FROM EVIDENCE TO ACTION: EXPERIENCE- AND KNOWLEDGE-GROUNDED EVOLUTION FOR MATERIALS DISCOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

Iterative materials discovery entails a loop of evaluating generated candidates and feeding the results back into search. Yet evaluation alone does not determine which evidence a task should use or which action a changing search state should trigger. We study this evaluation-to-action gap in heterogeneous, budget-constrained materials discovery. We propose Evidence-to-Action (E2A), an evidence-grounded decision framework for iterative materials discovery that converts internal search experience and retrieved external knowledge into task-conditioned workflows and state-conditioned search actions. The Workflow Policy selects an evidence-use pathway once per run, while within the Full-evidence pathway, the Evolution Policy periodically selects an executable search action from the available state and evidence. In experiments across fourteen tasks, E2A improves hit rate and stable-success rate on eleven tasks each under matched GPT-4o run-native evaluation. Task-conditioned routing exceeds every fixed workflow on the historical return table, while a matched three-task controller study shows task-dependent gains from learned state-conditioned search actions. Together, these findings show how evaluated outcomes become useful through task-appropriate evidence pathways and state-conditioned actions.

1 INTRODUCTION

Materials discovery undergoes a sequential decision process rather than a collection of independent generation calls. After each candidate is evaluated, the search landscape changes: evidence accumulates for particular material families, and the remaining evaluation budget becomes more valuable. Autonomous laboratories already combine historical knowledge, computation, experimentation, and lessons from failed trials (Szymanski et al., 2023). Meanwhile, modern materials generators and benchmarks show that validity, predicted stability, novelty, and experimental realization require different levels of evidence (Zeni et al., 2025; Riebesell et al., 2025). The central decision is therefore not simply what to generate next, but what the accumulated evidence warrants doing next.

Large language model (LLM) agents provide a flexible interface for aggregating evidence to support this decision, and recent systems already offer many of the required ingredients. They can invoke scientific tools iteratively (Bran et al., 2024), select materials data sources and tools (Zhang et al., 2024), maintain memory while refining candidates from property feedback (Takahara et al., 2025), and coordinate retrieval, reasoning, and experimentation in hierarchical systems (Shi et al., 2026). In-silico agents can likewise evaluate and revise materials hypotheses across multiple steps (Ghafarollahi & Buehler, 2026). The coordination of these capabilities remains a problem: a history, a retrieved passage, or a numerical evaluation becomes useful only when the agent can interpret it and translate it into an executable intervention.

We call this problem the *evaluation-to-action gap*. An evaluated trajectory contains successes, near-feasible candidates, recurring failures, and stagnation, but raw history does not identify which pattern can direct the next decision. External documents can supply missing mechanisms or material families, yet retrieval relevance alone does not reveal when such knowledge is needed or how it should interact with trajectory evidence. Experience-learning and verbal-reflection agents show how histories can be reused (Zhao et al., 2024; Shinn et al., 2023), while adaptive retrieval methods

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.

learn when or how deeply to retrieve (Asai et al., 2024; Jeong et al., 2024; Su et al., 2024). We ask a complementary decision question: *how should an agent choose an evidence-use workflow for a heterogeneous materials task, and how should evolving internal and external evidence be converted into a search action under a fixed task budget?*

To address this problem, we propose Evidence-to-Action (E2A), an evidence-grounded decision framework for iterative materials discovery that converts internal search experience and retrieved external knowledge into task-conditioned workflows and state-conditioned search actions. As Figure 1 shows, E2A inserts evidence formation and evidence-conditioned decisions between evaluation and the next search action on the shared generation–evaluation–archive backbone.

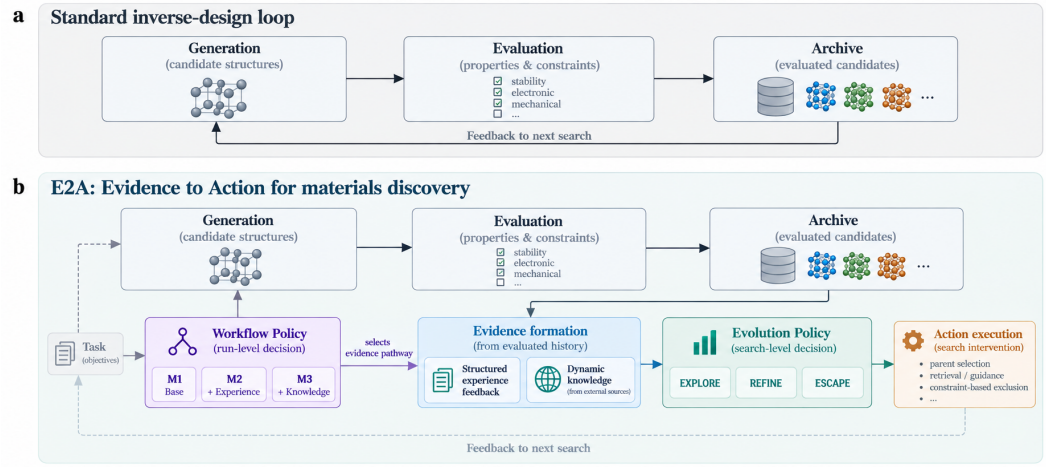

Figure 1: Overview of E2A. (a) A standard inverse-design loop generates, evaluates, and archives candidates before feeding results into the next search. (b) On the same generation–evaluation–archive backbone, E2A adds a run-level Workflow Policy, Evidence Formation from evaluated history and retrieved knowledge, an Evolution Policy in the Full-evidence workflow, and Action Execution that changes subsequent search.

Our key observation is that evidence value is task dependent. Base search can already expose useful families for some objectives; evaluated experience can reveal reusable constraint patterns for others; still other objectives benefit from external structural or chemical knowledge. Always enabling every source can spend context and evaluation budget without improving relevance, and may narrow exploration prematurely. We therefore make evidence use a task-conditioned, run-level decision. Before search begins, a Workflow Policy selects among fixed workflows that progressively expose structured experience and dynamic external knowledge; the selected workflow remains fixed throughout the run.

The Workflow Policy selects a task-conditioned evidence-use pathway before a run and keeps it fixed throughout the run, determining whether search relies on the base process alone, structured search experience, or both experience and retrieved external knowledge. Within the Full-evidence workflow, the Evolution Policy uses the evolving search state and accumulated evidence to select a short-horizon search intervention. Together, the two policies separate the run-level choice of what evidence to use from the in-run decision of how that evidence should change search. Both policies are learned from logged search outcomes and remain fixed during evaluation.

In experiments across fourteen heterogeneous materials-design tasks, E2A improves constraint satisfaction on most tasks compared with independent generation under the same generator and search budget. In the historical workflow evaluation, task-conditioned selection outperforms every fixed workflow, while matched controller comparisons show task-dependent gains from learned search actions. These findings highlight the importance of deciding how evidence is used, rather than assuming that additional evidence is uniformly beneficial.

