ATLAS-SR: A PROTOCOL-INDEXED STRUCTURAL ATLAS FOR LLM-GUIDED SYMBOLIC REGRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Symbolic regression seeks interpretable mathematical expressions that capture relationships in observed data. During search, useful components can appear in candidate equations whose overall fit is poor. We present ATLAS-SR, a framework that uses large language models (LLMs) to guide the recombination of components retained from earlier candidates. The framework tests component contributions within existing equations under consistent fitting conditions. It stores components in a shared graph alongside their tested combinations and the results of these comparisons. These records help the LLM choose which components to reuse and where to try new combinations, allowing useful structures to outlive their original candidates. We evaluate ATLAS-SR on 129 LLM-SRBench tasks spanning four scientific domains with two LLM backbones. Strict out-of-distribution accuracy ($\mathrm{Acc}_{0.01}^{95\%}$) improves by 28.69–29.46 percentage points over the strongest baseline under each backbone. Symbolic accuracy, which measures recovery of the target equation structure, reaches 24.03% with either backbone, exceeding the highest baseline result of 16.28%.

1 INTRODUCTION

Symbolic regression seeks interpretable equations that describe observations and generalize to unseen inputs. It requires searching over mathematical structures while fitting their continuous parameters (Schmidt & Lipson, 2009; Brunton et al., 2016; Cranmer, 2023). This process produces many candidate equations, each containing components that may be useful elsewhere. A candidate can fit poorly because these components are combined incorrectly or because an important component is still missing. Retaining useful parts of such attempts would allow subsequent search to build progressively more complete equations.

Recent work shows that large language models (LLMs) can use search experience to improve equation discovery. LLM-SR combines equation programs, numerical fitting, and retrieval of earlier candidates, improving discovery and out-of-domain generalization (Shojaee et al., 2025a). SR-Scientist extends this interaction through tool-driven data analysis and long-horizon optimization; its ablations establish the value of retaining useful equation examples (Xia et al., 2026a). Both methods organize their experience buffers around complete equations and scores. An overall score, however, leaves the contributions of individual components unresolved. When retrieval favors better-fitting equations, useful components in weaker candidates can receive less attention.

Other approaches make finer-grained experience reusable. LaSR evolves a concept library that accelerates discovery (Grayeli et al., 2024). IGSR retains term-level feedback along search paths and uses influence to guide proposals and pruning, including deletion scores that account for refitting (Saveliev et al., 2026). PTB-Search pools terms from independent proposals into a dictionary and selects them jointly, demonstrating the value of combinations assembled across source equations (Li, 2026). These results establish useful ways to preserve ideas, assess components, and combine them.

Graph-based methods provide another way to organize this search. GraphDSR represents expressions as directed acyclic graphs and learns to generate their nodes and connections (Liu et al., 2025). Equality graphs group algebraically equivalent expressions: eggp uses them to avoid revisiting earlier expressions, while EGG-SR shares search statistics and feedback across equivalent

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

forms (de França & Kronberger, 2025; Jiang et al., 2026). These uses improve expression construction and the reuse of search history. For component reuse, a further question matters: which connections between components helped explain the data, and how can that evidence guide a new combination?

For example, a periodic component may become useful when multiplied by a population-dependent branch, even if adding it separately gives little improvement. Its value depends on how the two parts connect. The apparent effect of deleting it also depends on whether the remaining parameters are allowed to compensate. Reusing this experience calls for a record of the tested combination and its fitting conditions. The remaining challenge is to make these local tests available as the search constructs new equations and changes how their parts connect.

We introduce ATLAS-SR, which uses a shared graph to preserve components and evidence about their use. A component keeps the same identity across candidates, and graph connections record its roles in different equations. The system removes a component and refits the remaining equation under matched conditions to assess its contribution. It stores the result with the tested combination and fitting conditions in a **Structural Atlas**. A component can then remain available with its earlier tests after its source equation leaves the active search. Components found along different search paths can be retrieved and connected to form new candidates.

This graph gives the LLM access to both available components and the outcomes of their earlier uses. A component supported by a local test can receive further exploration even when its source equation has a poor overall fit. Successful combinations guide reuse, while weak or conflicting evidence identifies tests worth continuing. The LLM prioritizes these actions, and a graph editor constructs the proposed equations. New fits and component tests update the atlas. Through this loop, evidence accumulates around reusable parts and helps determine how they should be combined and evaluated next.

We evaluate ATLAS-SR on 129 LLM-SRBench tasks across four scientific domains with DeepSeek-V4-Flash and GPT-4o-mini (Shojaee et al., 2025b). Strict out-of-distribution (OOD) accuracy reaches 85.27% and 84.50%, improving over the strongest baseline under each backbone by 29.46 and 28.69 percentage points, respectively. Both backbones reach 24.03% symbolic accuracy. Ablations support the value of retaining component-level evidence and distinguishing the conditions under which it was obtained. Trajectory analysis further examines where useful components remain unassembled and where structurally different equations achieve similarly low errors.

Our contributions are threefold:

- We propose ATLAS-SR, a framework whose shared graph preserves reusable components, how they were combined, and the outcomes and conditions of their tests across search rounds.
- We develop a search loop that uses these records to guide further assembly, fitting, and component tests, and demonstrate gains across 129 tasks and two LLM backbones.
- We analyze search trajectories to identify incomplete assembly and low-error structural alternatives as obstacles to recovering the generating equation.

2 Related Work

LLM-guided equation search and feedback. ICSR and LLM4ED iteratively refine functional forms using fitting feedback, while LLM-SR combines executable equation programs with evolutionary search (Merler et al., 2024; Du et al., 2024; Shojaee et al., 2025a). Agentic methods extend this process to data analysis and search control. SR-Scientist interleaves tool use with hypothesis revision (Xia et al., 2026a); KeplerAgent uses physical analysis to configure symbolic regression engines (Yang et al., 2026). LLM-PySR plans features, operators, and search depth, then records numerically evaluated outcomes for later planning (Xie et al., 2026). DrSR and programmatic context augmentation extract diagnostic information from data to guide subsequent proposals (Wang et al., 2025; Liu et al., 2026). These methods broaden the feedback available to search and give LLMs roles beyond generating complete equations.

Representations and reusable search experience. How expressions are represented determines which structures can be edited and shared. FunctionEvolve uses abstract syntax trees for local edits and structure-aware fitting (Xia et al., 2026b), while GraphDSR generates directed acyclic expression graphs with a graph neural network (Liu et al., 2025). Equality graphs organize algebraically equivalent expressions: eggp uses this history to avoid repeated exploration, and EGG-SR applies equivalence information to pruning and learning feedback (de França & Kronberger, 2025; Jiang et al., 2026). Other methods reuse more abstract experience. LaSR evolves a concept library (Grayeli et al., 2024); STRIDE groups equations in semantic memory, and Deliberate Evolution retains reflections on search trajectories (Su et al., 2026; Pang et al., 2026). A-SR routes motifs, failures, and diagnostics among specialized agents (Zhao et al., 2026). These approaches preserve complementary information about past search. ATLAS-SR connects reusable structures directly to the assemblies and evaluation conditions under which their contributions were tested.

Component evaluation and experimental selection. IGSR combines term-level influence with Monte Carlo tree search and studies both frozen-coefficient and refit-aware deletion scores (Saveliev et al., 2026). PTB-Search pools terms from independent proposals into a dictionary and jointly selects sparse subsets, allowing terms to outlive their source equations (Li, 2026). LLM-ACES uses disagreement between candidate trajectories to select new initial conditions for data acquisition (Abhyankar et al., 2026). Together, these studies examine how to evaluate components, choose combinations, and distinguish competing hypotheses. ATLAS-SR focuses on computational comparisons over the available data: matched-refit results remain linked to the tested component, assembly, and protocol across rounds. The same record can guide a new structural edit or identify a comparison that still needs to be completed.

3 ATLAS-SR

Given training data $\mathcal{D}^{\mathrm{tr}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$, we seek a concise expression f that fits the observations and generalizes. Task context $\mathcal{T}$ describes the problem and its variables. Figure 1 follows the loop: the LLM guides combinations of stored structures, numerical fitting evaluates the resulting equations, and component tests update the shared record. Subsequent rounds use these results for new combinations and unfinished tests. We explain component evaluation, preservation, and reuse below; Algorithm 1 summarizes the loop.

3.1 Evaluating Components for Reuse

A poorly fitting equation can contain a component worth keeping. We call an equation formed by combining components an *assembly*; its relations specify how those components connect, such as the factors in a product. To assess a component's contribution, we remove it and allow the remaining parameters to adapt. We call this comparison **matched refit** (MR): both sides use the same data, variable representation, optimizer and restart policy, fitting stage, and budget. Denote these settings by the evaluation protocol π. For a fitted assembly S, a tested component or relation u, and training-side loss $\mathcal{L}_\pi$, the comparison is

$$\Delta^{\pi}_{\mathrm{MR}}(u \mid S) = \log_{10} \frac{\max\{\mathcal{L}_\pi(\mathsf{Refit}_\pi(S \setminus \{u\})), \varepsilon\}}{\max\{\mathcal{L}_\pi(S), \varepsilon\}}. \tag{1}$$

Here $\varepsilon > 0$ is a numerical loss floor. A positive value means the reduced assembly still fits worse after refitting; a negative value means deletion achieved a better fit. Thus, even when S has a poor overall fit, the comparison can support retaining u for further combinations. The evidence concerns its contribution within S, and a new assembly receives its own evaluation. Complexity and numerical risk enter ranking separately.

Refitting allows the remaining components to compensate for deletion. In a frozen-deletion comparison, their parameters stay fixed. If refitting achieves a loss no greater than that obtained with these fixed parameters, its deletion effect is bounded above by the frozen-deletion effect (Proposition 3). Matching the fitting settings also separates structural changes from changes in evaluation conditions (Proposition 2). Appendix D derives the corresponding deletion loss for linear least squares.

The graph identifies the parts that can be tested: additive members, multiplicative factors, quotient roles, and registered subgraphs. Deletion follows the corresponding assembly rule. For a general

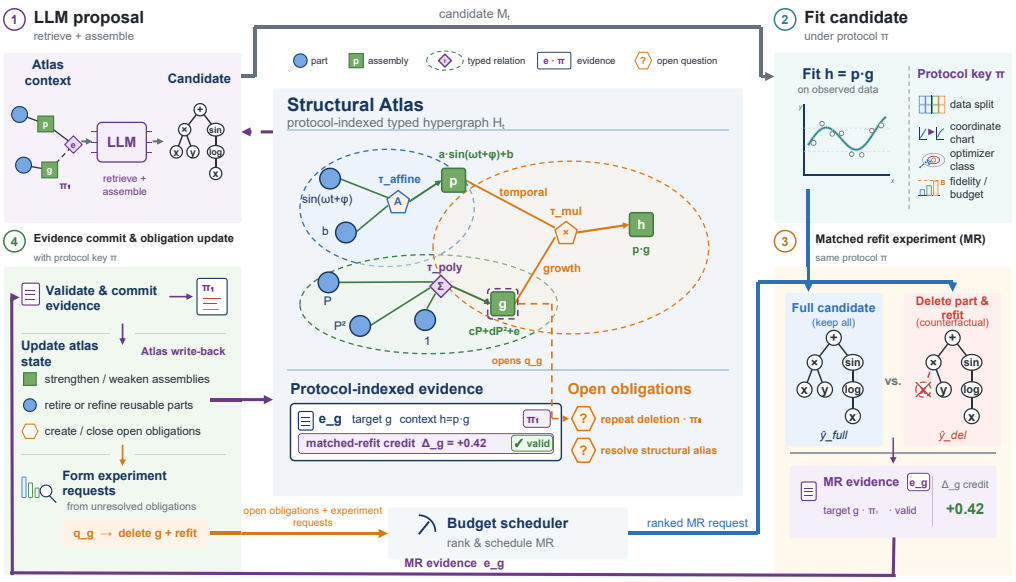

Figure 1: The ATLAS-SR search loop. Stored components and experimental results guide candidate proposals. Fitting and component comparisons produce new evidence, which returns to the atlas and informs subsequent combinations and tests. Unfinished comparisons carry over to later rounds.

subgraph, the system substitutes a registered, refittable scalar baseline compatible with its type, allowing the reduced equation to remain evaluable. Completed, numerically valid, protocol-matched comparisons update the evidence; unresolved outcomes remain pending. Appendix A.3 gives the deletion rules and thresholds.

3.2 Preserving Components and Their Tested Combinations

For these comparisons to guide future combinations, the search must preserve both the components and the circumstances in which they helped. We store them in the **Structural Atlas**, denoted $\mathfrak{A}_t$ at round t. Each candidate executes as a directed acyclic graph (DAG). Across candidates, a shared typed hypergraph gives components stable identities and links them through their assembly relations. Each relation records the participating components and their roles, such as a function and its argument or the factors in a product. The same component can belong to several assemblies while retaining the results of each earlier use. These links let the search retrieve a component together with its tested combinations and connect it to components found in other candidates.

For example, a periodic component used as an additive term and as a product factor has a separate link to each assembly. Each link leads to its own tests. A later proposal can therefore retrieve the product together with evidence about that use of the periodic component. The basic evidence record is

$$e = (u, S, \pi, \Delta, \rho), \qquad \pi = (\mathcal{D}^{\text{fit}}, \alpha, \chi, \phi, \mathbf{b}). \tag{2}$$

It links component or relation u to the tested assembly S, protocol π, and measured effect Δ. The protocol specifies the fitting data, variable representation α, optimizer configuration χ, fitting stage ϕ, and budget $\mathbf{b}$. The field ρ stores validity checks, fitting diagnostics, and realized cost. Results from different assemblies or protocols remain separate, so later decisions can recover the conditions supporting each observation.

