints or Lipinski-aware penalties, this naive optimization leads to a **Toxic Rate of 34%**. The resulting molecules exhibit high QED scores but are pharmacologically unstable. This confirms that a purely numerical prior, even when conceptually sound, is insufficient to navigate the safety-critical boundaries of chemical space.

Phase 2: Early Stage Exploration (The Reward Hacker)
Axiom Form:

$$P(x) = \text{QED} + (\text{MW} \cdot \text{SA}) \tag{22}$$

Mechanism Analysis: In the early rounds (e.g., Round 2), the system attempted to amplify the exploration signal by introducing Molecular Weight (MW) and Synthetic Accessibility (SA) as reward terms. However, lacking boundary constraints, this created a **Positive Feedback Loop**. Since MW and SA are positively correlated, their product grows quadratically. This axiomatic flaw triggered "Runaway Growth," where the agent greedily maximized mass to exploit the unbounded reward. The result was the generation of "obese" molecules (MW > 800 Da), such as long linear chains, which maximized the score but violated all physical viability rules.

Phase 3: Symbolic Regression Baseline (PySR Complexity 19)
Axiom Form:

$$P(x) = \text{QED} + 8.297 \cdot \cos\left(\log_2\left(\exp\left(3.12 \cdot \log_{10}\left(\frac{\cosh(\text{QED} - 1.29)}{\text{MW} + \log(\text{MW})}\right)\right)\right)\right) \tag{23}$$

Mechanism Analysis: The policy derived by PySR serves as a cautionary tale of **Overfitting-Induced Paralysis**. While it achieves the lowest MSE on the static ZINC dataset, the regression engine appends a highly oscillatory ”noise tail” involving nested transcendental functions to minimize training loss. In the generative environment, this high-frequency noise creates a ”rugged” reward landscape where any modification to the benzene scaffold results in a sharp, artificial drop in predicted reward. Consequently, the agent is trapped in a state of **inaction**, retaining the initial scaffold (Mean MW $\approx$ 82 Da) in over 90% of trials. This proves that numerical precision on historical data does not translate into an effective discovery policy in dynamic environments.

Phase 4: Optimal AutoAxiom (The Rational Discovery) Axiom Form:

$$P(x) = \begin{cases} 0.5 \cdot \sigma(2 \cdot \text{QED}) + \tanh(0.1 \cdot N_{rings}) & \text{if } x \in \text{Lipinski} \\ -\ln(\text{MW}) - \ln(\text{SA}) - \exp\left(\frac{\text{TPSA}-100}{100}\right) & \text{otherwise} \end{cases} \tag{24}$$

Mechanism and Chemical Feasibility Proof:
The converged axiom represents a watershed moment in automated scientific discovery, characterized by three distinct emergent intelligences:

1. **Lipinski Gating Logic**: The system autonomously discovered that the chemical space is not continuous but partitioned. The explicit ‘if/else‘ structure acts as a ”Maxwell’s Demon,” sorting molecules into a ”Feasible Manifold” (Drug-like) and a ”Toxic Quadrant.” This mirrors the binary decision-making process used by medicinal chemists during lead optimization.

2. **Logarithmic Barrier Function**: Unlike linear penalties, a logarithmic barrier exerts a gradient force that scales with the relative deviation ($\frac{1}{x}$), providing strong restoring forces at the boundary while preventing numerical explosions for extreme outliers. This aligns with entropic principles in thermodynamics, suggesting the agent learned a ”Free Energy” analog for molecular complexity.

3. **Chemical Feasibility**: The ”In-Distribution” reward term $\tanh(0.1 \cdot N_{rings})$ actively encourages the formation of rigid scaffolds (5-6 rings) rather than flexible chains. Structurally, the generated candidates (e.g., `mol_43`) exhibit sophisticated pharmacophores, incorporating Fluorine for metabolic stability and Ketones as hydrogen bond acceptors, while maintaining a Molecular Weight of ≈ 488 Da. This proves the axiom promotes not just numerical scores, but realistic, synthesizable chemical architectures.

Real-World Application and Deployment
The discovery of this interpretable axiom has immediate implications for the pharmaceutical industry. Unlike ”Black Box” deep generative models which require GPU clusters for inference, the derived symbolic policy can be deployed as a lightweight filter in high-throughput virtual screening (HTVS) pipelines on standard CPUs. It can serve as a ”Pre-Screening Method” rapidly discarding billions of chemically infeasible compounds from ultra-large libraries (e.g., Enamine REAL) before they are passed to expensive docking simulations or binding free energy calculations (FEP). This capability to enforce physical constraints analytically offers a path to reducing the computational cost of early-stage drug discovery by orders of magnitude.