Our contributions are:

- **Problem formulation.** To bridge the evaluation-to-action gap, we formulate iterative materials discovery as a decision problem: which evidence a task should use and how that evidence should guide subsequent search.
- **Evidence-grounded decision framework.** We introduce Evidence-to-Action (E2A), which combines structured search experience and retrieved external knowledge with two levels of decision-making: task-conditioned workflow selection before a run and state-conditioned search actions within the Full-evidence workflow.
- **Empirical effectiveness.** Across fourteen heterogeneous materials-design tasks, E2A improves run-native constraint satisfaction on most tasks relative to independent generation under the same search budget. Historical workflow evaluation shows that task-conditioned selection outperforms each fixed workflow, and matched controller comparisons show additional task-dependent gains from learned state-conditioned search actions.

2 Related Work

Scientific agents and closed-loop discovery. Scientific agents increasingly connect language models with tools, databases, simulators, and experiments. ChemCrow and Coscientist couple planning with chemistry tools or automated laboratories (Bran et al., 2024; Boiko et al., 2023), while A-Lab learns from computational screening and robotic experiments, including failures (Szymanski et al., 2023). Materials systems add domain grounding: HoneyComb and LLaMP route among knowledge sources and tools (Zhang et al., 2024; Chiang et al., 2025); LLMatDesign and MatAgent iteratively revise candidates from property feedback, memory, and external knowledge (Jia et al., 2024; Takahara et al., 2025); MARS and SparksMatter coordinate retrieval, reasoning, evaluation, and validation over longer loops (Shi et al., 2026; Ghafarollahi & Buehler, 2026). These systems establish the value of iteration and tool use. We focus on the decision interface between them: which evidence path a heterogeneous task should use, and how that evidence should produce an executable search intervention.

Experience and adaptive knowledge use. Self-Refine and Reflexion reuse model feedback or verbal reflections without updating the base model (Madaan et al., 2023; Shinn et al., 2023); ExpeL extracts reusable insights from successful and failed trajectories (Zhao et al., 2024). Our Structured Experience Feedback instead applies a materials-search schema to evaluated trajectories and exposes separate views for policy state and generator context. For external evidence, retrieval-augmented generation (RAG) integrates retrieved knowledge into generation (Lewis et al., 2020), and Self-RAG, Adaptive-RAG, and DRAGIN learn when, how deeply, or what to retrieve (Asai et al., 2024; Jeong et al., 2024; Su et al., 2024). Our Description-First Dynamic Knowledge follows this adaptive lineage but begins from the current scientific bottleneck and renders provenance-grounded evidence. The run-level policy then decides whether the task should use base search, trajectory evidence, or both trajectory and external evidence.

Return-guided temporal abstraction. Options formalize temporally extended actions (Sutton et al., 1999), while offline reinforcement learning (RL) emphasizes the risk of choosing actions weakly supported by logged data (Kumar et al., 2020). Advantage-weighted regression (AWR) offers a conservative supervised update from off-policy returns (Peng et al., 2019). We use these ideas in a narrower, fixed-interface setting: one workflow is selected for a complete run, and a separate policy chooses among three executable actions every three iterations. Both are trained from logged executions and frozen during evaluation. Recent recursive self-improvement work studies longer deployment–update cycles (Duan et al., 2026; Zheng et al., 2026); our evidence-to-policy update is one repeatable stage within such a cycle.

Evidence standards in materials generation. Crystal-generation studies distinguish validity, coverage, predicted stability, and novelty: CDVAE and DiffCSP introduced widely used generative tests and structure-aware models (Xie et al., 2022; Jiao et al., 2023), GNoME combines learned screening with first-principles evaluation (Merchant et al., 2023), and MatterGen uses progressively stronger computational checks plus selected experimental evidence (Zeni et al., 2025). Matbench

Discovery further separates formation-energy prediction from stability classification and retrospective benchmarking from prospective discovery (Riebesell et al., 2025). We adopt these distinctions in reporting: benchmark success, generated-structure surrogate quality, database-relative novelty, and higher-fidelity validation remain separate claims.

3 Evidence-Grounded Hierarchical Search

3.1 Overall framework

We study a budgeted materials-search task $\tau = (d_\tau, C_\tau, P_\tau, B_\tau)$, consisting of a natural-language objective d_τ, material and property constraints C_τ, evaluable properties P_τ, and an evaluation budget B_τ. At iteration t, the fixed generator and evaluator add a batch of evaluated candidates $\mathcal{D}_t$ to the archive,

$$\mathcal{H}_{t+1} = \mathcal{H}_t \cup \mathcal{D}_t. \tag{1}$$

Here, $\mathcal{H}_t$ denotes the archive before iteration t. Scientific outcomes are measured from this archive: hit rate (HR) records constraint satisfaction, stable success rate (SR) additionally requires the benchmark stability criterion, and unique stable-success rate (USR) normalizes distinct stable-success formulas by evaluated occurrences. Detailed metric definitions are provided in Appendix E.

On the shared generation–evaluation–archive backbone in Figure 1, E2A uses a two-timescale decision hierarchy: a run-level Workflow Policy maps the task specification to one evidence-use workflow before search, while an in-run Evolution Policy maps the evolving pre-decision state to an action every h iterations within the Full-evidence workflow:

$$\tau \xrightarrow{\text{once}} w_\tau, \qquad (\tau, o_{t_j}, e_{t_j}, k_{t_j}) \xrightarrow{\text{every } h \text{ iterations}} a_j. \tag{2}$$

w_τ is the run-level evidence-use workflow selected for task τ; a_j is the executable search action selected for decision block j; o_{t_j} is the search-state observation available at its decision boundary; e_{t_j} is structured internal evidence; k_{t_j} is external-knowledge evidence; and h is the action horizon. The Workflow Policy selects w_τ before search and holds it fixed for the entire run. Within the Full-evidence workflow, the Evolution Policy selects a_j for each action block. The observation o_t and internal evidence e_t update after each completed evaluation; external evidence k_t follows the action-dependent retrieval schedule. At each action boundary, the policy reads evidence already available; its action then shapes retrieval and generation over the next block.

3.2 Workflow Policy

Evidence sources are not uniformly useful across tasks. The Workflow Policy therefore selects from

$$\mathcal{W} = \{w^{\text{base}}, w^{\text{exp}}, w^{\text{full}}\}, \qquad w_\tau = \pi_{\theta_W}(x_\tau), \tag{3}$$

where x_τ contains d_τ, C_τ, P_τ, and B_τ. The base workflow (M1) uses the fixed evolutionary engine, the experience workflow (M2) additionally uses Structured Experience Feedback, and the Full-evidence workflow (M3) additionally uses Dynamic Knowledge and enables the Evolution Policy. The selected workflow remains fixed until the budget is exhausted, forming a run-level temporal abstraction (Sutton et al., 1999).

The policy function is a Qwen3-8B adapter (Yang et al., 2025) learned from logged workflow outcomes and frozen during evaluation. Its utility and training procedure are specified in Appendix D.