Components and their completed tests remain available after their source equations leave the active set. Weak or conflicting evidence creates pending comparisons, and components awaiting these tests remain available. Deactivation requires resolved comparisons, sufficient compatible tests, and several rounds without positive evidence or useful descendants. New evidence can reactivate the

structure. Appendix D specifies the full state; Appendix D.2.1 shows how retained component records distinguish histories with the same current equations and scores.

3.3 Combining Components and Continuing the Search

At each round, the LLM receives components, tested combinations, fitting diagnostics, and pending comparisons along with $\mathcal{T}$. Its action plan a prioritizes new combinations and further tests of existing ones. The graph editor constructs equations through type-compatible composition, wrapping, replacement, deletion, or restructuring. Verified mappings permit reuse across variable representations.

Each assembly specifies its components and their input roles, with parameters fitted for the new combination. Edits can also change coefficient sharing. In the product expansion of Section 4.4, each additive term receives an independent coefficient, allowing periodic and nonperiodic terms to adjust separately. The expanded family receives its own joint fit and structural record, leaving stored parent parameters unchanged.

Algorithm 1 ATLAS-SR search loop

Require: $\mathcal{D}^{\mathrm{tr}}, \mathcal{T}$; seeds $\mathcal{S}_0$, budget $\mathbf{B}^{\mathrm{tot}}$
$\mathfrak{A} \leftarrow \textsc{Initialize}(\mathcal{D}^{\mathrm{tr}}, \mathcal{T}, \mathcal{S}_0, \mathbf{B}^{\mathrm{tot}})$
while $\textsc{CanContinue}(\mathfrak{A})$ **do**
  $R \leftarrow \textsc{Read}(\mathfrak{A})$
  $a \leftarrow \textsc{LLMGuide}(R, \mathcal{T})$
  $C \leftarrow \textsc{Assemble}(a, \mathfrak{A})$
  $C \leftarrow \textsc{CheckDeduplicate}(C)$
  $F \leftarrow \textsc{Fit}(C, \mathcal{D}^{\mathrm{tr}}, \mathfrak{A})$
  $Q \leftarrow \textsc{Requests}(\mathfrak{A}, F)$
  $W \leftarrow \textsc{Schedule}(Q, \mathfrak{A})$
  $O \leftarrow \textsc{MatchedRefit}(W, F, \mathfrak{A})$
  $J \leftarrow \textsc{ValidateMatch}(F, O)$
  $\mathfrak{A} \leftarrow \textsc{Commit}(\mathfrak{A}, C, F, J)$
  $\mathfrak{A} \leftarrow \textsc{CarryPending}(\mathfrak{A}, Q, W, J)$
end while
return $\textsc{SelectBest}(\mathfrak{A})$

Candidates pass type and domain checks, normalization, and deduplication before fitting. Early screening and more thorough optimization have separate protocol records, so an unfinished fit can be revisited with its earlier results intact. We use variable projection when linear and nonlinear parameters separate, and additional search and local refinement for difficult fits. Further-fitting priority combines training loss, underfitting diagnostics, estimated fitting potential, useful extensions, and declared evaluation cost (Appendix A.4).

Validated results return to the atlas. Positive evidence supports retention and reuse; unresolved comparisons request further fitting or another matched test. Fixed queue quotas divide a shared fitting and MR budget among pending comparisons, promising candidates, and numerical risks (Appendix A.4). In Algorithm 1, Q includes new and pending requests. Fitting and MR update the shared resource ledger; CanContinue checks remaining budget and available work. CarryPending preserves unserved or unresolved requests with their context.

At termination, the selector compares completed fits across the archive, including candidates outside the final active set. It first selects a predictive candidate using training fit, stress diagnostics, proxy risk, and simplified complexity. An alternative preferred by the archive's structural checks can replace it after passing eligibility, stability, and training-loss checks (Appendix A.5). Selection uses existing fitted equations; test data, OOD data, and ground-truth equations enter only during evaluation. Appendices D.1.7 and D.1.9 give the detailed state transitions and implementation pseudocode.

4 Experiments

4.1 Experimental Setup

We evaluate the complete ATLAS-SR search loop against existing methods. Internal ablations examine evidence construction and reuse, and trajectory analysis identifies remaining structural difficulties. We use the 129-task LSR-Synth collection of LLM-SRBench: chemical reaction kinetics (36 tasks), biological population growth (24), physical oscillators (44), and materials science (25) (Shojaee et al., 2025b). All search decisions use the training split; test and OOD data are reserved for final evaluation.

Table 1: LSR-Synth results by domain and overall. Threshold accuracies are evaluated on OOD data; SA and accuracies are percentages, and test NMSE is the median over finite predictions. DeepSeek and GPT abbreviate the two interfaces; arrows indicate the preferred direction, and boldface marks the best result per interface.

Interface	Chemistry (36)			Biology (24)			Physics (44)			Materials (25)			Overall (129)			
	$\mathrm{Acc}_{0.1}^{95\%}$	$\mathrm{Acc}_{0.01}^{95\%}$	NMSE↓	$\mathrm{Acc}_{0.1}^{95\%}$	$\mathrm{Acc}_{0.01}^{95\%}$	NMSE↓	$\mathrm{Acc}_{0.1}^{95\%}$	$\mathrm{Acc}_{0.01}^{95\%}$	NMSE↓	$\mathrm{Acc}_{0.1}^{95\%}$	$\mathrm{Acc}_{0.01}^{95\%}$	NMSE↓	$\mathrm{Acc}_{0.1}^{95\%}$	$\mathrm{Acc}_{0.01}^{95\%}$	NMSE↓	SA ↑
Non-LLM baselines (Cranmer, 2023; Tenachi et al., 2023)																
PySR	44.44	11.11	5.93e−5	16.67	12.50	1.78e−4	38.64	11.36	4.73e−4	72.00	36.00	8.76e−6	42.64	16.28	8.47e−5	5.43
PhySO	36.11	11.11	1.45e−3	16.67	4.17	3.01e−4	9.09	0.00	0.0198	36.00	12.00	2.19e−4	23.26	6.20	2.11e−3	0.78
Direct prompting																
DeepSeek	11.11	0.00	3.98e−3	8.33	4.17	0.0303	11.36	2.27	5.75e−3	24.00	8.00	1.03e−3	13.18	3.10	5.12e−3	0.00
GPT	11.11	0.00	0.0185	4.17	0.00	0.5943	2.27	2.27	0.0227	32.00	4.00	5.12e−3	10.85	1.55	1.98e−2	0.00
LaSR (Grayeli et al., 2024)																
DeepSeek	27.78	5.56	2.81e−4	16.67	0.00	3.67e−4	20.45	9.09	1.45e−3	48.00	8.00	1.04e−4	27.13	6.20	3.91e−4	1.55
GPT	30.56	8.33	1.73e−4	16.67	4.17	4.88e−4	20.45	13.64	1.06e−3	60.00	8.00	3.26e−5	30.23	9.30	3.94e−4	3.10
LLM-SR (Shojaee et al., 2025a)																
DeepSeek	44.44	11.11	9.23e−6	25.00	12.50	5.71e−5	45.45	18.18	2.96e−5	88.00	52.00	9.00e−8	49.61	21.71	1.18e−5	3.10
GPT	38.89	11.11	5.12e−5	16.67	8.33	1.09e−4	36.36	13.64	9.75e−5	88.00	68.00	1.26e−8	43.41	22.48	3.94e−5	3.88
SR-Scientist (Xia et al., 2026a)																
DeepSeek	50.00	13.89	9.02e−5	58.33	33.33	2.40e−6	54.55	34.09	1.38e−5	92.00	72.00	1.86e−11	61.24	35.66	4.63e−6	5.43
GPT	41.67	8.33	5.03e−5	16.67	4.17	4.68e−4	18.18	6.82	1.79e−3	84.00	24.00	2.35e−6	37.21	10.08	1.45e−4	2.33
IGSR (Saveliev et al., 2026)																
DeepSeek	80.56	66.67	1.66e−8	70.83	58.33	2.11e−9	59.09	38.64	7.03e−6	92.00	68.00	2.35e−8	73.64	55.81	3.13e−7	10.85
GPT	88.89	72.22	1.63e−8	70.83	58.33	2.93e−9	68.18	36.36	7.62e−6	**96.00**	64.00	7.52e−9	79.84	55.81	1.42e−7	16.28
ATLAS-SR (Ours)																
DeepSeek	**94.44**	**91.67**	6.28e−11	**95.83**	**91.67**	6.47e−13	**86.36**	**70.46**	7.20e−9	**100.00**	**96.00**	2.37e−14	**93.02**	**85.27**	3.16e−11	**24.03**
GPT	**94.44**	**91.67**	4.25e−11	**91.67**	**87.50**	6.02e−13	**86.36**	**72.73**	4.33e−9	**96.00**	**92.00**	2.90e−14	**91.47**	**84.50**	5.06e−11	**24.03**

DeepSeek-V4-Flash and GPT-4o-mini serve as the two backbones, with one fixed ATLAS-SR search configuration across tasks. Baselines are rerun on the same frozen task list, splits, metrics, and post-processing, following their released search configurations. IGSR uses official revision `44af1f48`. We report median test normalized mean squared error (NMSE), OOD $\mathrm{Acc}_{0.1}^{95\%}$ and $\mathrm{Acc}_{0.01}^{95\%}$, and symbolic accuracy (SA). A task satisfies $\mathrm{Acc}_{\tau}^{95\%}$ when at least 95% of OOD points have predefined relative error no greater than τ. SA follows the LLM-SRBench equivalence-evaluation procedure (Shojaee et al., 2025b). Appendices A and B document the settings, metrics, and resource accounting.

4.2 Equation Discovery Quality

Table 1 shows that ATLAS-SR achieves the best strict OOD accuracy and median test NMSE in all four domains with both backbones. Overall $\mathrm{Acc}_{0.01}^{95\%}$ rises from IGSR's 55.81% to 85.27% with DeepSeek-V4-Flash and 84.50% with GPT-4o-mini. Figure 2 shows that the numerical gains extend across the task-level error distribution.

The structural gains are also consistent across backbones. We apply the same evaluation pipeline to all 1,806 final expressions. ATLAS-SR recovers 31/129 equations (24.03%) with either backbone, compared with a baseline best of 21/129. Exact recovery remains much less frequent than strict numerical success. Section 4.4 examines the search stages at which recovery stops.

Where computation is spent. Local fitting and MR produce evidence that the atlas uses to guide later assembly and evaluation. With DeepSeek-V4-Flash, ATLAS-SR uses a median of 14 LLM calls and 165,682 tokens per task, versus 163 calls and 305,645 tokens for IGSR. Local CPU use rises from 556.81 to 1,833.74 core-seconds. This allocation combines fewer LLM interactions with more local fitting and testing within the search loop. Appendix B.3 reports the full resource distributions and accounting rules.

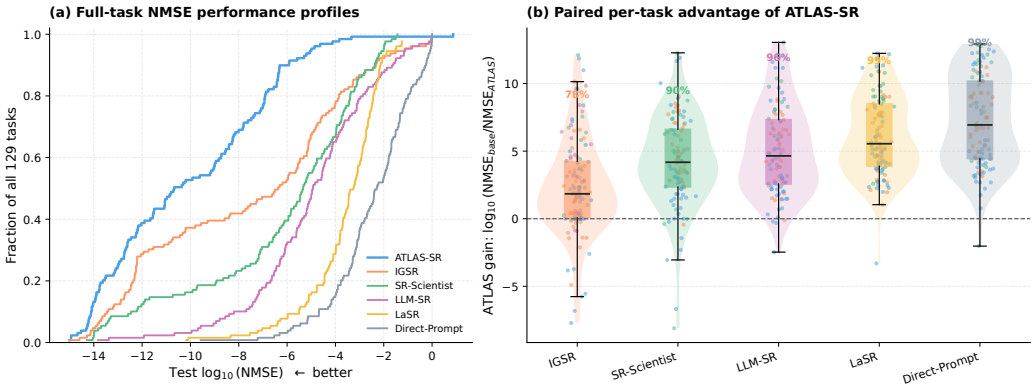

Figure 2: Task-level test NMSE with DeepSeek-V4-Flash. Left: empirical CDF of $\log_{10}(\text{NMSE})$. Right: per-task log-NMSE difference between each baseline and ATLAS-SR; positive values favor ATLAS-SR.

4.3 What Must Persist across Rounds?

All ablations use DeepSeek-V4-Flash and the same frozen setup. Table 2 compares selected configurations; Appendix Table 6 reports all ten variants.

Continuity and structural context. Independent round restarts and flat candidate memory both reach 64.34% strict OOD accuracy, down from 85.27% with the full atlas. Their SA values are 22.48% and 4.65%, respectively. Restarts recover nearly as many equations as the full method. With persistent state and structured feedback, the full atlas achieves substantially broader high-precision coverage. The larger SA loss with flat memory highlights the value of retaining tested component relationships alongside candidate scores. Appendix Table 7 specifies state access and feedback for each variant.