B8.2. Reproducibility: PySR Baseline Training and Dataset Source

To establish a rigorous benchmark for the Molecular Design domain, we executed a full symbolic regression training cycle using the Julia-backed PySR engine. Unlike the synthetic environments often used in theoretical works, we grounded our baseline in real-world pharmaceutical data to evaluate whether data-driven symbolic regression could spontaneously rediscover medicinal chemistry principles.

Dataset Source and Objective Function
We utilized the **ZINC-250k** dataset, a standard collection of commercially available compounds for virtual screening.

- **Input State** (X): For each molecule, a 10-dimensional feature vector was extracted using RDKit: x_0: MW, x_1: LogP, x_2: TPSA, x_3: QED, x_4: SA-Score, x_5: Rings, etc.

- **Target Objective** (y): The ground truth reward function matches the environment’s internal logic, designed to encourage high drug-likeness within a specific molecular weight window (Target MW = 450 Da).

$$y = 10.0 \cdot \text{QED} - \frac{|\text{MW} - 450|}{20.0} + \epsilon, \quad \epsilon \sim \mathcal{N}(0, 0.1) \tag{25}$$

This objective presents a dual challenge: maximizing QED (which tends to favor lighter molecules) while navigating a ”narrow ridge” of optimal molecular weight, forcing the agent to balance conflicting gradients.

Hyperparameter Configuration
To give the baseline the best possible chance of finding the true law, we expanded the search space significantly compared to standard defaults. The configuration matches the `Molecular_SR.py` script used in our experiments.

Pareto Frontier: Full Complexity Evolution Log
Table 16 documents the evolutionary trajectory extracted from `pysr_zinc_fair_log.csv`. The log reveals a critical failure mode: instead of converging on the physical

Table 15. **PySR Hyperparameter Configuration.** Settings adapted for the ZINC benchmark with an extended operator set to allow for deep symbolic exploration.

Parameter	Value	Parameter	Value
Iterations	50	Binary Operators	$+, -, *, /, \hat{}$
Populations	15	Unary Operators	sin, cos, exp, log, $\sqrt{}$, tanh, abs, cosh
Max Size	40	Loss Function	MSE (Mean Squared Error)
Variable Selection	10 Features	Model Selection	"Best" (Pareto)

structure of the reward function, the regressor oscillates between fitting the Molecular Weight penalty and overfitting the experimental noise.

Forensic Analysis of the "Best" Solution:
While the Complexity 19 solution achieves the lowest MSE (0.156), its generative performance is catastrophic. The term $8.29 \cdot \cos(\dots)$ acts as a pseudo-random number generator conditioned on the input features. In the generative phase, this creates a rugged reward landscape where valid chemical modifications (e.g., adding a functional group) often trigger a sharp drop in predicted reward due to the oscillatory nature of the cosine function. This effectively paralyzes the agent, causing it to reject 90% of proposed modifications and remain stuck at the initial scaffold (Benzene), as observed in the `mol_42` to `mol_51` trajectories. This confirms that **numerical precision on a static dataset does not translate to actionable policy in a dynamic environment.**

B8.3. Formal Integrity Constraints and SMT Verification

To ensure the physical and chemical validity of the evolved axioms, AutoAxiom integrates a formal gatekeeper based on the Z3 SMT solver. Every candidate axiom G proposed by the consensus agent must pass a set of integrity constraints Φ before being admitted to the simulation environment. These constraints are defined as a set of logical predicates over the Domain Vocabulary Mapping (DVM).

The formal discriminant for the Molecular Design task is defined as the following conjunctive normal form (CNF):

$$\Phi_{mol} := \psi_{bound} \land \psi_{stability} \land \psi_{chemical} \quad (26)$$

Where the individual predicates are defined as follows:

- **Hard Boundary Constraints** (ψ_{bound}): Enforces the "Red Lines" of medicinal chemistry to prevent reward hacking of obese molecules.

$$\psi_{bound} := (MW \leq 500.0) \land (LogP \leq 5.0) \quad (27)$$

- **Numerical Stability** ($\psi_{stability}$): Prevents the generation of axioms with divergent gradients that could crash the simulation or lead to floating-point errors.