3.3 Evidence Formation

Structured Experience Feedback. A raw trajectory is long, redundant, and heterogeneous. We therefore transform the completed archive into

$$e_t = F_{\text{EF}}(\tau, o_t, \mathcal{H}_{0:t}), \tag{4}$$

where $\mathcal{H}_{0:t}$ denotes the completed archive history through t; only information available before the next intervention is used. An *Experience Observation* summarizes search dynamics—constraint satisfaction, best-so-far margin, improvement velocity, stagnation, duplication, and failure modes—for policy decisions. A separate *Prompt Feedback* view supplies the generator with a bounded selection

of successes, near-feasible candidates, preserved constraints, bottlenecks, and repeatedly unproductive families. Thus evaluated outcomes become reusable evidence rather than an unstructured transcript.

Description-First Dynamic Knowledge. The trajectory cannot supply mechanisms or material families it has never encountered. External evidence therefore starts from the unresolved deficit exposed by o_t and e_t:

$$(o_t, e_t) \to d_t \to q_t \to R_t \to k_t, \tag{5}$$

where d_t is a knowledge demand, q_t a query, R_t the retrieved records, and k_t the rendered evidence. The current implementation uses BM25 under a shared record contract and reranks records through description facets such as prototype, coordination, connectivity, dimensionality, structural motif, and chemical system. Description facets drive retrieval and reranking, while available property values contribute a lower-weight auxiliary ranking term; missing property values are treated as unavailable rather than as negative evidence. Selected records become provenance-grounded Knowledge Cards; recurrent support may produce Rule Cards, while corpus support remains distinct from support accumulated through subsequently evaluated candidates. At an action boundary, the policy reads the prior retrieval summary k_{t_j}; newly retrieved descriptions condition subsequent generation.

3.4 Evolution Policy

Within w^{full}, the Evolution Policy reads a strictly pre-decision structured observation z_j and chooses

$$a_j = \pi_{\theta_A}(z_j), \qquad a_j \in \{\textsc{Explore}, \textsc{Refine}, \textsc{Escape}\}. \tag{6}$$

Explore broadens candidate regions, parents, and material families; Refine exploits promising or near-feasible candidates while preserving satisfied constraints; and Escape excludes recently saturated regions, motifs, rules, and parents. The action is held for h iterations, although evidence continues to update after every evaluation. Under Escape, Dynamic Knowledge refreshes at the first iteration and remains fixed for the rest of the action horizon.

The policy receives a structured pre-decision observation containing task semantics, search progress, experience feedback, retrieval state, and recent action history. Exact fields and diagnostic projections are provided in Appendix C.

The Evolution Policy is implemented with a Qwen3-8B adapter learned from logged action outcomes and frozen during evaluation. Its reward, temporal credit assignment, observation interface, and training protocol are specified in Appendices C and D. Overall, E2A contains two learned policy functions—the Workflow Policy and the Evolution Policy—which are trained separately from their respective logged outcomes rather than jointly optimized.

4 Experiments

4.1 Experimental setup

We evaluate the fourteen materials-design tasks from the LLEMA benchmark (Abhyankar et al., 2026), covering electronic, mechanical, dielectric, and stability objectives; the evaluation contracts used in this work are specified in Appendix A. Each task-conditioned run uses 300 search iterations and requests two candidates per generation call. E2A is evaluated with DeepSeek, GPT-4o, and GPT-5-mini as proposal backbones. For GPT-4o, we also run a matched independent-generation baseline with the same backbone and budget to separate the effect of the search framework from that of the proposal model. M1 is the evolutionary base configuration under the current execution stack, including its CIF-processing and structure-validation pipeline. M2 adds Structured Experience Feedback, and M3 additionally uses Dynamic Knowledge and the Evolution Policy. E2A selects among M1–M3, activating the Evolution Policy only when M3 is selected. Both policies are fixed during evaluation, and the action horizon is $h = 3$.

For the matched Independent GPT-4o and E2A runs, hit rate (HR) is the percentage of evaluated candidates satisfying task constraints; stable-success rate (SR) additionally requires the benchmark stability criterion. Task definitions, metric formulas, and implementation settings are in Appendices A, E, and F.

4.2 OVERALL MATERIALS-DISCOVERY PERFORMANCE

We first evaluate E2A at the level of overall materials-discovery performance across the fourteen task objectives. The resulting HR/SR comparison is summarized in Table 1.

Table 1: HR/SR (%) across fourteen materials-design tasks, with E2A reported using three proposal backbones.

Method	Wide-bandgap semiconductors		SAW/BAW substrates		High-k dielectrics		Solid-state electrolytes		Piezo harvesters		Transparent conductors		Insulating dielectrics	
	HR	SR	HR	SR	HR	SR	HR	SR	HR	SR	HR	SR	HR	SR
CDVAE	0.04	0.04	0.29	0.00	0.82	0.00	0.04	0.04	42.19	0.00	0.00	0.00	1.06	0.12
G-SchNet	0.00	0.00	0.42	0.00	0.00	0.00	0.00	0.00	0.01	0.00	2.49	0.00	0.01	0.00
DiffCSP	0.00	0.00	0.36	0.00	0.75	0.00	0.00	0.00	41.21	0.00	0.01	0.00	1.13	0.04
MatterGen	6.56	4.15	26.27	0.00	0.64	0.00	5.33	3.11	21.64	0.00	9.38	0.00	0.91	0.10
End2End	0.95	0.79	10.32	0.65	0.00	0.00	0.49	0.30	10.34	0.28	0.00	0.00	0.00	0.00
LLMatDesign	4.19	1.13	**47.59**	0.13	1.35	0.32	2.51	2.44	32.16	1.38	0.04	0.04	0.21	0.08
LLEMA	27.27	22.93	27.12	**14.15**	0.22	0.22	36.83	31.47	5.70	2.14	8.30	5.11	10.98	0.00
Independent GPT-4o	1.43	0.00	1.51	0.00	9.74	0.00	0.52	0.00	59.71	0.00	15.15	0.25	5.21	0.00
E2A + DeepSeek	48.06	42.67	31.32	5.40	13.68	**1.01**	88.13	70.18	68.93	**8.09**	20.55	5.53	17.75	0.00
E2A + GPT-4o	38.48	33.40	22.85	3.00	12.63	0.00	**89.37**	**85.32**	70.80	1.57	13.79	4.02	7.90	0.00
E2A + GPT-5-mini	**63.99**	**55.70**	**39.59**	**5.51**	**15.65**	0.43	79.84	60.49	**74.43**	2.03	**26.96**	**10.60**	**19.93**	0.00