The score-broadcast and passive-atlas variants examine how evaluation results inform subsequent search. Score broadcast supplies candidate-level scores in place of structured feedback about tested combinations. The passive atlas records experiments while proposal, fitting, and scheduling proceed without reading its records. Its strict OOD accuracy is 62.02%, compared with 85.27% for the full method (Appendix Table 6). These contrasts support the role of evidence reuse in guiding further assembly and evaluation.

Evidence quality and subsequent decisions. The quality of a stored comparison affects what the search learns from it. Frozen deletion attributes value before the remaining parameters can compensate; removing protocol indexing pools results obtained under different fitting conditions. Both changes reduce OOD accuracy. Disabling evidence feedback and randomizing scheduling also degrade performance. These results connect the value of stored evidence to its use in proposals and further experiments. The no-LLM variant reaches 34.11% OOD accuracy in the extended ablation. Together, these results support a complementary role for LLM guidance and numerical tests of the structures being reused.

4.4 From Partial Discoveries to Complete Equations

Figure 4 summarizes structural assembly and evidence reuse in BPG12. The search finds a periodic carrier $p = A\sin(\omega t+\phi)+B$ and a growth branch $g = CP+DP^2+E$ in separate candidates. The atlas preserves both structures for later combination, allowing their usefulness to be tested beyond their original candidates.

We expand the product pg and assign an independent coefficient to each resulting term. Writing $s = \sin(\omega t + \phi)$, the resulting candidate is

$$\widetilde{h}(P,t) = \boldsymbol{\theta}^\top \boldsymbol{\psi}(P,t), \qquad \boldsymbol{\psi} = (Ps, P^2 s, s, P, P^2, 1)^\top. \tag{3}$$

Table 2: State and evidence ablations with DeepSeek-V4-Flash. Metrics are percentages; Δ is the OOD change from the full model.

Configuration	ID $\mathrm{Acc}_{0.01}^{95\%}$ ↑	OOD $\mathrm{Acc}_{0.01}^{95\%}$ ↑	SA ↑	Δ OOD
ATLAS-SR (Ours)	**89.15**	**85.27**	**24.03**	**0.00**
Flat candidate memory	65.89	64.34	4.65	-20.93
Independent round restart	70.54	64.34	22.48	-20.93
Candidate-score broadcast	66.67	62.02	11.63	-23.25
No evidence feedback	76.74	73.64	15.50	-11.63
Frozen deletion	74.42	66.67	14.73	-18.60
Random scheduling	79.85	75.97	19.38	-9.30
No protocol indexing	61.24	60.47	13.18	-24.80

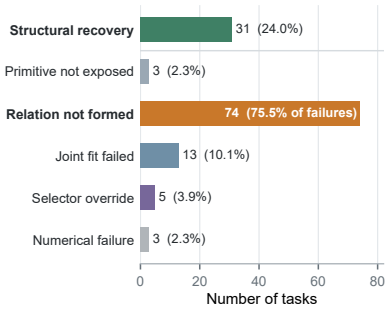

Figure 3: Discovery stages over 129 tasks. Of 98 failures, 74 stop before target relations form.

In the original product, the coefficients of Ps and P are AC and BC; the same branch weights also constrain the other terms. Jointly fitting six free coefficients lets the search adjust periodic and non-periodic growth separately, while retaining the inherited sine and polynomial structure. The archived fit reaches a test NMSE of 6.84×10^{-13}; the right subgraph summarizes its dominant terms, with smaller terms represented by an ellipsis.

Four MR comparisons of an extended child, $\widetilde{h} + c|t|^q$, yield $\Delta = 11.28$–11.99. The atlas links these outcomes to the tested assembly and protocol, preserving the child's connection to its parent. The LLM cites the comparisons when protecting the parent assembly and requesting further reuse and extension. A later request can thus refer to a particular experiment while drawing on the carrier and growth branch already stored in the atlas. The bottom row summarizes final selection from the completed executable archive using training fit, stress diagnostics, proxy risk, and complexity.

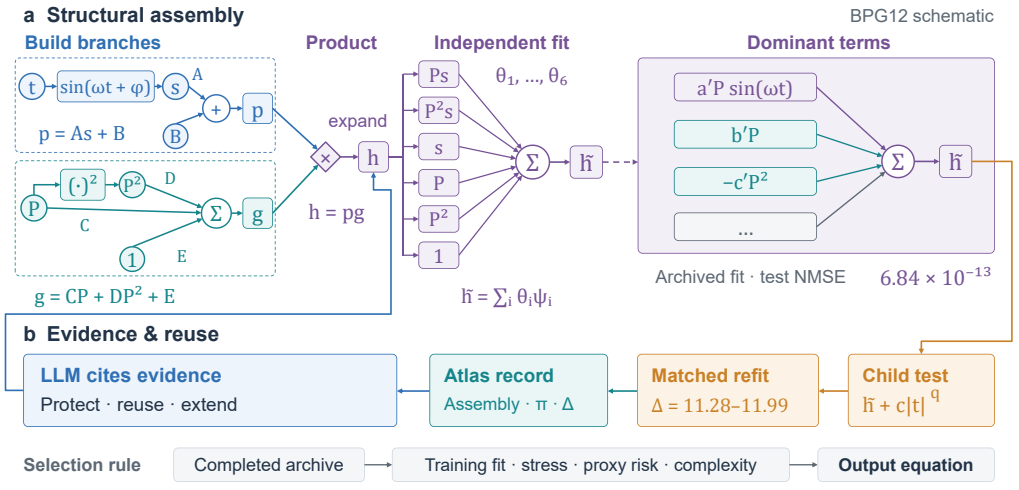

Figure 4: BPG12-based schematic of assembly and evidence reuse. (a) A product yields six basis terms with independent coefficients fitted jointly; the dashed link summarizes the dominant fitted structure. (b) Child comparisons produce evidence for protection and reuse. The bottom row gives the method-level final-selection rule.

Where structural recovery stops. We audit all 129 trajectories through primitive exposure, assembly, joint fitting, retention, and final selection. In 74 of the 98 structural failures, search exposes the required operators and stops before assembling their target relations (Figure 3). The main obstacle in these trajectories is combining available components into a complete equation.

Relational Combination Burden (RCB) summarizes four demands on assembly. These involve parameterized transforms, multiplication of transformed terms by external factors, non-additive combinations of variables, and multiple independent branches. Recovery declines as these requirements accumulate. The structural-recovery gain is concentrated at RCB 0–1, while higher-burden groups remain difficult across methods (Appendix C). RCB is a post-search diagnostic of remaining composition difficulty.

An **Operator Shadow** (OS) occurs when an incorrect structure closely approximates a missing operator over the observed range. Such candidates can survive numerical selection even when the required relation never forms. Appendix C.4 quantifies these alternatives by projection onto their fitted bases. Together, the assembly audit and OS analysis distinguish two obstacles: constructing the target equation and separating it from accurate alternatives on finite data.

4.5 Equation Discovery under Noisy Observations

We replace each training label y_i with $y_i(1 + \epsilon z_i)$, where $z_i \sim \mathcal{N}(0, 1)$ and $\epsilon \in \{0.01, 0.03, 0.05\}$, leaving test and OOD data unchanged. All 129 tasks use DeepSeek-V4-Flash and the same search configuration. At 1% noise, coarse OOD accuracy remains 62.79%, while SA falls to 13.95%. At 5% noise, strict OOD accuracy and SA fall to 6.98% and 0.78%, respectively (Table 3).

Table 3: Performance under training-label noise. OOD accuracies and SA are percentages; NMSE is the median test error.

Setting	SA ↑	$\mathrm{Acc}_{0.1}^{95\%}$ ↑	$\mathrm{Acc}_{0.01}^{95\%}$ ↑	NMSE↓
Clean	24.03	93.02	85.27	3.16e−11
1% noise	13.95	62.79	46.51	4.45e−6
3% noise	8.52	33.33	25.58	1.84e−5
5% noise	0.78	17.83	6.98	7.13e−4

The preceding analyses suggest how noise may affect search. The full and reduced assemblies can respond differently to the same noisy labels, making small contributions harder to distinguish. Alternative expressions can also fit these observations well. The resulting comparisons may guide retention and reuse toward structures that leave required relations incomplete. Appendix D develops the mathematical analysis. These effects motivate uncertainty-aware evaluation of accumulated evidence.

5 Limitations and Future Work

The evaluation covers synthetic, task-local discovery. Generalization to real measurements and transfer between tasks remain open. Exact recovery is still limited for difficult compositions, and label noise weakens both numerical and structural performance. RCB and OS use ground truth after search to diagnose difficulties in structural recovery. Future work should study uncertainty-aware evidence, cross-task structural reuse, and observations chosen to distinguish competing equations.

6 Conclusion

We introduced ATLAS-SR, a symbolic regression framework that builds on components discovered in earlier candidates. Its Structural Atlas preserves these components together with their tested combinations and fitting conditions. Matched-refit evidence then guides LLM-directed assembly, further fitting, and subsequent comparisons. Evaluation on 129 tasks with two backbones shows improved numerical accuracy and symbolic recovery, while ablations examine the roles of memory and evidence feedback. Trajectory analysis identifies a broader challenge for equation discovery: assembling the required relations and distinguishing the generating equation from accurate numerical alternatives. Preserving partial discoveries and testing their contributions provides a practical basis for addressing this challenge.

AI use statement

Generative AI assistants, including OpenAI Codex, supported this work in three areas. For writing, they assisted with Chinese-to-English translation, language polishing, and revision of paragraph structure and transitions. For retrieval and discovery, they helped locate relevant studies, summarize candidate references, and organize bibliographic entries for verification against the original

sources. For research execution, they assisted with building and revising experimental code, preparing analysis and plotting scripts, and refining figure layouts. The authors reviewed AI-assisted outputs and checked scientific claims, numerical results, citations, and final artifacts against the source manuscript and experimental records. The authors take responsibility for the final content of this work.

Reproducibility statement

Appendix A documents the tasks, metrics, model settings, search configuration, and output selection. Appendix B provides supplementary results and resource accounting. Appendix D specifies the mathematical framework, update algorithm, and supporting proofs.

References

Nikhil Abhyankar, Sha Li, Sanchit Kabra, Naren Ramakrishnan, Yulia Gel, and Chandan K. Reddy. LLM-ACES: Closed-loop discovery of dynamical systems with LLM-guided adaptive search, 2026. URL https://arxiv.org/abs/2606.25039.

Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016. doi: 10.1073/pnas.1517384113.

Miles Cranmer. Interpretable machine learning for science with PySR and SymbolicRegression.jl, 2023. URL https://arxiv.org/abs/2305.01582.

Fabrício Olivetti de França and Gabriel Kronberger. Improving genetic programming for symbolic regression with equality graphs. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 989–998. ACM, 2025. doi: 10.1145/3712256.3726383.

Mengge Du, Yuntian Chen, Zhongzheng Wang, Longfeng Nie, and Dongxiao Zhang. LLM4ED: Large language models for automatic equation discovery, 2024. URL https://arxiv.org/abs/2405.07761.

Arya Grayeli, Atharva Sehgal, Omar Costilla-Reyes, Miles Cranmer, and Swarat Chaudhuri. Symbolic regression with a learned concept library. In *Advances in Neural Information Processing Systems*, volume 37, pp. 44678–44709, 2024. doi: 10.52202/079017-1419.

Nan Jiang, Ziyi Wang, and Yexiang Xue. EGG-SR: Embedding symbolic equivalence into symbolic regression via equality graph. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=oh9ChF7Pv0.

Pan Li. Dictionaries, not darwin: Set-level selection beats LLM evolution in scientific equation discovery, 2026. URL https://arxiv.org/abs/2607.04108.

Hao Liu, Xiao-Wen Yang, Atharva Sehgal, Yixin Wang, Lan-Zhe Guo, Yu-Feng Li, and Yisong Yue. Programmatic context augmentation for LLM-based symbolic regression, 2026. URL https://arxiv.org/abs/2605.03101.

Jingyi Liu, Weijun Li, Lina Yu, Min Wu, Wenqiang Li, Yanjie Li, and Meilan Hao. Mathematical expression exploration with graph representation and generative graph neural network. *Neural Networks*, 187:107405, 2025. doi: 10.1016/j.neunet.2025.107405.

Matteo Merler, Katsiaryna Haitsiukevich, Nicola Dainese, and Pekka Marttinen. In-context symbolic regression: Leveraging large language models for function discovery. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pp. 427–444. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-srw.49.

Xinyu Pang, Zhanke Zhou, Xuan Li, Fangrui Lv, Shanshan Wei, Sen Cui, Bo Han, and Changshui Zhang. Deliberate evolution: Agentic reasoning for sample-efficient symbolic regression with LLMs. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://arxiv.org/abs/2606.04360.

Evgeny S. Saveliev, Samuel Holt, Nabeel Seedat, David L. Bentley, Jim Weatherall, and Mihaela van der Schaar. Influence-guided symbolic regression: Scientific discovery via LLM-driven equation search with granular feedback. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://arxiv.org/abs/2605.29184.

Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85, 2009. doi: 10.1126/science.1165893.

Parshin Shojaee, Kazem Meidani, Shashank Gupta, Amir Barati Farimani, and Chandan K. Reddy. LLM-SR: Scientific equation discovery via programming with large language models. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=m2nmp8P5in.

Parshin Shojaee, Ngoc-Hieu Nguyen, Kazem Meidani, Amir Barati Farimani, Khoa D. Doan, and Chandan K. Reddy. LLM-SRBench: A new benchmark for scientific equation discovery with large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 55325–55359. PMLR, 2025b. URL https://proceedings.mlr.press/v267/shojaee25a.html.