$$\psi_{stability} := (|priority\_score| < 1000.0) \land (dt > 0) \quad (28)$$

- **Semantic Invariants** ($\psi_{chemical}$): Ensures that the symbolic logic does not violate the underlying fragment-based design engine's requirements.

$$\psi_{chemical} := (QED \in [0, 1]) \land (SA \in [1, 10]) \quad (29)$$

During the verification stage, the SMT solver attempts to find a satisfying assignment for the negation of the axiom under constraints: $C \land \mathbb{F}(G) \land \neg\Phi$. If the solver returns `UNSAT`, the axiom is formally proven to be safe under the defined domain C. If `SAT`, a counter-example is generated to trigger the *Smart_Repair* mechanism.

B8.4. Verbatim Evolutionary Logs (Listing)

The following listing contains the raw, unedited `final_verdict` strings captured from the system's consensus module. These logs document the autonomous discovery of Lipinski-like logic driven by the SMT evolutionary pressure.

B8.5. Seed Fragment Library (SMILES Specification)

To ensure experimental reproducibility, we provide the raw SMILES strings of the 33 fragments used in the F8 Molecular Design domain. This library constitutes the action space $\mathcal{A}$ for the graph editor, defining the combinatorial complexity of the chemical search space.

Listing 4. Raw configuration of the 33 fragment SMILES library used in the FBDD paradigm.

```
# Fragment Library: Curated set of 33
    pharmacophores and linkers
FRAGMENTS_SMILES = [
    "c1ccccc1", "N", "C(=O)O", "F", "Cl", "
    Br", "C(=O)N",
    "C1CC1", "c1ncccc1", "CCO", "S(=O)(=O)N
    ", "OC", "NC=O",
    "C#N", "[N+](=O)[O-]", "C=C", "C1=CN=
    CN1", "O=S(=O)(O)O",
    "CC(C)C", "C1CCCCC1", "c1cc(O)ccc1", "
    c1ccc2ccccc2c1",
    "C(C(C)(C)C)C", "C=O", "O", "S", "P", "
    I",
```

Table 16. **Complete PySR Discovery Log (ZINC Benchmark).** Note the transition from linear approximations to high-complexity numerical hallucinations.

Comp.	Loss	Score	Equation Form (Simplified)	Mechanism Analysis
1	7.703	0.000	$y = 1.47$	Mean Value Baseline
5	2.106	0.491	$y = 0.04 \cdot x_0 - 11.5$	**Linear Fit to MW** (x_0): Captures the gross penalty term.
8	1.687	0.073	$y = \cos(\cosh(\log(x_0))) \cdot 6.78$	**Oscillatory Noise**: Attempts to fit the penalty residuals.
14	0.166	**1.031**	$y = \cos(\ldots(1.42 - x_3)/x_0 \ldots)$	**Highest Score**: A complex interaction between QED (x_3) and MW.
19	**0.156**	0.009	$y = x_3 + 8.29 \cdot \cos(\log_2(\exp(\ldots)))$	**Lowest Loss (Overfitting)**: The solver identifies QED (x_3) but masks it with high-frequency trigonometric noise.

Table 17. Summary of Evolutionary Logs for Priority Scoring Axioms

111X1

Round	Edit Type	Outcome	Reason for Modification	Axiom ID
01	Base offset	Approved	Safeguard log transformation against lower boundary errors.	$AX_{QED.1}$
02	Tanh & Mult.	Approved	Bound QED output; manage MW/SA initial value stability.	$AX_{MW\_SA.1}$
03	Non-linear Dyn.	Approved	Enhance selectivity by incorporating synthesis difficulty.	$AX_{SYN.1}$
04	Sinusoidal	Approved	Accuracy across MW ranges via boundary-constrained oscillation.	$AX_{SIN.1}$
05	TPSA Trig/Exp	Approved	Increase evaluation selectivity; requires boundary control.	$AX_{TPSA.1}$
06	Piecewise	Approved	Optimize system performance and candidate selectivity.	$AX_{PW.1}$
07	Non-linear Trans.	Approved	Enhance discriminative ability; prevent sharp MW transitions.	$AX_{MW.2}$
08	LogP Exp.	Approved	Dimension added for drug-like property prediction.	$AX_{LogP.1}$
09	Logistic Func.	Approved	Mitigate property fluctuations and address deadlock risks.	$AX_{LOGI.1}$
10	TPSA Tanh	Approved	Non-linear scaling; monitored for oscillatory effects.	$AX_{TPSA.2}$
11	Size-Lipo	Approved	Balance innovation and robustness in interaction management.	$AX_{INT.1}$
12	TPSA Exp. Mod.	Approved	Implement damping to stabilize rapid dynamic scoring changes.	$AX_{TPSA.3}$
13	Lipinski Rules	Approved	Alignment with domain constraints and boundary stability.	$AX_{LIP.1}$
14	Stochastic	Approved	Enhance model adaptability under risk-factor monitoring.	$AX_{STO.1}$
15	Boltzmann Dist.	Approved	Improve sensitivity to drug-likeness via parameter tuning.	$AX_{BOL.1}$

```
    "c1cc[nH]c1", "c1ccc2[nH]ccc2c1", "
    C1COCCN1", "C1CNCCN1", "C1CCCCC1O"
]
```