Method	Photovoltaic absorbers		Hard coatings		Hard, stiff ceramics		Structural aerospace		Acousto-optic hybrids		Low-density structural		Perovskite oxides	
	HR	SR	HR	SR	HR	SR	HR	SR	HR	SR	HR	SR	HR	SR
CDVAE	1.07	0.00	0.00	0.00	15.25	0.11	1.18	0.00	**21.85**	0.00	0.00	0.00	0.00	0.00
G-SchNet	0.00	0.00	0.00	0.00	0.20	0.20	0.06	0.00	0.01	0.00	0.17	0.00	0.04	0.00
DiffCSP	1.11	0.00	0.00	0.00	14.75	0.00	0.09	0.00	21.53	0.01	0.00	0.00	0.04	0.00
MatterGen	1.88	0.00	2.12	0.00	8.23	0.00	7.34	0.00	11.24	0.00	0.27	0.00	0.93	0.00
End2End	**24.59**	**10.72**	0.00	0.00	14.27	5.13	0.00	0.00	8.57	0.64	1.99	0.40	0.00	0.00
LLMatDesign	3.92	0.00	0.00	0.00	19.00	0.41	0.00	0.00	15.45	0.55	0.07	0.00	1.10	0.81
LLEMA	10.16	0.53	0.00	0.00	57.18	13.05	29.71	14.19	0.68	0.45	17.55	5.05	1.91	0.54
Independent GPT-4o	7.22	0.18	0.00	0.00	27.91	0.00	12.03	0.00	9.33	0.00	8.14	0.00	1.29	0.00
E2A + DeepSeek	18.80	0.75	7.43	0.00	71.36	23.53	**59.34**	**22.98**	14.63	**2.13**	**60.39**	**6.06**	**4.85**	1.98
E2A + GPT-4o	13.56	5.50	**11.17**	0.00	49.51	6.68	39.30	1.39	16.81	0.18	3.36	0.18	0.87	0.17
E2A + GPT-5-mini	13.50	**6.64**	3.56	0.00	**77.16**	**27.40**	57.18	11.06	16.43	1.69	0.37	0.00	4.28	**2.85**

E2A provides the broadest gains across the benchmark. An E2A variant attains at least one best HR or SR on twelve of fourteen tasks and the highest positive value in twenty-one of the twenty-eight task–metric columns. Under the matched GPT-4o setting, E2A improves HR on eleven tasks and SR on eleven. The largest joint gains appear on solid-state electrolytes, where HR/SR increases from $0.52\%/0.00\%$ to $89.37\%/85.32\%$, and on wide-bandgap semiconductors, where it increases from $1.43\%/0.00\%$ to $38.48\%/33.40\%$. Hard coatings also move from zero HR under independent generation to 11.17% with E2A + GPT-4o, although stable successes remain absent for this task.

The three proposal backbones are complementary rather than uniformly ordered. E2A + GPT-5-mini is strongest on wide-bandgap semiconductors and hard, stiff ceramics; E2A + GPT-4o leads on solid-state electrolytes and hard-coating HR; and E2A + DeepSeek leads on structural aerospace, low-density structural materials, and several stable-success objectives. The remaining task-level exceptions are concentrated in SAW/BAW substrates and photovoltaic absorbers, where established methods retain the best values. These results show that E2A delivers strong discovery performance across proposal models while preserving task-dependent specialization. Novel-hit results are reported in Appendix G.1.

4.3 TASK-DEPENDENT EVIDENCE PATHWAYS

We summarize each workflow's search trajectory using a utility that combines HR, SR, and unique stable-success rate (USR) over the search horizon (Appendix D). Table 2 reports the historical 300-iteration utilities: M1 is highest for stable wide-bandgap semiconductors, M2 for SAW/BAW acoustic substrates, and M3 for solid-state electrolytes. The best fixed workflow varies across tasks, motivating task-conditioned selection. Complete M1–M3 HR/SR results remain in Appendix G.2.

Table 2: Historical trajectory utilities and Workflow Policy choices across fourteen tasks at 300 iterations.

Task	M1 Utility	M2 Utility	M3 Utility	Workflow Policy Choice
Acousto-optic hybrids	**0.076842**	0.064371	0.060384	M2
Electrically insulating dielectrics	0.057538	**0.074697**	0.038008	M2
Hard coating materials	0.023860	**0.025764**	0.004786	M3
Hard, stiff ceramics	0.388371	**0.471280**	0.405326	M3
High-k dielectrics	0.060384	**0.077944**	0.042858	M2
Low-density structural (aerospace)	0.176756	**0.252288**	0.244551	M2
Photovoltaic absorbers	0.095559	0.100536	**0.158869**	M3
Piezo energy harvesters	0.277176	0.271788	**0.287162**	M2
SAW/BAW acoustic substrates	0.278123	**0.320489**	0.249591	M2
Solid-state electrolytes	0.354860	0.386600	**0.807679**	M3
Stable wide-bandgap semiconductors	**0.362233**	0.226986	0.216710	M1
Structural materials for aerospace	0.323146	0.347104	**0.383674**	M3
Toxic-free perovskite oxide	**0.038130**	0.025871	0.031326	M3
Transparent conductors	**0.106424**	0.068008	0.046046	M2
Mean utility of fixed M1				0.187100
Mean utility of fixed M2				0.193838
Mean utility of fixed M3				0.212641
Workflow Policy mean utility				0.234534

The task-conditioned Workflow Policy reaches mean utility 0.234534, exceeding every fixed workflow and improving on the strongest, M3 (0.212641), by 10.30%.

4.4 State-conditioned action selection

We compare the Learned Policy (the learned Evolution Policy) with the Rule-based Controller on solid-state electrolytes, photovoltaic absorbers, and structural aerospace materials. This matched controller comparison uses the same M3 workflow, seed 95, generator, evaluator, 300-iteration budget, and three-iteration action blocks; only action selection differs.

The Learned Policy achieves higher recorded HR and SR on all three tasks (Appendix G.3). For solid-state electrolytes, HR increases from 83.01% to 88.13% and SR from 64.52% to 70.18%. For photovoltaic absorbers, HR increases from 17.92% to 18.80% and SR from 0.22% to 0.75%. Structural aerospace materials provide a case in which HR is nearly unchanged (59.34% for the Learned Policy versus 59.31% for the Rule-based Controller), while SR increases from 19.12% to 22.98%.

Figure 2 follows this case from state to action to outcome. Panel (a) shows the saved pre-decision constraint satisfaction, stagnation fraction, and duplicate rate, which vary as each search progresses. Panel (b) shows distinct action patterns over these evolving states. On the learned run's own pre-decision states, the Learned Policy differs from the recorded rule recommendation at 57 of 100 decision boundaries, showing that its choices do not simply reproduce the rule.

Panel (c) shows higher final cumulative stable-success occurrences for the Learned Policy: 91 versus 78 for the Rule-based Controller, including the initial pool. Together, the logged states, executed actions, and outcome trajectories provide behavioral evidence of state-conditioned policy execution with a different action pattern and higher stable-success accumulation in this matched case, without attributing the improvement to any individual action.