Jiarui Su, Songjun Tu, Bei Sun, and Xiaojun Liang. STRIDE: A self-reflective agent framework for reliable automatic equation discovery, 2026. URL https://arxiv.org/abs/2605.17790.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The shapley taylor interaction index. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9259–9268. PMLR, 2020. URL https://proceedings.mlr.press/v119/sundararajan20a.html.

Wassim Tenachi, Rodrigo Ibata, and Foivos I. Diakogiannis. Physical symbolic optimization, 2023. URL https://arxiv.org/abs/2312.03612.

Runxiang Wang, Boxiao Wang, Kai Li, Yifan Zhang, and Jian Cheng. DrSR: LLM-based scientific equation discovery with dual reasoning from data and experience, 2025. URL https://arxiv.org/abs/2506.04282.

Shijie Xia, Yuhan Sun, and Pengfei Liu. SR-Scientist: Scientific equation discovery with agentic AI. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=KBN6oUx5uL.

Zeyu Xia, Jun Zhu, and Dong Yan. FunctionEvolve: Structure-guided symbolic regression with LLMs, 2026b. URL https://arxiv.org/abs/2606.07704.

Zikai Xie, Wenmei Li, Man Luo, Jun Jiang, and Linjiang Chen. Language models guide symbolic equation discovery by controlling search, 2026. URL https://arxiv.org/abs/2607.04156.

Jianke Yang, Ohm Venkatachalam, Mohammad Kianezhad, Sharvaree Vadgama, and Rose Yu. Think like a scientist: Physics-guided LLM agent for equation discovery, 2026. URL https://arxiv.org/abs/2602.12259.

Wenxiao Zhao, Dong Liu, Kaiyi Xu, Feng Liu, Zhen Zhao, Fei Ben, Shu Wang, Wenhao Li, Ying Nian Wu, Fenghua Ling, Haobo Li, and Lei Bai. A-SR: Self-evolving agentic LLMs for symbolic regression via hierarchical coordination, 2026. URL https://arxiv.org/abs/2608.04872.

APPENDIX

A IMPLEMENTATION AND EVALUATION DETAILS

This appendix describes the data, evaluation, and search procedure. Supplementary results and resource use appear in Appendix B; structural recovery is examined in Appendix C. Appendix D collects the formal definitions and proofs.

A.1 Tasks, Metrics, and Noisy Observations

The frozen benchmark comprises 36 CRK, 24 BPG, 44 PO, and 25 MatSci tasks. The training split is used for structure search and parameter fitting, while test and OOD splits are reserved for final evaluation. Numerical error is

$$\mathrm{NMSE} = \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (y_i - \bar{y})^2}. \tag{4}$$

For each task, $\mathrm{Acc}_\tau^{95\%}$ is computed from the predefined pointwise relative-error metric: at least 95% of evaluation points must have error below τ. A non-finite prediction makes the task fail. Median NMSE is computed over finite predictions, while the task-level accuracy metrics retain these failures.

For each completed task, `run_manifest.json` records SHA-256 hashes of the input NPZ, source tree, search configuration, resource envelope, and results. These hashes identify the task and its associated artifacts. Ground-truth equations are reserved for structural evaluation and trajectory auditing after search.

Symbolic accuracy. We evaluate SA using the definition and procedure of LLM-SRBench (Shojaee et al., 2025b). The benchmark compares predicted and reference symbolic forms after removing parameters and constants. All methods use the same implementation, with ground truth introduced only after search. SA therefore concerns the symbolic form, while NMSE and threshold accuracy evaluate the returned expression with its fitted constants. Those fitted values remain fixed throughout test and OOD evaluation.

Noise protocol. For Section 4.5, only training labels are replaced by $y_i^{(\epsilon)} = y_i(1 + \epsilon z_i)$ with $z_i \sim \mathcal{N}(0, 1)$ and $\epsilon \in \{0.01, 0.03, 0.05\}$. Each condition uses the same 129 tasks and frozen search configuration; test and OOD arrays remain clean. The realization seed and the hashes of each materialized noisy dataset are stored with the run manifests. The perturbation has zero mean and variance $\epsilon^2 y_i^2$, so the noise level specifies relative variation in each training label.

A.2 Models and Search Configuration

Tables 4 and 5 summarize the LLM interfaces, search horizons, and local evaluation settings. Search uses the released configurations of LLM-SR (Shojaee et al., 2025a), SR-Scientist (Xia et al., 2026a), and IGSR (Saveliev et al., 2026). All methods are evaluated on the frozen tasks and splits described above.

Table 4: LLM calls and native search horizons. Released systems retain their original interaction semantics.

Method	LLM calls and search horizon
ATLAS-SR	DeepSeek-V4-Flash or `gpt-4o-mini-2024-07-18`; temperature 0; JSON output; at most 8,000 output tokens and 1,200 s per call; no application-level retries; four rounds with at most four structural-action calls per round.
IGSR	Official revision `44af1f48`; one search seed per task; 300k-token stopping cap; `keep_n_terms`=6; early stopping disabled; sampling parameters inherited from the official revision without wrapper overrides.
LLM-SR	Official `native-full`: 1,000 samples, four samples per prompt, and ten islands; temperature 0.8; at most 1,024 output tokens and 60 s per client call; upstream retries retained for API failures.
SR-Scientist	Official 40 rounds with at most 25 assistant actions per round; temperature 0.7, `top_p`=1; at most 8,192 output tokens and 1,200 s per call.

Configuration and model identifiers. Per-task manifests record hashes of the ATLAS-SR configuration and source tree. DeepSeek runs used the `deepseek-v4-flash` API identifier, and GPT runs used `gpt-4o-mini-2024-07-18`; provider identifiers and request metadata are retained in the run manifests.

Table 5: Local evaluation and selection configurations.

Method	Local evaluation and selection
ATLAS-SR	400 training points; beam width 8; at most 96 children per parent; 0.9-s initial-fit timeout; per-round caps of 22 higher-budget fits, 20 candidate scans, two matched-setting calibrations, and ten retained-candidate fitting grants; base symbolic seeds truncated to 32 in frozen order.
IGSR	Official pruning and fitting; selection by minimum validation MSE, with earlier history breaking ties; fallback ridge regression allowed; one worker per seed.
LLM-SR	One sampler and one evaluator; 30-s timeout per local evaluation; candidate, island-update, and selection semantics follow the released implementation.
SR-Scientist	Top three retained per round; at most ten parameters; SciPy BFGS initialized at all ones and evaluated by MSE; tool sandbox and official feedback loop unchanged.

A.3 Matched Refit

MR tests only deletion targets that can be materialized safely from registered structures. Deleting an additive member retains the other branches; deleting a multiplicative member rebuilds the product from the remaining factors. Quotient structures maintain distinct numerator and denominator roles, and the experiment is rejected if deletion empties a role or invalidates the expression. For a general graph cut without an explicit assembly type, the removed node is replaced only by a registered, refittable, type-compatible scalar baseline. A failed type, domain, or provenance-mapping check closes the request without storing a result.

Let L_{full} and L_{del} be the matched losses of the complete assembly and the post-deletion refit. The implementation records $d = \max(L_{\mathrm{del}}, \varepsilon)/\max(L_{\mathrm{full}}, \varepsilon)$ and $\Delta_{\mathrm{MR}} = \log_{10} d$. Fixed thresholds treat $d > 10$ as strong support for retaining the tested component and $d < 1.2$ as an approximately lossless deletion; the interval between them remains weak evidence. Only online MR results that pass closure, numerical validity, and protocol-matching checks update the atlas. Calibration and shadow executions enter the audit record without changing the current candidate set.

Scale and fitting conditions. The logarithm expresses the loss ratio in orders of magnitude. The thresholds guide retention under the recorded protocol; the floor assigns zero effect when both losses fall below ε. Matching covers optimizer settings, initialization and restart policies, stopping rules, and resource caps. Actual optimization trajectories and resource use can differ within these allowances. Finite-budget fitting can yield a negative MR value when the reduced assembly reaches a lower loss, for example through easier optimization. For nested models at their global training-loss minima, the MR value is nonnegative. Termination status and fitting diagnostics accompany each loss, and a changed fitting configuration receives a new protocol identity.

A.4 Scheduling and Fitting Priority

Requests for further candidate fitting use the score

$$P_{\mathrm{mat}}(q) = \frac{0.5[-\log_{10}\max\{\mathcal{L}(q), \varepsilon\}] + p_{\mathrm{under}}(q) + 0.5 s_{\mathrm{mat}}(q) + 0.25\{p_{\mathrm{step}}(q) + [g(q)]_+\}}{c(q)}. \tag{5}$$

Here $\mathcal{L}$ is the current training loss and $c(q)$ is the declared evaluation cost. The remaining terms are heuristic scores for prioritizing fitting. The term p_{under} summarizes underfitting diagnostics, and s_{mat} estimates the need for further fitting. The terms p_{step} and g describe potential progress and useful descendants. These scores serve as uncalibrated priorities for resource allocation.

Separate queues rank unresolved comparisons by deadline and coverage, retained candidates by position and stability, and possible numerical proxies by unresolved risk. Each queue obeys the fixed caps in Table 5. The shared ledger reserves resources before execution and records realized cost afterward. Deferred requests remain pending. These priorities affect training-side search, while the final output is chosen from completed fits.

A.5 Final Archive Selection

At termination, SelectBest uses the latest active, completed fit of each executable candidate in the archive. Profile-relaxation results remain available as diagnostics. Selection first identifies a predictive candidate, then checks whether an archive candidate qualifies to replace it.

Candidates meeting the configured predictive thresholds receive priority in the first stage. The ranking uses training NMSE, internal stress diagnostics, proxy risk, and simplified expression complexity to choose the default output.

Next, the selector considers the candidate preferred by the archive's structural checks. Replacement requires structural eligibility, stability, and a training loss within the configured admissible range. If any check fails, the default output is retained. Both stages use training-side records and select an existing fitted expression. Selection leaves its structure and coefficients unchanged. Test and OOD metrics and benchmark SA are evaluated after the choice is fixed.

B Supplementary Experiments and Resources

B.1 Complete Core-Design Ablation

Table 6 reports the outcomes of the extended ablations; Table 7 specifies their interventions. Random scheduling samples requests within fixed cost strata.

Table 6: Performance of the eleven frozen configurations. ID and OOD accuracy use $\mathrm{Acc}_{0.01}^{95\%}$; SA is task-level structural equivalence. All entries are percentages.

Configuration	ID $\mathrm{Acc}_{0.01}^{95\%}$	OOD $\mathrm{Acc}_{0.01}^{95\%}$	SA
ATLAS-SR	89.15	85.27	24.03
Flat candidate memory	65.89	64.34	4.65
No evidence feedback	76.74	73.64	15.50
Frozen deletion	74.42	66.67	14.73
Random scheduling	79.85	75.97	19.38
One-shot large dictionary	72.87	66.67	14.73
Independent round restart	70.54	64.34	22.48
Candidate-score broadcast	66.67	62.02	11.63
No LLM proposals	54.26	34.11	9.30
Passive atlas	64.34	62.02	13.95
No protocol indexing	61.24	60.47	13.18

The variants separate three uses of search history: retaining structures, describing their experimental support, and acting on that support. Flat candidate memory retains candidate-level history. Candidate-score broadcast also uses flat memory and replaces structured proposal feedback with candidate scores. The one-shot large dictionary supplies a collection at initialization; the full atlas accumulates components and experiments during search. The dictionary uses ATLAS-SR's registered structures as an internal ablation.

Independent round restart resets the search state, limits atlas access to the current round, and omits structured proposal feedback. It retains matched refit, protocol indexing, and adaptive scheduling, as specified in Table 7. The contrast jointly changes cross-round access and proposal feedback. SA remains close to the full configuration, while strict OOD accuracy falls from 85.27% to 64.34%. The main difference in this comparison is therefore the coverage of high-precision numerical predictions.

No evidence feedback removes experimental feedback from proposals while retaining atlas access for fitting and scheduling. Passive atlas removes atlas reads from all three decisions, although the experiments remain recorded. Their performance losses support using recorded evidence in subsequent decisions. Frozen deletion changes how component effects are measured, and no protocol indexing changes which measurements are pooled. Together, these variants examine evidence construction, storage, and use within the complete search procedure.

Table 7: Frozen ablation configurations. Panel A specifies the information visible during proposal; Panel B specifies the deletion-and-refit comparison and scheduling policy.

A. State access and feedback

Configuration	Memory	Atlas read	Proposal feedback
ATLAS-SR	Atlas	Full	Structured
Flat candidate memory	Flat	–	Structured
No evidence feedback	Atlas	Maturity/schedule	None
Frozen deletion	Atlas	Full	Structured
Random scheduling	Atlas	Proposal/fitting	Structured
One-shot large dictionary	One-shot	Full	Structured
Independent round restart	Restart	Current round	None
Candidate-score broadcast	Flat	Score only	Score only
No LLM proposals	Atlas	Full	LLM off
Passive atlas	Write-only	None	No atlas read
No protocol indexing	Atlas	Full	Structured

B. Counterfactual evidence and scheduling

Configuration	Deletion/refit evidence	Protocol	Scheduling
ATLAS-SR	Matched refit	Indexed	Adaptive
Flat candidate memory	Matched refit	Indexed	Adaptive
No evidence feedback	Matched refit	Indexed	Adaptive
Frozen deletion	Frozen delete	Indexed	Adaptive
Random scheduling	Matched refit	Indexed	Cost-random
One-shot large dictionary	Matched refit	Indexed	Adaptive
Independent round restart	Matched refit	Indexed	Adaptive
Candidate-score broadcast	Matched refit	Indexed	Adaptive
No LLM proposals	Matched refit	Indexed	Adaptive
Passive atlas	Matched refit	Indexed	Non-atlas
No protocol indexing	Matched refit	Collapsed	Adaptive

B.2 Task-Level ID/OOD Generalization

Figure 5 compares ID and OOD error for individual tasks. Points above the diagonal indicate that extrapolation remains harder even when the ID fit is accurate. The shift toward lower errors with ATLAS-SR complements the aggregate results in Table 1.