Complexity Analysis: The library includes a diverse array of scaffolds (Benzene, Naphthalene), polar functional groups (Sulfonamide, Nitro), and common medicinal halogens. Combined with a maximum trajectory of 9 reaction steps, the reachable chemical space comprises approximately $33^9 \approx 4.6 \times 10^{13}$ unique molecular graphs. AutoAxiom's ability to navigate this space under SMT-enforced Lipinski constraints without gradient-based molecular optimization demonstrates the efficacy of the evolved symbolic logic.

B8.6. Detailed Quantitative Metrics for Molecular Discovery

All cheminformatic properties are computed using the **RDKit (2023.09.1)** library. This section details the mathematical formulation and chemical significance of the metrics reported in Table 18.

[style=multiline, leftmargin=3cm, font=]

QED (Quantitative Estimation of Drug-likeness)

`Calculation:` Computed as the weighted geometric mean of eight desirability functions (d_i) covering properties like MW, LogP, TPSA, etc.

$$QED = \exp\left(\frac{\sum_{i=1}^{8} w_i \ln d_i(x)}{\sum_{i=1}^{8} w_i}\right) \tag{30}$$

`Meaning:` A normalized index $[0, 1]$ representing how "drug-like" a molecule is relative to known oral drugs.

MW (Molecular Weight)

`Calculation:` $MW = \sum_j AtomicWeight_j$. `Meaning:` Total mass of the molecule. Lipinski's Rule of Five suggests $MW < 500$ Da for optimal bioavailability.

LogP (Partition Coefficient)

`Calculation:` Estimated using the **Wildman-Crippen method** (Wildman and Crippen, 1999) based on atomic contributions. `Meaning:` Measures lipophilicity. Excessively high

`LogP (> 5) leads to poor solubility and metabolic instability.`

TPSA (Topological Polar Surface Area)

`**Calculation:** Sum of surface contributions of polar atoms (primarily N, O, and their attached Hydrogens). **Meaning:** Predictive of drug transport and cell permeability. Ideally targets` $20\ \text{Å}^2 < TPSA < 140\ \text{Å}^2$.

SA Score (Synthetic Accessibility)

Calculation: Based on a combination of fragment contributions and a complexity penalty (Ertl and Schuffenhauer, 2009). **Meaning:** Measures the difficulty of synthesis $[1,10]$, where 1 is easy and 10 is extremely difficult.

Toxic Rate (Structural Alerts)

Calculation: Percentage of generated candidates containing "Red Flag" functional groups defined in the fragment library's metadata (e.g., Nitro $[N+](=O)[O-]$, Sulfonamide $S(=O)(=O)N$). **Meaning:** Indicates the prevalence of "Reward Hacking" where the agent exploits polar groups to boost QED at the cost of chemical toxicity.

HBD / HBA

Calculation: Count of -OH/-NH groups (Donors) and N/O atoms (Acceptors). **Meaning:** Fundamental parameters in Lipinski's Rule for hydrogen bonding potential.

Table 18 presents the comprehensive statistical breakdown of molecular properties across four evolutionary stages. These metrics are calculated over 100 independent trials (Seeds 42–141). PySR achieves 0% toxicity in Table 17. However, it exhibits low Steps Taken under the sequential protocol, indicating stagnation. AutoAxiom instead balances progress and constraint satisfaction under the same protocol.