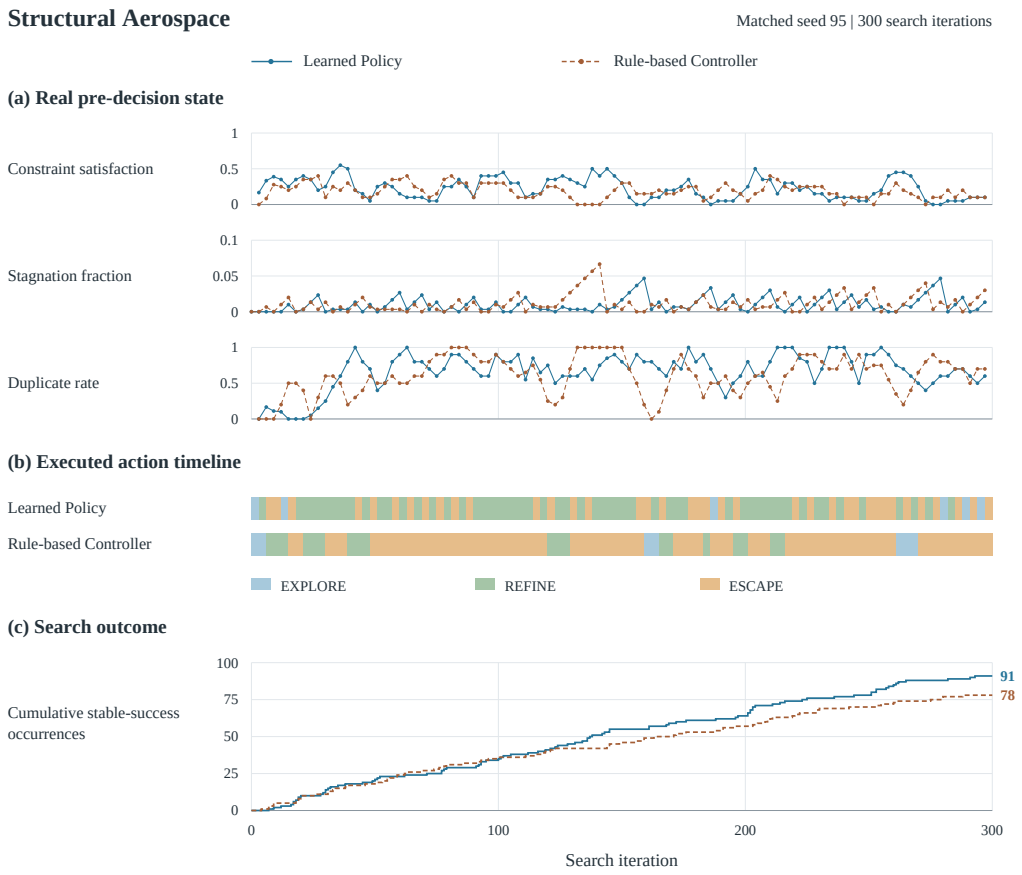

Figure 2: Logged states, executed actions, and search outcomes for Structural Aerospace under the matched seed-95 protocol. (a) Saved pre-decision State v3 values at three-iteration action boundaries: constraint satisfaction (`constraint_satisfaction_ema`), stagnation fraction (`stagnation_fraction`), and duplicate rate (`duplicate_rate_ema`). (b) Executed EXPLORE, REFINE, and ESCAPE actions for the Learned Policy and the Rule-based Controller. (c) Cumulative stable-success occurrences under the recorded evaluation labels, including the initial pool. The Learned Policy differs from the rule recommendation at 57 of 100 action boundaries on the learned-run states. The figure shows temporal correspondence rather than causal effects of individual states or actions.

5 Conclusion

Method. We introduced E2A, an evidence-grounded hierarchical framework for iterative materials discovery that connects evaluated search trajectories to subsequent search decisions. E2A first converts completed search experience into structured internal evidence and, when enabled by the selected workflow, complements it with retrieved external knowledge. A run-level Workflow Policy then selects which evidence-use pathway should govern the run, while an in-run Evolution Policy uses the evolving search state and available evidence to choose short-horizon search interventions. The two policies are learned separately from logged search outcomes, separating task-conditioned evidence selection from state-conditioned action selection.

Findings. Across fourteen materials-design tasks, E2A shows strong overall discovery performance across multiple proposal backbones and improves matched run-native HR and SR on most tasks relative to independent generation. Historical workflow evaluation shows that no single fixed evidence pathway is uniformly preferred: task-conditioned routing achieves higher trajectory utility than every fixed workflow. At the action level, the matched three-task controller study shows task-

dependent gains from the learned state-conditioned policy, supporting the value of adapting search actions as evidence accumulates.

Implications and future work. These results suggest that the value of scientific evidence lies not simply in adding more feedback or external knowledge, but in deciding when each source should matter and how it should change subsequent search. A natural next step is to move beyond policies that are trained once and then frozen, toward a continual improvement loop in which newly collected discovery trajectories are used to refine evidence formation, workflow selection, action policies, and long-horizon credit assignment before the updated system is redeployed. This would extend evidence-to-action from a fixed decision framework into an iterative cycle of search, experience accumulation, policy refinement, and renewed discovery.

AI USE STATEMENT

AI tools were used to assist with research organization, code development, and manuscript drafting. All authors remain responsible for the scientific claims, experimental design, source verification, code, data, and final text.

REFERENCES

Nikhil Abhyankar, Sanchit Kabra, Saaketh Desai, and Chandan K. Reddy. LLEMA: Evolutionary search with LLMs for multi-objective materials discovery. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=TIqzhBvCNB.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/25f7be9694d7b32d5cc670927b8091e1-Abstract-Conference.html.

Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624:570–578, 2023. doi: 10.1038/s41586-023-06792-0. URL https://www.nature.com/articles/s41586-023-06792-0.

Andres M. Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D. White, and Philippe Schwaller. Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 6:525–535, 2024. doi: 10.1038/s42256-024-00832-8. URL https://www.nature.com/articles/s42256-024-00832-8.

Yuan Chiang, Elvis Hsieh, Chia-Hong Chou, and Janosh Riebesell. LLaMP: Large language model made powerful for high-fidelity materials knowledge retrieval. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 25189–25221. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1280. URL https://aclanthology.org/2025.emnlp-main.1280/.

Kamal Choudhary and Brian DeCost. Atomistic line graph neural network for improved materials property predictions. *npj Computational Materials*, 7:185, 2021. doi: 10.1038/s41524-021-00650-1. URL https://www.nature.com/articles/s41524-021-00650-1.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 10088–10115, 2023. doi: 10.52202/075280-0441. URL https://papers.neurips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract-Conference.html.

Yi Duan, Ying Liu, Zirui Tang, Haodong Chen, Jun Zhou, Yumou Liu, Bangrui Xu, Yukai Wu, Sidi Chen, Yuhan Zhou, et al. The last AI built by humans: Toward genuine recursive self-improvement. *arXiv preprint arXiv:2609.11873*, 2026. doi: 10.48550/arXiv.2609.11873. URL https://arxiv.org/abs/2609.11873.

Alireza Ghafarollahi and Markus J. Buehler. Autonomous in-silico inorganic materials discovery via multi-agent physics-aware scientific reasoning. *npj Computational Materials*, 2026. doi: 10.1038/s41524-026-02205-8. URL https://www.nature.com/articles/s41524-026-02205-8. Advance online publication.

Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin A. Persson. Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013. doi: 10.1063/1.4812323. URL https://doi.org/10.1063/1.4812323.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7036–7050. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.naacl-long.389. URL https://aclanthology.org/2024.naacl-long.389/.

Shuyi Jia, Chao Zhang, and Victor Fung. LLMatDesign: Autonomous materials discovery with large language models. *arXiv preprint arXiv:2406.13163*, 2024. doi: 10.48550/arXiv.2406.13163. URL https://arxiv.org/abs/2406.13163.