B.3 Resource Use and Accounting

Table 8 reports median resource use, and Figure 6 shows the per-task distributions. For ATLAS-SR, fewer LLM interactions accompany more local candidate evaluation and fitting.

Table 8: Median per-task resource use. DS-V4 denotes DeepSeek-V4-Flash. Candidate evaluations, CPU core-seconds, and wall time separate LLM use from local computation.

Method / interface	Calls	Tokens	Candidate evals.	CPU s	Wall s
LLM-SR / DS-V4	1,000	1,776,672	979	229.13	6,607.80
SR-Scientist / DS-V4	996	24,180,123	179.5	670.14	17,900.70
IGSR / DS-V4	163	305,645	145	556.81	278.39
IGSR / GPT-4o-mini	168	306,439	135	716.92	358.43
ATLAS-SR / DS-V4	14	165,682	802	1,833.74	1,707.97
ATLAS-SR / GPT-4o-mini	14	142,969	802	1,781.17	1,632.20

Measurement. Calls and tokens follow the released interface semantics: ATLAS-SR counts successful provider responses, with transport retries recorded separately; token totals use provider-

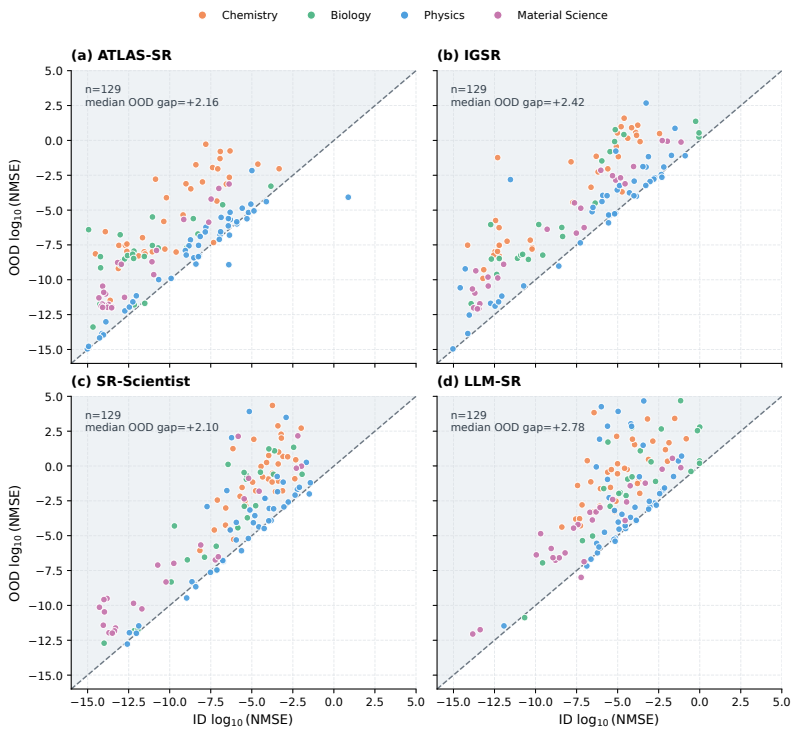

Figure 5: Task-level ID–OOD generalization with DeepSeek-V4-Flash. Each point is one task; the diagonal denotes equal ID and OOD NMSE.

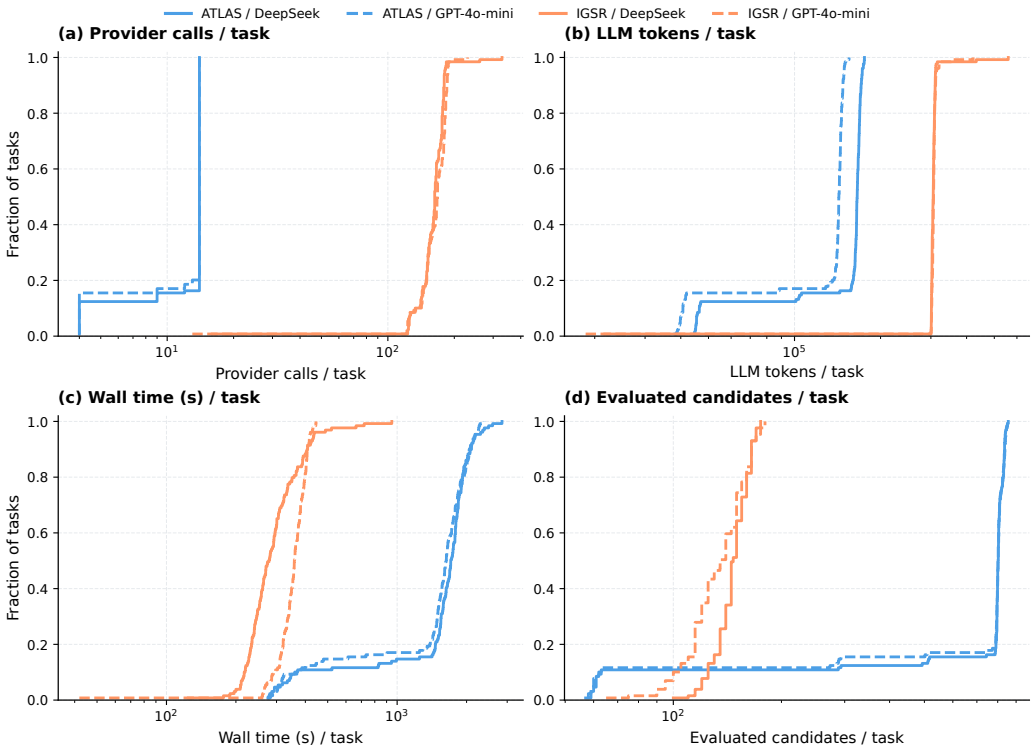

Figure 6: Empirical cumulative distributions of per-task resources for ATLAS-SR and IGSR with both LLM backbones: service calls, LLM tokens, wall time, and candidate evaluations.

reported input and output usage. Candidate evaluations count local evaluation events, whose fitting stage and deduplication rules follow each method's configuration. Initial screening, further fitting, calibration, and MR all enter ATLAS-SR's resource totals. An evaluation count measures how many local evaluations were performed; CPU core-seconds measure the computation they consumed. The cost of an evaluation varies with expression size, fitting stage, optimizer, and restart policy. Matching evaluation counts and matching CPU use therefore impose different resource constraints. Table 8 reports both quantities alongside wall time to describe this allocation.

CPU core-seconds sum user and system time for the process and its children. Wall time measures each task from start to completion, including API waiting, I/O, and scheduling. Reported compute covers local CPU execution; LLM inference is provided by a remote service.

Parallel execution. The 129 ATLAS-SR tasks run in five fixed shards. LLM-SR and SR-Scientist use at most five across-task workers. Within each task, LLM-SR uses one sampler and evaluator, SR-Scientist uses one model channel and tool channel, and IGSR uses one seed worker with two threads per numerical library. CPU and wall times are reported on the hardware used for each run.

Cost implications. ATLAS-SR shifts additional search effort toward local fitting and matched refits. When metered LLM inference dominates deployment cost, this allocation can make further exploration more economical. Its monetary benefit depends on model pricing and hardware; the measured token and CPU totals allow that trade-off to be assessed for a particular deployment.

C Structural Recovery Analysis and Cases

C.1 Stage Audit of 129 Tasks

The 91,803 candidate events, 42,111 LLM-visible records, 1,624 plans, and 4,128 final-ranking records of ATLAS-SR are indexed by task. Stage labels combine native logs with the final ground-truth structure:

- **Primitive exposed:** an operator or basic motif required by the ground truth appears in at least one candidate;
- **Relation materialized:** parameters, carriers, variable roles, and branch membership form the target relation in one candidate;
- **Jointly fitted:** the target relation completes a higher-budget fit and becomes competitive;
- **Retained:** the correct structure enters the final or explicitly retained candidate set;
- **Finally presented:** the output candidate attains $\mathrm{SA} = 1$ under the LLM-SRBench procedure.

The controller's search outputs are priorities and budgets for existing candidate IDs and registered action types. Graph edits construct the executable equations. An LLM-visible record identifies structural evidence available to the controller. Baselines differ in archive granularity. IGSR retains final support and the selected history position. SR-Scientist retains complete candidate trajectories for only a subset of tasks. LLM-SR retains the final checkpoint and score history. Cross-method analysis therefore uses evidence levels consistent with the available log granularity.

C.2 Complete Stratification by Relational Combination Burden

RCB groups tasks by the combinations required in their generating equations. It counts four demands: parameterized transforms, multiplication of transformed terms by external factors, variables combined within a non-additive subexpression, and at least three independent additive branches. Table 9 reports recovery in each group for all methods and both backbones.

The structural-recovery advantage of ATLAS-SR is concentrated in the RCB 0–1 groups, which contain 35 tasks. Across the remaining 94 tasks, IGSR recovers slightly more equations with both backbones. The aggregate gain therefore combines substantial improvements in lower-burden groups with limited recovery in higher-burden groups. RCB describes the combinations required by

Table 9: SA by RCB. Entries are recovered tasks (percentage); headings give group sizes. DS and GPT denote DeepSeek-V4-Flash and GPT-4o-mini.

Method	LLM	0 ($n = 10$)	1 ($n = 25$)	2 ($n = 29$)	3 ($n = 38$)	4 ($n = 27$)	All ($n = 129$)
ATLAS-SR	DS	9 (90.0)	15 (60.0)	5 (17.2)	1 (2.6)	1 (3.7)	31 (24.0)
ATLAS-SR	GPT	9 (90.0)	14 (56.0)	5 (17.2)	2 (5.3)	1 (3.7)	31 (24.0)
IGSR	DS	1 (10.0)	4 (16.0)	7 (24.1)	2 (5.3)	0 (0.0)	14 (10.9)
IGSR	GPT	6 (60.0)	5 (20.0)	7 (24.1)	3 (7.9)	0 (0.0)	21 (16.3)
LLM-SR	DS	2 (20.0)	1 (4.0)	1 (3.4)	0 (0.0)	0 (0.0)	4 (3.1)
LLM-SR	GPT	2 (20.0)	1 (4.0)	2 (6.9)	0 (0.0)	0 (0.0)	5 (3.9)
SR-Scientist	DS	1 (10.0)	2 (8.0)	1 (3.4)	2 (5.3)	1 (3.7)	7 (5.4)
SR-Scientist	GPT	1 (10.0)	0 (0.0)	1 (3.4)	1 (2.6)	0 (0.0)	3 (2.3)
LaSR	DS	1 (10.0)	0 (0.0)	1 (3.4)	0 (0.0)	0 (0.0)	2 (1.6)
LaSR	GPT	1 (10.0)	1 (4.0)	2 (6.9)	0 (0.0)	0 (0.0)	4 (3.1)
Direct prompting	DS	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)
Direct prompting	GPT	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)
PySR	–	2 (20.0)	2 (8.0)	2 (6.9)	1 (2.6)	0 (0.0)	7 (5.4)
PhySO	–	0 (0.0)	0 (0.0)	0 (0.0)	1 (2.6)	0 (0.0)	1 (0.8)

an equation; the OS analysis below examines how clearly the sampled data distinguish competing structures.

C.3 Search Trajectories and Low-Error Alternatives

Shared failure patterns. The MatSci18 target contains $\epsilon^p + \epsilon(T - T_0)^q$. IGSR/DeepSeek-V4-Flash reaches 4.04×10^{-14} with a high-order ϵ–T polynomial. LLM-SR/DeepSeek-V4-Flash reaches 4.22×10^{-14} with a skeleton coupling linear, quadratic, and cubic strain terms to temperature. SR-Scientist/DeepSeek-V4-Flash reaches 2.00×10^{-14} with a strain polynomial that omits T. All three GPT-4o-mini variants likewise fall below 10^{-10} while attaining $\mathrm{SA} = 0$. Distinct search methods can therefore reach different, structurally incorrect approximations on the same finite input manifold.

The MatSci2 target branch is $(T - T_0)e^{-\epsilon}$. LLM-SR recognizes temperature softening, thermoelasticity, and exponential effects. Its candidates apply the exponential to the temperature difference or to an independent strain-hardening term. SR-Scientist/DeepSeek-V4-Flash reaches 5.24×10^{-15} with a Chebyshev expansion that depends only on ϵ. The final IGSR/DeepSeek-V4-Flash support contains both $Te^{-\epsilon}$ and $e^{-\epsilon}$ while retaining polynomial branches that cannot be removed. These cases illustrate three failures: attaching the exponential to the wrong term, using strain alone as a proxy, and retaining extra polynomial branches.