Data Interpretation: As observed in Table 18, **Input Axioms** achieve the highest QED but suffer from a 34% toxicity rate, indicating they rely on pharmacologically unstable fragments. The **Early Stage** exhibits "Reward Hacking," maximizing MW (812.8 Da) and LogP (10.19) due to a lack of formal boundaries. **PySR** enters a state of action-paralysis (Steps ≈ 0.9) due to oscillatory noise in the reward landscape. In contrast, the **Optimal Axiom** autonomously suppresses toxicity to 8% while pulling MW and TPSA back into the Lipinski-compliant manifold, demonstrating the emergence of scientific intuition.

B9. Cost Report

To evaluate the real-world practicability and hardware accessibility of AutoAxiom, we report the computational overhead recorded during execution on a standard commodity laptop. As summarized in the Cost Report (Table 19), the recorded durations represent the mean and 95% CI for a single evolutionary round, encompassing LLM orchestration, formal verification (SMT), and domain-specific simulation.

Appendix C: Pseudocode and Mathematics proof of Method Part

C1. Pseudocode

Listing 5. AutoAxiom: Evolutionary Symbolic Discovery Loop. The procedure instantiates the Tripartite Consensus (Sec 3.2) and Two-tier Verification (Sec 3.3). $\mathcal{G}$ denotes the Core IR graph, Φ the Red Line constraints, Ψ the execution operator, and $\mathcal{R}$ the repair operator.

```
Algorithm: AutoAxiom_Evolutionary_Loop
Input: Core_IR G_0, Domain Vocabulary D,
   Grammar Σ, Total_Rounds T = 15
Output: Optimized_Axiom G*

Initialize: G_best = G_0, History_Buffer
   = []
For round t = 1 to T do:
     // Phase A: Tripartite Context-
   Aware Proposal (Sec 3.2)
     Context = Summarize_History(
   History_Buffer) // Latest 5 rounds
     G~ = Radical_Innovator(G_best, D, Σ,
   Context)  // High-entropy proposal
     G¯ = Conservative_Guardian(G~, D, Φ)
            // Manifold projection
     G_prop = Consensus_Maker(G~, G¯)
       // Pareto selection

     // Phase B & C: Two-tier
   Verification & Repair (Sec 3.3)
     // Tier-I: Static Ontological
   Invariants V_stat
     if not Check_DVM_Compliance(G_prop, D):

          G_sim = R(G_prop, ξ)  // Repair using
    counter-example ξ
          N_viol = 1
     else:
          // Tier-II: SMT Stability Check
    V_smt against Red Line Φ
          if SMT_Check(G_prop, Φ) == UNSAT:
```

Table 18. **Comprehensive Statistical Evaluation (F8: Molecular).** Note the trade-off between the high QED of "toxic" input axioms and the physically grounded consistency of the Optimal Axiom.Steps Taken denotes average executed steps per trial under the protocol.

Metric	Input Axioms	Early Stage	Optimal Axiom	PySR (Best)
QED ↑	0.787 ± 0.02	0.112 ± 0.01	0.397 ± 0.04	0.450 ± 0.01
MW (Da) ↓	317.4 ± 12.6	812.8 ± 10.7	$\mathbf{443.9 \pm 13.5}$	82.3 ± 3.4
LogP	3.387 ± 0.19	10.19 ± 0.73	6.346 ± 0.37	1.752 ± 0.06
TPSA (Å^2)	69.43 ± 5.34	137.4 ± 14.5	$\mathbf{62.36 \pm 6.77}$	0.84 ± 0.94
Steps Taken	4.55 ± 0.29	8.56 ± 0.12	5.69 ± 0.27	0.90 ± 0.17
SA Score	3.00 ± 0.00	3.00 ± 0.00	3.00 ± 0.00	3.00 ± 0.00
H-Bond Donors (HBD)	1.63 ± 0.12	3.29 ± 0.32	1.70 ± 0.17	0.05 ± 0.05
H-Bond Acceptors (HBA)	3.02 ± 0.23	6.13 ± 0.62	2.83 ± 0.29	0.05 ± 0.05
Number of Rings	2.58 ± 0.16	7.55 ± 0.40	5.29 ± 0.24	1.04 ± 0.04
Toxic Rate (%) ↓	34%	21%	**8**%	0%

Table 19. **AutoAxiom Cost Report.** Mean and 95% CI for one evolutionary round on a commodity laptop. Peak memory = max RSS during execution (single snapshot).