Rui Jiao, Wenbing Huang, Peijia Lin, Jiaqi Han, Pin Chen, Yutong Lu, and Yang Liu. Crystal structure prediction by joint equivariant diffusion. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/38b787fc530d0b31825827e2cc306656-Abstract-Conference.html.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/0d2b2061826a5df3221116a5085a6052-Abstract.html.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2019. URL https://papers.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

Amil Merchant, Simon Batzner, Samuel S. Schoenholz, Muratahan Aykol, Gowoon Cheon, Ekin Dogus Cubuk, et al. Scaling deep learning for materials discovery. *Nature*, 624:80–85, 2023. doi: 10.1038/s41586-023-06735-9. URL https://www.nature.com/articles/s41586-023-06735-9.

Shyue Ping Ong, William Davidson Richards, Anubhav Jain, Geoffroy Hautier, Michael Kocher, Shreyas Cholia, Dan Gunter, Vincent L. Chevrier, Kristin A. Persson, and Gerbrand Ceder. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013. doi: 10.1016/j.commatsci.2012.10.028. URL https://www.sciencedirect.com/science/article/pii/S0927025612006295.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019. doi: 10.48550/arXiv.1910.00177. URL https://arxiv.org/abs/1910.00177.

Janosh Riebesell, Rhys E. A. Goodall, Philipp Benner, Yuan Chiang, Bowen Deng, Gerbrand Ceder, Mark Asta, Alpha A. Lee, Anubhav Jain, and Kristin A. Persson. A framework to evaluate machine learning crystal stability predictions. *Nature Machine Intelligence*, 7:836–847, 2025. doi: 10.1038/s42256-025-01055-1. URL https://www.nature.com/articles/s42256-025-01055-1.

Tongyu Shi, Yutang Li, Zhanlong Wang, Wenhe Xu, Guolai Jiang, Dawei Dai, Jie Zhou, Hao Huang, Rui He, Seeram Ramakrishna, Paul K. Chu, Wenhua Zhou, and Xue-Feng Yu. Knowledge-driven autonomous materials research via collaborative multi-agent and robotic system. *Matter*, 9(2): 102577, 2026. doi: 10.1016/j.matt.2025.102577. URL https://www.sciencedirect.com/science/article/pii/S2590238525006204.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0377. URL https://papers.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. DRAGIN: Dynamic retrieval augmented generation based on the real-time information needs of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12991–13013. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.702. URL https://aclanthology.org/2024.acl-long.702/.

Richard S. Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1–2):181–211, 1999. doi: 10.1016/S0004-3702(99)00052-1. URL https://www.sciencedirect.com/science/article/pii/S0004370299000521.

Nathan J. Szymanski, Bernardus Rendy, Yuxing Fei, Rishi E. Kumar, Tanjin He, David Milsted, Matthew J. McDermott, Max Gallant, Ekin Dogus Cubuk, Amil Merchant, Haegyeom Kim, Anubhav Jain, Christopher J. Bartel, Kristin Persson, Yan Zeng, and Gerbrand Ceder. An autonomous laboratory for the accelerated synthesis of inorganic materials. *Nature*, 624:86–91, 2023. doi: 10.1038/s41586-023-06734-w. URL https://www.nature.com/articles/s41586-023-06734-w.

Izumi Takahara, Teruyasu Mizoguchi, and Bang Liu. Accelerated inorganic materials design with generative AI agents. *Cell Reports Physical Science*, 6(12):103019, 2025. doi: 10.1016/j.xcrp.2025.103019. URL https://www.sciencedirect.com/science/article/pii/S2666386425006186.

Tian Xie and Jeffrey C. Grossman. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Physical Review Letters*, 120(14):145301, 2018. doi: 10.1103/PhysRevLett.120.145301. URL https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.120.145301.

Tian Xie, Xiang Fu, Octavian-Eugen Ganea, Regina Barzilay, and Tommi S. Jaakkola. Crystal diffusion variational autoencoder for periodic material generation. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=03RLpj-tc_.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. doi: 10.48550/arXiv.2505.09388. URL https://arxiv.org/abs/2505.09388.

Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Zilong Wang, Aliaksandra Shysheya, Jonathan Crabbé, Shoko Ueda, Roberto Sordillo, Lixin Sun, Jake Smith, Bichlien Nguyen, Hannes Schulz, Sarah Lewis, Chin-Wei Huang, Ziheng Lu, Yichi Zhou, Han Yang, Hongxia Hao, Jielan Li, Chunlei Yang, Wenjie Li, Ryota Tomioka, and Tian Xie. A generative model for inorganic materials design. *Nature*, 639:624–632,

2025. doi: 10.1038/s41586-025-08628-5. URL https://www.nature.com/articles/s41586-025-08628-5.

Huan Zhang, Yu Song, Ziyu Hou, Santiago Miret, and Bang Liu. HoneyComb: A flexible LLM-based agent system for materials science. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3369–3382. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.192. URL https://aclanthology.org/2024.findings-emnlp.192/.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: LLM agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19632–19642, 2024. doi: 10.1609/aaai.v38i17.29936. URL https://ojs.aaai.org/index.php/AAAI/article/view/29936.

Tong Zheng, Xidong Wu, Zheng Zhang, Zhankui He, Chaoyi Zhang, Benjamin Coleman, Ruoqiao Wei, Di Bai, Haolin Liu, Rui Liu, Xue Wang, Yue Zhuan, Wang-Cheng Kang, Renkai Xiang, Heng Huang, Xinwu Cheng, and Yunsong Guo. Dream-RSI: Recursive self-improvement through evolving worlds. *arXiv preprint arXiv:2609.14858*, 2026. doi: 10.48550/arXiv.2609.14858. URL https://arxiv.org/abs/2609.14858.

A Task and Workflow Definitions

A.1 Task contract

Table 3 and the grouped panel contracts in Appendix A.1 instantiate the task representation in Section 3: the table records the exact scientific constraints, and the grouped text separates the evaluation and execution contracts used by the reported panels.

Table 2 uses full task names; other tables use their corresponding short names. "Low-density structural" denotes Low-Density Structural (Aerospace), whereas "Structural aerospace" denotes Structural Materials for Aerospace; these are distinct tasks. CIF stands for Crystallographic Information File.

Table 3: Exact hit constraints from the versioned evaluation contract. Intervals include both endpoints. "Same" refers to the benchmark column. Task-spec constraints also define Uniform-CIF hit qualification; their property sources differ (Appendix A.1).