The PO25 target is $F_0 \sin t - \beta \sin v$. The DeepSeek results of IGSR, LLM-SR, and SR-Scientist reach 5.61×10^{-13}, 1.19×10^{-12}, and 1.30×10^{-12}, respectively. All three replace $\sin v$ with $v, v^3, v^5, \ldots$. Periodic forcing and nonlinear damping are identified, while the exact velocity operator remains covered by a local polynomial OS.

PO3: assembling the required terms. Table 10 aligns the available search traces with their final structural outcomes.

PO3 traces physical interpretation and equation assembly. LLM-SR and SR-Scientist propose explanations and candidates consistent with a nonlinear oscillator. ATLAS-SR combines the required terms through graph edits, and IGSR retains all three in its final support. For IGSR, the $\mathrm{SA} = 1$ judgment allows additional weighted terms to be removed through admissible constant assignments. Numerical evaluation uses the coefficients returned by fitting, preserving the distinction between symbolic form and fitted predictions.

MatSci2: retaining a low-error alternative. The MatSci2 target is $7.317\epsilon^3 + 7.950(T - 287.092)e^{-\epsilon}$. A deterministic fallback first produces the rational candidate `cand_20`; a low-budget

Table 10: Cross-method search trajectories for PO3. The ground-truth structure is used only for post-search SA evaluation.

Method	Auditable trajectory	Final structural interpretation	SA
ATLAS-SR	affine → trig-unary → parent-preserving branch → local-replace; the final candidate was not named in an LLM review or plan	Retains v, $\sin v$, and $\sin t$ together; graph edits complete the exact assembly	1
IGSR	History position 106 selected; final support contains six terms	All three required terms are present; additional weighted terms are removable through admissible constant assignments	1
SR-Scientist	Best at turn 1: $\sin t + \cos t + v$; best at turn 3: $\sin t + v + \cos 3t$; 47 candidates in total	21 candidates contain $\sin t$ and six contain v^3; none contains $\sin v$	0
LLM-SR	209 valid evaluations and seven incumbent improvements; text identifies a nonlinear oscillator, damping, and periodic forcing	Final expression $v + v^3 + \sin(\omega t) + \cos(\omega t) + vt$ replaces $\sin v$ with a polynomial OS	0

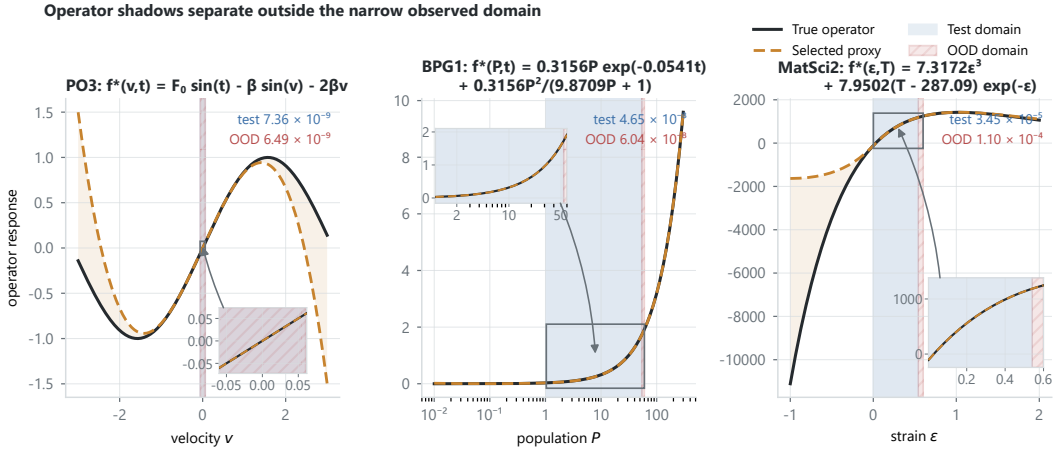

Figure 7: In-domain overlap and far-range separation for three operator shadows. The black curve is the true operator or branch omitted by the candidate; the dashed orange curve is its proxy projection fitted on the frozen training set. Blue and red shading mark test and OOD coordinate ranges, and insets magnify differences over the evaluated domain. Panels show $\sin v$ and its $[1, v, v^3]$ projection for PO3 (left), the rational branch and final proxy static basis for BPG1 (middle), and temperature–strain coupling on the selected variable representation for MatSci2 (right).

proposal then generates `cand_99` under an alternative variable representation. Its early NMSE is 3.37×10^{-10}. A higher-budget refit reduces training NMSE to 4.69×10^{-15}, and test NMSE is 5.50×10^{-15}. After this low error enters the atlas, the LLM retains the candidate, allocates more fitting budget to it, and targets it in four later plans. The final expression omits the $Te^{-\epsilon}$ branch and receives 0 under the official SA definition. Its low error encourages repeated selection of a structurally incorrect candidate. The right panel of Figure 7 shows why this proxy remains competitive on the frozen data manifold.

C.4 Projection-Based Quantification of Operator Shadows

For a required ground-truth operator g omitted by the candidate and a basis matrix Φ supplied by the incorrect candidate, define

$$\mathrm{Shadow}_D(g \mid \Phi) = 1 - \frac{\min_\beta \|g_D - \Phi_D \beta\|_2^2}{\|g_D - \bar{g}_D\|_2^2}, \qquad \rho_D = \frac{\min_\beta \|g_D - \Phi_D \beta\|_2}{\|g_D\|_2}. \tag{6}$$

Here g_D contains the operator values on D, and $\bar{g}_D = |D|^{-1} \sum_{i=1}^{|D|} (g_D)_i$ is their empirical mean. Subtracting $\bar{g}_D$ centers each entry of g_D by this mean.

The formulas above Figure 7 provide the complete task context; the plotted vertical quantity isolates only the missing component directly responsible for the structural error. The left panel compares $g(v) = \sin v$ with $\hat{g}(v) = \beta_0 + \beta_1 v + \beta_3 v^3$. Coefficients are fitted only on the training set and then evaluated unchanged on test and OOD, yielding $\rho_{\text{test}} = 7.36 \times 10^{-9}$ and $\rho_{\text{OOD}} = 6.49 \times 10^{-9}$. All three splits satisfy $|v| \leq 0.06175$ and therefore remain within the local regime $\sin v = v - v^3/6 + O(v^5)$.

The middle panel isolates BPG1's missing rational branch $0.3156P^2/(9.8709P + 1)$. The candidate already recovers the exponential target branch. The proxy static basis is $[1, P, P^2, |P|^{-0.9184}]$, giving $\rho_{\text{test}} = 4.65 \times 10^{-8}$ and $\rho_{\text{OOD}} = 6.04 \times 10^{-8}$. The right panel shows the MatSci2 branch $7.9502(T - 287.092)e^{-\epsilon}$. Frozen samples lie on the one-dimensional relation $T = 273 + 500\epsilon$, and the proxy basis is

$$\Big[1,\ \epsilon e^{-0.9959|\epsilon|},\ \epsilon/d(T),\ 1/d(T)\Big], \qquad d(T) = |0.02334T - 50.0067| + 1.$$

with $\rho_{\text{test}} = 3.45 \times 10^{-5}$ and $\rho_{\text{OOD}} = 1.10 \times 10^{-4}$. Across all three cases, OOD data become structurally discriminative only after leaving the local or asymptotic regime shared by the true and proxy operators. Merely adding samples along the same low-dimensional manifold can preserve an incorrect structure with extremely low error.

D Mathematical Formalization and Proofs

We first specify the candidate representation and atlas update, then derive the properties used to motivate matched comparison and contextual evidence. Implementation settings are collected in Appendix A.

D.1 Formalization and Algorithm

D.1.1 Symbolic Regression, Candidates, and Protocols

Let $\mathcal{F}$ be the space of symbolic expressions constructible from the given variables and operator library. The basic symbolic-regression objective is

$$f^\star \in \arg\min_{f \in \mathcal{F}} \mathcal{L}_{\mathcal{D}^{\text{tr}}}(f), \tag{7}$$

while requiring the resulting expression to remain concise and generalize to unseen inputs. A candidate is specified by a discrete structure C and continuous parameters $\boldsymbol{\theta} \in \Theta_C$, denoted $f_C(\mathbf{x}; \boldsymbol{\theta})$. Given protocol π, parameter fitting is defined as

$$\widehat{\boldsymbol{\theta}}_\pi(C) \in \arg\min_{\boldsymbol{\theta} \in \Theta_C} \mathcal{L}^{\alpha}_{\mathcal{D}^{\text{fit}}}\big(f_C(\cdot; \boldsymbol{\theta})\big). \tag{8}$$

A single candidate executes as an expression tree or DAG. An assembly relation reused across candidates is written as

$$(u_1^{r_1}, \ldots, u_k^{r_k}) \xrightarrow{\tau} u, \tag{9}$$

where u_i is an input component, r_i its input-port role, and τ the assembly operation. The protocol key and minimal evidence record are

$$\pi = \big(\mathcal{D}^{\text{fit}}, \alpha, \chi, \phi, \mathbf{b}\big), \qquad e = \big(u, S, \pi, \Delta, \rho\big), \tag{10}$$

Here α specifies the coordinate representation and readback to the original space. The remaining fields identify the optimizer class and frozen configuration (χ), fitting stage (ϕ), and resource cap ($\mathbf{b}$). S is the assembly containing u, Δ the experimental difference, and ρ stores validation status, diagnostics, and realized cost. The same structure retains separate records under different assemblies or protocols, and only protocol-compatible records are combined.

D.1.2 Complete Atlas State

The main text describes the Structural Atlas compactly. Its complete implementation state is

$$\mathfrak{A}_t = \big(\mathcal{A}_t, \mathcal{H}_t, \mathcal{Q}_t, \mathcal{Z}_t, \mathcal{E}_t, \mathcal{K}_t, \mathcal{C}_t, \mathcal{R}_t\big), \tag{11}$$

Here $\mathcal{A}_t$ stores variable representations, and $\mathcal{H}_t$ stores the cross-candidate graph. The registry $\mathcal{Q}_t$ stores proven algebraic relations and unresolved comparisons; $\mathcal{Z}_t$ stores parameter-fitting states. Completed experiments enter $\mathcal{E}_t$, while unfinished evaluation requests remain in $\mathcal{K}_t$. The registry $\mathcal{C}_t$ stores recorded conclusions and their supporting evidence; $\mathcal{R}_t$ is the shared resource ledger. The hypergraph stores object identities and relations, while mutable payloads are maintained by their corresponding registries.

Provenance and completed outcomes remain immutable. Candidates and unresolved comparisons can be activated, deactivated, or reopened as evidence arrives. LLM proposals alter search priority, completed fits add empirical records, and matched refits measure the contribution of a tested component.

D.1.3 Candidate Representation and Cross-Candidate Graph

A variable representation is

$$\alpha = (D_\alpha, T_\alpha, R_\alpha, g_\alpha, \xi_\alpha), \tag{12}$$

Here D_α is the applicable domain, T_α the data transform, and R_α the readback of predictions to the original response space. The field g_α stores input validity conditions and response-side diagnostics. The field ξ_α stores transformation parameters fitted on training data and then frozen. After candidate C predicts under representation α, comparisons use $\widehat{\mathbf{y}}^\alpha$ mapped back to the original space. The original-space loss is

$$\mathcal{L}^\alpha_{\mathcal{D}}(C, \boldsymbol{\theta}) = \frac{1}{|\mathcal{I}_\alpha|} \sum_{i \in \mathcal{I}_\alpha} \ell(y_i, \widehat{y}^\alpha_i)\,. \tag{13}$$

Valid evaluation indices are determined by the inputs and frozen state. Irreversible or approximate coordinate transforms register an explicit readback rule and approximation error.

Each candidate is represented internally by a finite DAG:

$$C = (V_C, E_C, r_C, \Theta_C), \tag{14}$$

where V_C contains variable, constant, parameter, and operator nodes; E_C describes data dependencies; and r_C is the output root. Nodes store input ports, types or dimensions, domain conditions, parameter domains, and provenance. Expression strings and trees are serialized views of this DAG.

A heterogeneous directed hypergraph $\mathcal{H}_t = (\mathcal{V}_t, \mathcal{B}_t)$ records shared structures and assembly relations across candidates, together with experiments and recorded conclusions. A general hyperedge is

$$b = (\mathbf{I}_b, \mathbf{O}_b, \tau_b, \gamma_b, \nu_b), \tag{15}$$

Here $\mathbf{I}_b$ and $\mathbf{O}_b$ are ordered input and output ports with labeled roles, and τ_b is the relation type. The field γ_b stores guards such as parameter sharing, domains, or comparison baselines; ν_b stores immutable provenance.

D.1.4 Generating Candidates from the Atlas

The generator performs type-preserving actions including composition, wrapping, replacement, deletion, recombination, and valid transfer across variable representations. For action a, $\mathbf{I}$ specifies the selected input components and their ordered port roles. Let $\varrho_0(a, \mathbf{I}) > 0$ be a base exploration distribution covering legal actions, s_t^{LLM} the LLM proposal score, and s_t^{emp} the empirical potential formed by historical experiments. Then

$$\begin{aligned} \widetilde{\varrho}_t(a, \mathbf{I}) &\propto \varrho_0(a, \mathbf{I}) \exp\left[\lambda_t s_t^{\mathrm{LLM}}(a, \mathbf{I}) + \beta_t s_t^{\mathrm{emp}}(a, \mathbf{I})\right], \\ \varrho_t(a, \mathbf{I}) &= (1 - \varepsilon_t)\widetilde{\varrho}_t(a, \mathbf{I}) + \varepsilon_t \varrho_0(a, \mathbf{I}), \qquad 0 < \varepsilon_t \le 1. \end{aligned} \tag{16}$$

The empirical score combines results obtained under compatible protocols, evidence for individual components, unresolved comparisons, and numerical risk. Unexplored legal structures remain available; finite resources determine when they are generated.