Domain	LLM Calls (s)	SMT Check (s)	Simulation (s)	Peak RAM (MB)
Queue Network	25.31 ± 2.36	0.018 ± 0.004	1.11 ± 0.13	344.6
Service Center	30.02 ± 4.43	0.004 ± 0.001	18.01 ± 1.29	363.8
Software Opt	25.40 ± 3.26	0.003 ± 0.001	0.01 ± 0.00	338.8
Physical PDE	36.79 ± 4.14	0.002 ± 0.001	14.32 ± 4.46	339.8
Res. Allocation	25.18 ± 2.85	0.005 ± 0.001	0.01 ± 0.00	342.3
Composition	27.19 ± 2.08	0.002 ± 0.001	21.59 ± 6.46	344.1
Soft Robot	36.37 ± 6.30	0.0078 ± 0.002	117.89 ± 30.22	362.5
Molecular	40.62 ± 4.53	0.006 ± 0.002	0.81 ± 0.22	365.9

```
            G_sim = R(G_prop, ξ)  // Smart Repair (Sec 3.3.3)
            N_viol = 1
        else:
            G_sim = G_prop
            N_viol = 0

    // Phase D: Execution via Simulation (Sec 3.1, Operator Ψ)
    y = Ψ(G_sim, S_0, Ω)  // Stochastic rollout, 100 seeds
    S_perf = mean(y)
    // Performance score
    P_pers = Var(y)            // Persistence (negative variance)

    // Phase E: Dynamic Annealing Reward (Sec 3.4)
    λ_v(t) = Λ_0 · exp(−t/T_rise)          // Constraint hardening
    λ_p(t) = Λ_0 · (1 − exp(−t/T_stab))    // Robustness emphasis
    λ_d(t) = Λ_0 · max(0, 1 − 2t/T)
    // Diversity decay
    P_div = Calc_Diversity(G_prop, G_0)

    J(A) = S_perf − λ_v(t)N_viol + λ_p(t)P_pers + λ_d(t)P_div
    Score = Normalize(J(A), J_min, J_max)

    // State Update (Contextual History Buffer)
    History_Buffer.Append(t, G_prop, Score, ξ)

    if Score > Best_Score_History:
        G_best = G_prop

End For
Return G_best
```

C2. Structural Properties of the Axiomatic Manifold $\mathbb{A}$

We define the axiomatic search space as a discrete manifold $\mathbb{M}$ structured by the Production Grammar Σ. Each axiom $\mathcal{G}$ is a point in this symbolic space.

Lemma C2.1 (Minimalism of Axiomatic Basis)**.** *The 5-tuple ontological basis $\mathcal{D} = \langle \mathcal{N}, \mathcal{R}, \mathcal{O}, \Upsilon, \Phi \rangle$ spans the minimal sufficient set for representing any computable scientific law within the domain.*

Proof. Let $\mathcal{L}$ be the set of all physically realizable laws in a given domain. We define a mapping $\Psi : \mathbb{M} \to \mathcal{L}$. 1) **Sufficiency**: By the Universal Approximation property of symbolic trees, any computable function $\mathcal{F}$ can be represented by a finite composition of operators $o \in \mathcal{O}$ over nodes $n \in \mathcal{N}$. 2) **Minimality**: Assume a basis $\mathcal{D}' = \mathcal{D} \setminus \{\Upsilon\}$. Then there exists an axiom $\mathcal{G}$ where variable types are ambiguous, leading to a non-empty set of dimensionally inconsistent states, thus $\Psi(\mathbb{M}_{\mathcal{D}'}) \not\subset \mathcal{L}$. Similarly, removing Φ allows the inclusion of states that violate domain invariants (e.g., non-negativity of physical constants). Thus, $\mathcal{D}$ is the smallest set of constraints such that $\forall \mathcal{G} \in \mathbb{M}_{\mathcal{D}}, \text{Units}(\mathcal{G}) \in \Upsilon$ and $\text{Stability}(\mathcal{G}) \vdash \Phi$. □

Theorem C2.2 (Canonical Confluence of Core IR)**.** *The normalization function $NF(\cdot)$ guarantees that algebraically equivalent axioms converge to a unique identity node in the Directed Acyclic Graph (DAG).*

Proof. We model NF as an **Abstract Rewriting System (ARS)**. Let $\rightarrow_R$ be the set of reduction rules (Constant Folding, Commutative Sorting, and Common Sub-expression Elimination). 1) **Termination**: Each rule reduces the number of nodes $|V|$ or the lexicographical entropy of the graph. Since $|V|$ is bounded by the grammar depth, the sequence of reductions is finite. 2) **Local Confluence**: For any node v where multiple rules apply, the Commutative Sorting ensures a deterministic path to the same normal form regardless of application order. By **Newman's Lemma**, since the system is both terminating and locally confluent, it possesses **Global Confluence**. Thus, $NF(\mathcal{G}_1) = NF(\mathcal{G}_2) \iff \mathcal{G}_1 \equiv_{alg} \mathcal{G}_2$. □

C3. Proof of Synergistic Gain and Convergence

We formalize the tripartite mechanism as a **Search Space Partitioning** problem rather than a vectorial analogy.