Task	Benchmark-v1 hit	Task-spec-v2 / Uniform-CIF hit
Acousto-optic hybrids	$d \in [3, 9]$, $\epsilon_p \in [8, 85]$	Same
Insulating dielectrics	$E_g \geq 2.5$, $\epsilon \geq 8$	Same
Hard coatings	$K \in [200, 500]$, $G \in [100, 300]$	$K \geq 200$, $E_f \leq -1$, $E_g \geq 3$
Hard, stiff ceramics	$K \in [100, 300]$, $G \in [60, 200]$	Same
High-k dielectrics	$\epsilon \in [10, 90]$, $E_g \in [2.5, 6.5]$	Same
Low-density structural	$\rho \leq 3.5$, $G \in [65, 195]$	Same
Photovoltaic absorbers	$E_g \in [0.7, 2]$, $E_f \leq 0$; restrictions P	Same
Piezo harvesters	$d \geq 8$, $\epsilon_p \in [10, 8000]$	Same
SAW/BAW substrates	$G \in [25, 150]$, $\epsilon \in [3.7, 95]$	Same
Solid-state electrolytes	$E_f \leq -1$, $E_g \geq 2$; restriction S	Same, plus $E_h \leq 5$
Wide-bandgap semiconductors	$E_g \geq 2.5$, $E_f \leq -1$	Same, plus $E_h \leq 5$
Structural aerospace	$K \in [100, 300]$, $G \in [60, 200]$, $E_f < 0$	$\rho \leq 5$, $K \geq 100$, $G \geq 40$, $E_h \leq 5$
Toxic-free perovskite oxide	$E_g \geq 2$, $K \in [90, 135]$; restriction T	Same
Transparent conductors	$E_g > 1.5$, $\sigma \in [500, 30000]$	$E_g > 3$, $\sigma \in [5000, 500000]$

K, G are moduli in GPa, E_g is the band gap in eV, E_f, E_h are formation energy and energy above hull in eV/atom, and ρ is density in g/cm^3. The fields d, ϵ_p, ϵ, and σ denote `piezo_max_dij`, `piezo_max_dielectric`, `dielectric_constant`, and `electrical_conductivity`; their thresholds are applied in the frozen property outputs' native scale.

Restriction S requires at least one of Li, Na, K, Mg, Ca, or Al. Restriction T excludes Pb, Cd, Hg, Tl, Be, As, Sb, Se, U, and Th. Restriction P excludes Hg, Cd, Pb, Tl, As, Be, Se and the implemented non-abundant set: Au, Ag, Pt, Pd, Rh, Ir, Os, Ru, La, Ce, Pr, Nd, Pm, Sm, Eu, Gd, Tb, Dy, Ho, Er, Tm, Yb, Lu, Sc, Y, In, Ga, Ge, Te, Ta, Hf, Re. These are the implemented composition tests, rather than additional structural-family constraints inferred from task names.

PANEL CONTRACTS

Historical workflow ablations retain their run-specific seeds.

Evaluation protocols. Run-native Legacy uses Task-spec-v2 hit; success additionally passes benchmark stability, and available run-native property evidence is retained. Benchmark-compatible evaluation uses Benchmark-v1 hit and reports thermodynamic stability separately from the hit constraints. Uniform-CIF applies Task-spec-v2 hit to the generated CIF; uSR adds $E_h \leq 0.1$ (or 0.2 in the supplementary tier), and missing required properties are excluded from the complete-case denominator.

Run budgets and pairing. The seed103 and seed95 active panels use 300 iterations and request two candidates per generation call; actual evaluated occurrences vary and include the initial pool and repeated formulas. Seed103 pairs Independent GPT-4o and E2A under the same generator and budget, while run-native and complete-case Uniform-CIF endpoints retain their own denominators. The matched controller study covers three tasks at seed95 and fixes M3, generator, retriever, evaluator, budget, action vocabulary, and dwell while changing only the action selector.

Cohort provenance. DeepSeek E2A task-level frozen outputs assemble seed93 workflow runs and six seed95 Evolution Policy slots; this is not one single-seed fourteen-task cohort.

Accounting and stability rules. Run-native denominators retain evaluated occurrences, including repeats. Unique formula numerators deduplicate only after the panel-specific frozen formula parser succeeds. For Legacy stability, when both energies are available, $E_h \leq 0.1$ and $E_f \leq -1$; if only one is available, its corresponding bound is tested, and if neither is available, stability is unknown.

The generated-CIF stack uses the Crystal Graph Convolutional Neural Network (CGCNN) for E_g; the Atomistic Line Graph Neural Network (ALIGNN) for K, G, E_f, E_h, ϵ, d, and ϵ_p; pymatgen structure density for ρ; and an ALIGNN-derived conductivity proxy using power-factor and Seebeck outputs for σ. The frozen surrogate manifest binds the model artifacts and evaluator code. Run-native property values may follow different source routes and are not relabeled as generated-CIF predictions.

A.2 FIXED WORKFLOW CONFIGURATIONS

Independent GPT-4o is the independent-generation reference. M1–M3 are workflow configurations rather than names for the complete method; the Workflow Policy selects M1, M2, or M3 before search and holds that workflow for the run. M1 is the evolutionary base configuration under the current execution stack, M2 adds Structured Experience Feedback, and M3 additionally adds Description-First Dynamic Knowledge; the Evolution Policy is enabled only in M3.

The LLEMA baseline preserves the original LLEMA candidate-processing contract. Candidates that become evaluable only through the CIF-processing or structure-validation procedures of the current execution stack are not credited to LLEMA. M1 instead uses the current execution stack and serves as the base configuration for the M1–M3 internal ablations.

Table 4: Compact mapping from experimental shorthand to evidence-use configurations.

Label	Workflow	Search	Structured EF	Dynamic knowledge
M1	Evolutionary base	evolutionary + native history	–	–
M2	Experience	evolutionary + native history	✓	–
M3	Full-evidence	evolutionary + native history	✓	✓

B EVIDENCE REPRESENTATIONS

B.1 STRUCTURED EXPERIENCE FEEDBACK

Internal and external evidence are represented by four bounded objects:

Table 5: Internal and external evidence representations.

Evidence object	Main content	Consumer
Experience Observation	progress, margin, stagnation, failures, diversity	Evolution Policy
Prompt Feedback	bottlenecks, successes, near-feasible cases, failed/recent patterns	generator
Knowledge Card	source, material identity, structural context, available properties	generator / evidence summary
Rule Card	recurring supported pattern, applicable context	generator / evidence summary

The internal entries are derived from completed candidate evaluations; the observation used for an action contains no post-decision candidates or rewards. Prompt Feedback retains the progress summary, dominant constraint failures, representative successful and near-feasible candidates, recurrent failures, and recently tried rules. Knowledge Cards retain source/document identifiers, material formulas, chemical-system context, explicit structural descriptors or motifs with evidence spans, and available measured properties. Rule Cards retain supporting source records and evidence spans for recurring structural patterns in their shared context. The policy reads the structured interface below; rendered feedback text and diagnostic projections are not substitutes for State v3.

B.2 Description-First Dynamic Knowledge

The external-evidence path follows

$$(o_t, e_t) \rightarrow \text{knowledge deficit} \rightarrow \text{query} \rightarrow \text{retrieved records} \rightarrow \text{evidence cards}. \tag{7}$$

The implementation uses BM25 over the frozen run corpus and reranks with description facets including prototype, coordination, connectivity, dimensionality, motif, and chemical system. Available property values provide a lower-weight auxiliary signal; missing properties are unavailable rather than negative evidence. In the six seed-95 matched runs, the property coefficient is $\lambda_p = 0.2$, multiplied by each record's available-property coverage. Under ESCAPE, retrieval refreshes in the first iteration of the three-iteration block and is then held fixed.