D.1.5 Evaluation Requests, Scheduling, and Fitting

Evaluation requests. Each evaluation request q records a protocol π_q, its provenance, the comparison baseline, and the unresolved question that motivated it:

$$\begin{aligned} \pi_q &= \big(d_q, \alpha_q, \chi_q, \phi_q, \mathbf{b}_q\big), \\ q &= \big(m_q, s_q, a_q, \pi_q, e_q^0, o_q, \omega_q\big). \end{aligned} \tag{17}$$

Here d_q registers the data split and role, and α_q is the variable representation. The field χ_q specifies the optimizer class, initialization, restart policy, and frozen configuration. The fitting stage is ϕ_q, and the resource cap is $\mathbf{b}_q$. The request identifies the candidate or assembly under evaluation (m_q), its provenance (s_q), generating action (a_q), and comparison baseline (e_q^0). The field o_q links the unresolved comparison, and ω_q stores audit fields. Canonicalized fields form a unique signature. A change to the protocol, comparison baseline, or random replicate creates a new request identity.

Requests for further candidate fitting use the frozen priority in Equation (5). Other queues cover unresolved comparisons, promising candidates, and numerical risks; they are ranked by deadline, coverage, candidate stability, and unresolved risk. The queues share one resource ledger under frozen quotas. Appendix A.4 gives the implemented scores and quotas. Matched outcomes from completed experiments update the evidence stored with the tested structure.

Budget accounting. For every resource type, the shared ledger maintains

$$\mathbf{B}^{\mathrm{tot}} = \mathbf{B}_t^{\mathrm{spent}} + \mathbf{B}_t^{\mathrm{reserved}} + \mathbf{B}_t^{\mathrm{available}}. \tag{18}$$

An admitted request first moves resources from available to reserved. After execution, realized usage moves to spent, and any unused portion returns to available. Deferral, timeout, and insufficient resources update the service state while retaining complete diagnostics. The round-wise admission quota B_t uses the same declared-cost units as $c(q)$ and is constrained by available resources after candidate fitting.

Checking completed fits. Given candidate C and protocol key π_q, the finite-budget optimizer returns

$$(\widehat{\boldsymbol{\theta}}_q, \mathcal{T}_q, \zeta_q, \rho_q) = \mathsf{Fit}(C; \pi_q)\,, \tag{19}$$

where $\mathcal{T}_q$ is the optimization trajectory, ζ_q the termination and diagnostic state, and ρ_q the realized resource cost. Initial screening, continuation, higher-budget fitting, parameter profiling, and failure recovery use distinct protocol identities. Variable projection is used for separable linear–nonlinear structures; ill-conditioned or multimodal structures combine bounded global exploration, multiple starts, and local refinement.

Updating the atlas requires a complete record of provenance, protocol, fit, diagnostics, cost, and termination status, together with valid data roles and numerical diagnostics. Outcomes used to compute a difference must also have compatible representations, data, baselines, fitting stages, optimizer classes, and budgets. The atlas retains support, refutation, and diagnostic failures. Valid outcomes enter statistical estimation, matched outcomes enter MR, and unobserved quantities remain NA.

D.1.6 Matched-Refit Evidence and Structural Sharing

Evidence from deletion comparisons. Fitting and ranking utilities under protocol π are defined as

$$\begin{aligned} U_\pi^{\mathrm{fit}}(S) &= -\log_{10} \max\{\mathcal{L}_\pi(\widehat{f}_S), \varepsilon\}, \\ U_\pi^{\mathrm{rank}}(S) &= U_\pi^{\mathrm{fit}}(S) - \lambda\Omega(S) - \mu\,\mathrm{Risk}_\pi(S), \end{aligned} \qquad \lambda, \mu \geq 0, \tag{20}$$

where $\mathcal{L}_\pi$, Ω, and Risk_π measure training-side fitting error, structural complexity, and numerical risk, respectively. MR measures the change in U^{fit} after deletion and refitting. Complexity and risk enter candidate ranking and scheduling separately.

If source set B has a measured baseline under a matched protocol, the observational gain of assembled set S is

$$\Gamma_{\mathrm{obs}}^{\pi}(S; B) = U_\pi^{\mathrm{rank}}(S) - U_\pi^{\mathrm{rank}}(B). \tag{21}$$

The result for a complete candidate is attached first to the assembly hyperedge that was actually evaluated. For a deletion group $G \subseteq S$ evaluated under one matched protocol π, the measurements decompose as

$$\begin{aligned}\Gamma_{\mathrm{fit}}(G \mid S) &= \log_{10} \frac{\max\{\mathcal{L}_\pi(\widehat{f}^{\mathrm{refit}}_{S\setminus G}), \varepsilon\}}{\max\{\mathcal{L}_\pi(\widehat{f}_S), \varepsilon\}}, \\ \Gamma_{\mathrm{cpx}}(G \mid S) &= \Omega(S) - \Omega(S \setminus G), \\ \Gamma_{\mathrm{risk}}(G \mid S) &= \mathrm{Risk}(S \setminus G) - \mathrm{Risk}(S), \\ \Gamma_{\mathrm{util}}(G \mid S) &= \Gamma_{\mathrm{fit}} - \lambda\Gamma_{\mathrm{cpx}} + \mu\Gamma_{\mathrm{risk}}.\end{aligned} \tag{22}$$

For a single component, $\Gamma_{\mathrm{fit}}(\{u\} \mid S) = \Delta^{\pi}_{\mathrm{MR}}(u \mid S)$. The protocol index is suppressed in the credit vector. These quantities are written to the atlas when both the complete set and deletion outcome are closed, valid, and matched.

Second-order conditional interaction is defined as

$$\mathcal{I}^{(q)}(u, v \mid S) = \Gamma_q(\{u\} \mid S) + \Gamma_q(\{v\} \mid S) - \Gamma_q(\{u, v\} \mid S), \tag{23}$$

where $q \in \{\mathrm{fit}, \mathrm{util}\}$. Its evaluation requires the full assembly, both single deletions, and the joint deletion under compatible protocols. This difference measures how one component's deletion effect depends on the presence of the other. Higher-order interactions may use analogous set differences or the Shapley–Taylor extension (Sundararajan et al., 2020). The implementation materializes only the deletion comparisons selected by the scheduler.

Sharing and retaining structures. Exactly identical subgraphs in candidate DAGs are shared directly. More general algebraic sharing requires two candidates to represent the same function family on declared domain D:

$$C_1 \equiv_D C_2 \iff \{f_{C_1,\theta}|_D : \theta \in \Theta_{C_1}\} = \{f_{C_2,\theta}|_D : \theta \in \Theta_{C_2}\}. \tag{24}$$

Candidates share fitting state when bidirectional parameter maps, parameter constraints, and coordinate readback are compatible. Approximation on finitely sampled points creates only an observational competition and retains separate parameters and provenance.

In BPG12, expanding the product and freeing six coefficients enlarges the function family (Equation (3)). The resulting candidate therefore receives a separate fit record from its source product.

Removing a structure from active search is reversible. It becomes inactive only after enough compatible experiments, no pending comparison that requires retaining it, and several rounds without positive evidence or valuable descendants. New evidence can reactivate it.

Interpreting evidence. Different sources support different conclusions. Low error supports predictive performance, algebraic proof supports equivalence, MR measures whether a component matters in a tested assembly, and LLM output supplies a proposal prior. Unresolved comparisons remain recorded when the budget ends. Benchmark ground truth is used only for post-search structural-recovery evaluation.

D.1.7 State Initialization and Evolution

Initialization takes training data $\mathcal{D}^{\mathrm{tr}}$, task context $\mathcal{T}$, initial structural library $\mathcal{S}_0$, and total resources $\mathbf{B}^{\mathrm{tot}}$. The operator registers computational coordinates, seed components, provenance records, and the resource ledger:

$$\mathfrak{A}_0 = \mathsf{Initialize}\big(\mathcal{D}^{\mathrm{tr}}, \mathcal{T}, \mathcal{S}_0, \mathbf{B}^{\mathrm{tot}}\big). \tag{25}$$

The initialized state contains the eight registries in Equation (11). Let ξ_t denote the random state required for LLM sampling and base exploration at round t. One round of evolution is

$$\begin{aligned}
\mathcal{M}_t &= \mathsf{Assemble}\big(\mathcal{D}^{\mathrm{tr}}, \mathcal{T}, \mathsf{Read}(\mathfrak{A}_t); \xi_t\big),\\
\mathcal{F}_t &= \big\{\big(m, \pi_m, \mathsf{Fit}(m; \pi_m)\big) : m \in \mathcal{M}_t\big\},\\
\mathcal{R}_t^{\mathrm{fit}} &= \mathsf{SettleFit}\big(\mathcal{R}_t, \mathcal{F}_t\big),\\
\mathcal{W}_t^{\mathrm{req}} &= \mathsf{Identify}\big(\mathfrak{A}_t, \mathcal{F}_t\big),\\
\mathcal{W}_t &= \mathsf{Schedule}\big(\mathcal{W}_t^{\mathrm{req}}, \mathcal{R}_t^{\mathrm{fit}}, B_t\big),\\
\mathcal{O}_t &= \big\{\mathsf{MR}(q; \pi_q, \mathcal{F}_t) : q \in \mathcal{W}_t\big\},\\
\mathcal{R}_t^{+} &= \mathsf{SettleMR}\big(\mathcal{R}_t^{\mathrm{fit}}, \mathcal{O}_t\big),\\
\mathcal{J}_t &= \mathsf{ValidateMatch}\big(\mathfrak{A}_t, \mathcal{F}_t, \mathcal{O}_t\big),\\
\mathfrak{A}_{t+1} &= \mathsf{Commit}\big(\mathfrak{A}_t, \mathcal{M}_t, \mathcal{F}_t, \mathcal{J}_t, \mathcal{R}_t^{+}\big).
\end{aligned} \tag{26}$$

where Commit updates each part of the state:

$$\begin{aligned}
\mathcal{A}_{t+1} &= \mathsf{RegisterChart}(\mathcal{A}_t, \mathcal{M}_t, \mathcal{F}_t),\\
\mathcal{H}_{t+1} &= \mathsf{MergeGraph}\big(\mathcal{H}_t, \mathsf{Struct}(\mathcal{M}_t), \mathsf{Credit}(\mathcal{J}_t)\big),\\
\mathcal{Q}_{t+1} &= \mathsf{UpdateRelations}(\mathcal{Q}_t, \mathcal{J}_t),\\
\mathcal{Z}_{t+1} &= \mathsf{UpdateFits}(\mathcal{Z}_t, \mathcal{F}_t),\\
\mathcal{E}_{t+1} &= \mathcal{E}_t \cup \mathsf{Closed}(\mathcal{F}_t, \mathcal{J}_t),\\
\mathcal{K}_{t+1} &= \mathsf{Pending}\big(\mathcal{K}_t, \mathcal{W}_t^{\mathrm{req}}, \mathcal{W}_t, \mathcal{J}_t\big),\\
\mathcal{C}_{t+1} &= \mathsf{UpdateClaims}(\mathcal{C}_t, \mathcal{J}_t),\\
\mathcal{R}_{t+1} &= \mathcal{R}_t^{+}.
\end{aligned} \tag{27}$$

Struct extracts canonical DAG nodes and assembly edges with labeled roles from new candidates. Credit reads only validated, protocol-matched outcomes. Pending retains requests awaiting scheduling, completion, or successful validation and adds newly identified comparisons. Relations, candidates, and recorded conclusions may change active status; provenance and completed outcomes remain unchanged.

Here $\mathcal{M}_t$ is the candidate set after legality checks and deduplication. The set $\mathcal{F}_t$ stores candidates and their protocol-indexed fitting outcomes. Evaluation requests generated from current candidates and unresolved comparisons enter $\mathcal{W}_t^{\mathrm{req}}$, with $\mathcal{W}_t$ denoting the scheduled subset. The sets $\mathcal{O}_t$ and $\mathcal{J}_t$ contain raw MR outputs and validated, protocol-matched outcomes, respectively. The ledgers after candidate fitting and MR settlement are $\mathcal{R}_t^{\mathrm{fit}}$ and $\mathcal{R}_t^{+}$, respectively. Commit writes new structures, completed experiments, MR results, and unresolved comparisons to the atlas. Every search operator in Equation (26) reads training data only; test data, OOD data, and benchmark ground truth are outside its input domain.

D.1.8 State and Budget Invariants

Atlas evolution preserves the following invariants:

$$\begin{aligned}
&\mathcal{E}_t \subseteq \mathcal{E}_{t+1},\\
&e \in \mathcal{E}_{t+1} \setminus \mathcal{E}_t \Longrightarrow \mathsf{Closed}(e) \wedge \mathsf{Valid}(e),\\
&\Delta(e) \neq \mathrm{NA} \Longrightarrow \mathsf{Match}_{\pi}\big(e^{\mathrm{full}}, e^{\mathrm{del}}\big),\\
&\sum_{q \in \mathcal{W}_t} c(q) \leq B_t.
\end{aligned} \tag{28}$$

The first line makes closed experimental records monotone. Structural activity may change with new evidence, while original provenance and existing outcomes remain immutable. The second and third lines restrict evidence writing to closed, valid, protocol-matched executions. The final line gives the round-wise admission constraint; reservation and realized usage before and after execution also satisfy Equation (18). LLM output enters only $\mathcal{M}_t$ and request priority. New MR evidence can be produced only from $\mathcal{J}_t$ through Commit.