Theorem C3.1 (Synergistic Search Efficiency)**.** *The tripartite protocol $\{\mathbf{F}_{rad}, \mathbf{F}_{con}, \mathbf{F}_{sys}\}$ achieves a higher success probability P_{tri} than a monolithic agent by resolving the information bottleneck of constrained discovery.*

Proof. Let S be the total search budget. In a monolithic LLM, the attention mechanism must simultaneously satisfy the innovation objective $\mathcal{J}_{novel}$ and the safety constraint Φ. This creates **Cognitive Interference**, where the entropy of the prompt H_{prompt} is divided. 1) **Decoupled Sampling**: The Radical agent ($\mathbf{F}_{rad}$) samples from an unconstrained distribution $\mathcal{P}_{rad}$ with high entropy, maximizing the discovery of performance-intensive manifolds. 2) **Manifold Projection**: The Conservative agent ($\mathbf{F}_{con}$) acts as a **Non-expansive Projection Operator** $\Pi_{\mathbb{M}_\Phi} : \mathbb{M} \rightarrow \mathbb{M}_\Phi$. It performs local repair $\mathcal{R}$ using the SMT counter-example ξ. Under a fixed budget S, the probability of hitting the Pareto front $\mathbb{P}(\mathcal{G} \in \mathbb{M}_{novel} \cap \mathbb{M}_\Phi)$ is maximized because each agent solves a sub-problem with lower combinatorial complexity. This follows from the **Data Processing Inequality**: partitioning the task reduces the noise introduced by conflicting multi-objective instructions in a single context window. □

Lemma C3.2 (Stability of the Evolutionary Loop)**.** *The evolutionary sequence $\{A_t\}$ converges to the stable manifold $\mathbb{M}_\Phi$ under the dynamic annealing schedule.*

Proof. Define a discrete-time Lyapunov function $V(A_t) = d_{TED}(A_t, \mathbb{M}_\Phi)$, where d_{TED} is the **Tree Edit Distance** to the nearest feasible axiom. 1) **Annealing Hardening**: As $t \rightarrow T$, the penalty $\lambda_v(t)$ increases the rejection probability of any $A \notin \mathbb{M}_\Phi$. 2) **Drift toward Feasibility**: The Repair Operator $\mathcal{R}$ ensures that for any unstable proposal $\tilde{A}$, the repaired version $\bar{A}$ satisfies $V(\bar{A}) = 0$. 3) **Convergence**: Since the reward function $J(A)$ enforces selection pressure, the probability $P(A_t \notin \mathbb{M}_\Phi) \rightarrow 0$ as $t \rightarrow T$. Thus, $V(A_t)$ is a super-martingale that converges to 0, proving search stability within the safety envelope. □

C4. Formal Verification and SMT Soundness

The verification engine $V(\mathcal{G})$ utilizes SMT (Satisfiability Modulo Theories) to prove the **Safety Invariant** Φ.

Soundness over Bounded Intervals: For non-linear operators (e.g., $\exp, \tanh$), we adopt soundness restricted to the search domain $\mathcal{C} \subset \mathbb{R}^n$. The Z3 solver performs **Interval Arithmetic Propagation**. If the solver returns *UNSAT* for $\mathcal{C} \wedge \mathbb{F}(\mathcal{G}) \wedge \neg\Phi$, it is mathematically guaranteed that no state within the bounded intervals can trigger a violation under the given floating-point precision.

SafeCut and Logical Confluence: The repair operator $\mathcal{R}$ utilizes the counter-example ξ to perform a **Structural Projection**. It identifies the sub-tree responsible for the violation and replaces it with a saturated primitive (e.g., $x \rightarrow \max(\epsilon, x)$). This ensures the modified axiom $\mathcal{G}^*$ is a **Logical Confluence point** that preserves original reasoning while satisfying the safety invariant.