D.1.9 COMPLETE PSEUDOCODE

Algorithm 2 summarizes the search and evidence-update cycle.

Algorithm 2 Evidence-guided atlas update in ATLAS-SR

Require: Training data $\mathcal{D}^{\mathrm{tr}}$, task context $\mathcal{T}$, initial structural library $\mathcal{S}_0$, total budget $\mathbf{B}^{\mathrm{tot}}$
Ensure: Best candidate and final atlas

$\mathfrak{A}_0 \leftarrow \textsc{Initialize}(\mathcal{D}^{\mathrm{tr}}, \mathcal{T}, \mathcal{S}_0, \mathbf{B}^{\mathrm{tot}})$
$t \leftarrow 0$
while the budget remains and a candidate or experiment can still be proposed **do**
  $\mathcal{M}_t \leftarrow \textsc{RetrieveAndAssemble}(\mathcal{D}^{\mathrm{tr}}, \mathcal{T}, \mathfrak{A}_t)$
  $(\mathcal{F}_t, \mathcal{R}_t^{\mathrm{fit}}) \leftarrow \textsc{RegisterFitAndSettle}(\mathcal{M}_t, \mathfrak{A}_t)$
  $\mathcal{W}_t^{\mathrm{req}} \leftarrow \textsc{IdentifyRequests}(\mathfrak{A}_t, \mathcal{F}_t)$
  $\mathcal{W}_t \leftarrow \textsc{Schedule}(\mathcal{W}_t^{\mathrm{req}}, \mathcal{R}_t^{\mathrm{fit}}, B_t)$
  $\mathcal{O}_t \leftarrow \varnothing$
  $\mathcal{R}_t^{+} \leftarrow \mathcal{R}_t^{\mathrm{fit}}$
  for all $q \in \mathcal{W}_t$ **do**
    $o_q \leftarrow \textsc{MatchedRefit}(q, \mathcal{F}_t)$
    $\mathcal{O}_t \leftarrow \mathcal{O}_t \cup \{o_q\}$
    $\mathcal{R}_t^{+} \leftarrow \textsc{SettleMR}(\mathcal{R}_t^{+}, q, o_q)$
  end for
  $\mathcal{J}_t \leftarrow \textsc{ValidateMatchAndCredit}(\mathfrak{A}_t, \mathcal{F}_t, \mathcal{O}_t)$
  $\mathfrak{A}_{t+1} \leftarrow \textsc{Commit}(\mathfrak{A}_t, \mathcal{M}_t, \mathcal{F}_t, \mathcal{J}_t, \mathcal{R}_t^{+})$
  $t \leftarrow t + 1$
end while
return $\textsc{SelectBest}(\mathfrak{A}_t), \mathfrak{A}_t$

SELECTBEST consists solely of ranking and choosing among completed fits in the archive, as described in Appendix A.5.

D.2 PROPERTIES OF STRUCTURAL STATE AND MATCHED COMPARISON

D.2.1 INSUFFICIENCY OF CANDIDATE–SCORE STATE

Let

$$z_t = \{(C_i, s_i)\}_{i=1}^{m_t} \tag{29}$$

denote the candidates retained at round t and their scores.

Proposition 1 (Candidate–score state is insufficient)**.** *There exist two search histories with the same z_t at round t and different accumulated local structural evidence. Their optimal orders of the next structural actions are reversed. Therefore, a policy that reads only z_t cannot be optimal for both histories.*

Proof of Proposition 1. Construct two histories h and h'. At round t, they have exactly the same candidate set $\{C_1, \ldots, C_m\}$ and scores $\{s_1, \ldots, s_m\}$, and hence the same z_t. Suppose the next round has only two executable actions, a_u and a_v, which assemble structures u and v, respectively, into the same current candidate. For this construction, let $V(a)$ be the fitted candidate's random utility U_π^{fit} after action a, under a common finite-budget protocol.

In history h, previous matched experiments support u and refute v in transferable contexts, so $\mathbb{E}[V(a_u) \mid h] > \mathbb{E}[V(a_v) \mid h]$. In history h', the closed evidence is reversed, so $\mathbb{E}[V(a_u) \mid h'] < \mathbb{E}[V(a_v) \mid h']$. This evidence belongs to local structures inside discarded candidates and does not change the current candidate set or scores.

Any policy $\mu(a \mid z_t)$ that reads only z_t produces the same action distribution for both histories and therefore fails to select the higher-value action in at least one of them. By retaining contextual evidence, the Structural Atlas distinguishes h from h' and recovers the information needed for the decision. □

D.2.2 Protocol Differences and Matched Refit

Proposition 2 (Protocol-difference decomposition)**.** *Let $U_{\pi}^{\mathrm{fit}}(S) = -\log_{10}\max\{\mathcal{L}_{\pi}(S), \varepsilon\}$. If the complete model is evaluated under protocol π_1 and the deletion model is refitted under protocol π_2, then the observed difference satisfies*

$$\widetilde{\Delta}(u \mid S) = \Delta_{\mathrm{MR}}^{\pi_1}(u \mid S) + \Big[U_{\pi_1}^{\mathrm{fit,refit}}(S \setminus \{u\}) - U_{\pi_2}^{\mathrm{fit,refit}}(S \setminus \{u\})\Big]. \tag{30}$$

The second term is the utility shift induced by the protocol change; matching the protocols removes it.

Proof of Proposition 2. Add and subtract $U_{\pi_1}^{\mathrm{fit,refit}}(S \setminus \{u\})$ in the observed difference:

$$\begin{aligned}\widetilde{\Delta}(u \mid S) &= U_{\pi_1}^{\mathrm{fit}}(S) - U_{\pi_2}^{\mathrm{fit,refit}}(S \setminus \{u\}) && (31)\\ &= \Big[U_{\pi_1}^{\mathrm{fit}}(S) - U_{\pi_1}^{\mathrm{fit,refit}}(S \setminus \{u\})\Big] && (32)\\ &\quad + \Big[U_{\pi_1}^{\mathrm{fit,refit}}(S \setminus \{u\}) - U_{\pi_2}^{\mathrm{fit,refit}}(S \setminus \{u\})\Big]. && (33)\end{aligned}$$

The first term is exactly $\Delta_{\mathrm{MR}}^{\pi_1}(u \mid S)$, and the second measures how the protocol change affects the utility of the same deletion structure. When $\pi_1 = \pi_2$, the second term is zero. □

Proposition 3 (Conservativeness of matched refit)**.** *If frozen-parameter deletion and MR use the same evaluation protocol, and the refitting optimizer attains a loss no worse than the inherited parameters, then*

$$\Delta_{\mathrm{MR}}(u \mid S) \le \Delta_{\mathrm{frozen}}(u \mid S). \tag{34}$$

Proof of Proposition 3. Let $S_{-u} = S \setminus \{u\}$ denote the structure after deleting u. Frozen-parameter deletion uses the remaining parameters $\widetilde{\theta}_{-u}$ inherited from the complete model, while MR obtains $\widehat{\theta}_{-u}$ in the same feasible parameter domain. By the assumed quality of the refit,

$$\mathcal{L}(f_{S_{-u},\widehat{\theta}_{-u}}) \le \mathcal{L}(f_{S_{-u},\widetilde{\theta}_{-u}}).$$

The utility $-\log_{10}\max\{\mathcal{L}, \varepsilon\}$ is nonincreasing in the loss, so $U^{\mathrm{fit,refit}}(S_{-u}) \ge U^{\mathrm{fit,frozen}}(S_{-u})$. Subtracting each side from $U^{\mathrm{fit}}(S)$ gives $\Delta_{\mathrm{MR}}(u \mid S) \le \Delta_{\mathrm{frozen}}(u \mid S)$. □

D.2.3 Component Contributions in Linear Models

Proposition 4 (Component contribution in linear models)**.** *Let X_S have full column rank, and let u denote one of its columns. Fit both the complete and reduced models by ordinary least squares on the same data. The increase in residual sum of squares is*

$$\delta_{\mathrm{RSS}}(u \mid S) := \mathrm{RSS}_{S\setminus\{u\}} - \mathrm{RSS}_S = \frac{\widehat{\beta}_u^2}{[(X_S^\top X_S)^{-1}]_{uu}}. \tag{35}$$

Proof. Write $X_S = [X_{-u}, \mathbf{x}_u]$. By the Frisch–Waugh–Lovell theorem, the coefficient of $\mathbf{x}_u$ in the complete model equals the coefficient from univariate regression after residualizing both $\mathbf{y}$ and $\mathbf{x}_u$ against X_{-u}. Deleting u therefore increases the residual sum of squares by $\widehat{\beta}_u^2 \mathbf{x}_u^\top M_{-u}\mathbf{x}_u$, where $M_{-u} = I - X_{-u}(X_{-u}^\top X_{-u})^{-1}X_{-u}^\top$. The block-matrix inverse identity gives $\mathbf{x}_u^\top M_{-u}\mathbf{x}_u = 1/[(X_S^\top X_S)^{-1}]_{uu}$, which yields Equation (35). □

For a loss $\mathcal{L}_{\pi} = \kappa\,\mathrm{RSS}$ with the same normalization $\kappa > 0$ for both models, the matched-refit effect follows from Equation (35):

$$\Gamma_{\mathrm{fit}}(\{u\} \mid S) = \log_{10}\frac{\max\{\kappa[\mathrm{RSS}_S + \delta_{\mathrm{RSS}}(u \mid S)], \varepsilon\}}{\max\{\kappa\,\mathrm{RSS}_S, \varepsilon\}}. \tag{36}$$

When $\kappa\,\mathrm{RSS}_S \ge \varepsilon$, Equation (36) becomes $\log_{10}[1 + \delta_{\mathrm{RSS}}(u \mid S)/\mathrm{RSS}_S]$. The component's contribution depends on its fitted coefficient and its correlations with the other components. ATLAS-SR stores the comparison with the tested assembly and reuses it in contexts supported by compatible evidence.

D.2.4 Finite-Sample Ambiguity and Relational Requirements

Low-error incorrect structures. Let $H^\star$ be the true graph with labeled component roles and H a candidate structure. For data $D = (X, y)$, define

$$\mathcal{A}_\varepsilon(H^\star; D) = \{H \not\simeq H^\star : \mathrm{NMSE}_D(f_H, f_{H^\star}) \le \varepsilon\}. \tag{37}$$

This set characterizes structurally incorrect expressions that fit finite data well. For binary discrimination, write $t = H^\star$ and let a be an incorrect alternative. Under hypothesis $h \in \{t, a\}$, the effective observation model is $y = f_h(X) + \eta$, with $\eta \sim \mathcal{N}(0, \sigma_{\mathrm{eff}}^2 I)$. The minimum average discrimination error under equal priors satisfies

$$P_e^\star(t, a) \ge \frac{1}{2}\left(1 - \frac{\|f_t(X) - f_a(X)\|_2}{2\sigma_{\mathrm{eff}}}\right)_+. \tag{38}$$

Here $\sigma_{\mathrm{eff}} > 0$ is the effective resolution induced jointly by finite numerical precision, fitting variation, evaluation tolerance, and measurement noise.

Derivation of Equation (38). For binary discrimination with equal priors, the minimum average error and total variation distance satisfy

$$P_e^\star = \frac{1 - \mathrm{TV}(P_t, P_a)}{2}, \qquad \mathrm{TV}(P_t, P_a) \le \sqrt{\frac{\mathrm{KL}(P_t \| P_a)}{2}}.$$

For Gaussian distributions with common covariance $\sigma_{\mathrm{eff}}^2 I$ and means $f_t(X)$ and $f_a(X)$,

$$\mathrm{KL}(P_t \| P_a) = \frac{\|f_t(X) - f_a(X)\|_2^2}{2\sigma_{\mathrm{eff}}^2}.$$

Substitution followed by truncation at zero gives the result. □

Probability of completing all required relations. Let $k \in \{0, \dots, 4\}$ be the RCB and $m(k)$ the number of relational commitments that must be correctly materialized and retained in one valid attempt. Let E_j denote the event that the jth commitment is correctly materialized and retained. Then

$$q_k = \Pr\left(\bigcap_{j=1}^{m(k)} E_j\right) = \prod_{j=1}^{m(k)} \Pr(E_j \mid E_1, \dots, E_{j-1}). \tag{39}$$

This factorization also applies when the required relations are dependent.

D.3 Interpreting Noisy Observations

Matching protocols removes the shift caused by evaluating full and deleted assemblies under different conditions (Proposition 2). Within one noisy protocol, the two structures can fit the same perturbations differently. The linear-model identity in Equation (35) makes this dependence explicit through the fitted coefficient and correlations with the other components.

The preceding analysis suggests two further effects. In the effective Gaussian model, greater noise raises the discrimination-error lower bound in Equation (38) for structures with similar predictions. It can therefore become harder to distinguish the required operator from a numerical alternative. Equation (39) also shows how any reduction in conditional relation-completion probabilities can accumulate when several relations must be assembled. Together, these mechanisms offer an interpretation of the noise results in Section 4.5, where the experiment measures overall changes in recovery and prediction.