The Knowledge Block renders the external objects in Table 5. For example, the frozen solid-state-electrolyte record at iteration 0 includes `Na3CaMg3AlF14 (mp-1191931)`; the identifier itself is not treated as property evidence. Corpus files, complete prompts, and retrieval records are retained as run artifacts rather than reproduced in the paper.

C Policy Interfaces and Executable Actions

C.1 Run-level Workflow Policy

Before the first search iteration, the Workflow Policy reads the task description, constraints, evaluable properties, and budget horizon and returns one workflow corresponding to M1–M3. The decision remains fixed while subsequent evaluations update the evidence.

The frozen interface accepts the task name and objective, explicit success conditions, evaluable properties, and search horizon. It returns exactly {`"workflow_option":"M1|M2|M3", "budget_horizon":integer`}; the returned horizon must match the request. Complete prompts and task instances are run artifacts rather than appendix text.

C.2 Three-iteration Evolution Policy

The Evolution Policy is enabled in the full-evidence workflow. At each three-iteration boundary, it reads the pre-decision observation and selects one action for the block.

The action changes candidate sources, parent selection, exclusions, rule use, and generation instructions; the fixed generator remains responsible for producing formulas and structures.

Table 6: Executable Evolution Action contract. All actions are held for three search iterations.

Action	Search intervention	Retrieval within the block
EXPLORE	broaden families, regions, parents, and rules	use the enabled evidence
REFINE	focus on near-feasible or high-quality candidates	use the enabled evidence
ESCAPE	exclude saturated parents, motifs, rules, and regions	refresh once, then hold

The Evolution Policy reads the canonical pre-decision interface below and returns exactly

$$\{\texttt{evolution_action} : \texttt{EXPLORE|REFINE|ESCAPE},\ \texttt{action_dwell_rounds} : 3\}. \quad (8)$$

Extra keys and free-form explanation are disallowed. Action-specific instructions are combined with task constraints, design rules, evaluated candidates, and the evidence enabled by the fixed workflow. The action controls the next candidate proposal while the evaluator and archive remain downstream.

C.3 OBSERVATION REPRESENTATIONS

State v3 is the 24-dimensional execution record. The Learned Evolution Policy input combines task semantics, the State v3 snapshot and missing-value mask, Structured Experience Feedback, the latest prior retrieval summary, action history, and completed-block summaries. The 18-feature classical and eight-feature identifiability views are audit projections, not the language-model input.

STRUCTURED PRE-DECISION SCHEMA

Task fields are fixed per run; the remaining fields are refreshed from completed history and read at action boundaries.

Task context. Name, goal, and evaluable properties from the task specification.

Search state. Iteration, state version, and named State v3 features. Masked values serialize as null rather than zero.

Experience. Stage, stagnation length, pool size, pool chemical-system count, and recent hit, success, and generation-failure rates.

Prior retrieval. Availability, source iteration, goal, bottleneck, refresh reason, pool size, card counts, and retrieval overlap. An absent snapshot has `availability=false`.

Action history. Previous action, dwell time, and recent action distribution. Completed-block summaries belong to history and are not the current block return. Experience Feedback's generator view renders search progress (best score, stagnation and recent success), constraint failures (counts, rates and representative gaps), and frequencies of recently tried rules. Enabled sections depend on the search stage. The compact policy view exposes the aggregate fields above rather than the full rendered text.

State v3 separates progress, feasibility, validity, success, best margin and margin velocity, stagnation, duplicate/failure statistics, violation concentration, archive diversity, island statistics, and action history. Its 24 numeric entries are bounded to $[-1, 1]$ and accompanied by a missing-value mask. The 18-feature classical projection and eight-feature identifiability view are diagnostic projections; neither defines the full language-model input. Query text and current action-conditioned retrieval are excluded from the pre-decision retrieval summary.

D OFFLINE POLICY LEARNING

D.1 WORKFLOW-POLICY TRAINING

The Workflow Policy is a Qwen3-8B adapter (Yang et al., 2025) trained from logged workflow outcomes. Its trajectory utility is

$$U(\tau, w, B_\tau) = 0.35\ \mathrm{AUC(HR)} + 0.55\ \mathrm{AUC(SR)} + 0.10\ \mathrm{AUC(USR)}. \quad (9)$$

Here AUC (area under the curve) is implemented as the normalized discrete average of each cumulative metric over evaluated candidate occurrences within the search horizon; USR is the formula-deduplicated stable-success rate (Appendix E). A numerical critic fits these utilities. Contract supervised fine-tuning (SFT) teaches the legal outputs, and critic-greedy labels supervise one hard-label quantized low-rank adaptation (QLoRA) update (Dettmers et al., 2023).

D.2 Evolution-policy training

The low-level immediate reward is

$$r_t = 0.30\,\Delta M_t + 0.25\,S_t^{\mathrm{uniq}} + 0.15\,\Delta HR_t + 0.10\,N_t^{\mathrm{arc}} + 0.10\,\Delta D_t - 0.05\,C_t^{\mathrm{dup}} - 0.10\,C_t^{\mathrm{inv}} - 0.05\,C_t^{\mathrm{repeat}}. \tag{10}$$

Here ΔM_t is improvement in the best-so-far normalized worst-constraint margin, S_t^{uniq} is the number of newly successful formulas divided by evaluated candidates in the iteration, and ΔHR_t is the change in hit rate between consecutive iterations. N_t^{arc} is the fraction of attempted candidates with formulas new to the archive, and ΔD_t is the change in the evaluator's diversity score. C_t^{dup}, C_t^{inv}, and C_t^{repeat} penalize deduplication removals, invalid generation, and evaluations of previously archived formulas, respectively, normalized by generation attempts; a failed generation round sets $C_t^{\mathrm{inv}} = 1$. Components are clipped to $[-1, 1]$, initial change terms are zero, and new-success, novelty, and diversity rewards are zero when no candidate is evaluable. The action-block return G_j^A is the normalized discounted sum of immediate rewards in the three-iteration block starting at t_j, with discount factor γ:

$$G_j^A = \frac{\sum_{i=0}^{2} \gamma^i r_{t_j+i}}{\sum_{i=0}^{2} \gamma^i}, \qquad \gamma = 0.95. \tag{11}$$

An offline critic and out-of-fold advantages select supported positive-advantage actions for one QLoRA+AWR update (Peng et al., 2019; Dettmers et al., 2023). The balanced data contain 28 runs and 448 action blocks, with seed 88 used for fitting and seed 89 for validation.

Table 7: Final adapter training configuration. Sample counts refer to serialized inputs.

Policy	Train / validation	Target	Rank / alpha	Epochs / max tokens
Workflow	252 / 14	critic-greedy workflow	8 / 16	3 / 768
Evolution	60 / 135	supported positive advantage	16 / 32	1 / 2048

Both adapters use Qwen3-8B with NF4 double quantization, bfloat16 computation, dropout 0.05, microbatch size 1, weight decay 0.01, warmup ratio 0.05, and an A100-SXM4-40GB. Workflow training uses learning rate 5×10^{-5} and gradient accumulation 8; Evolution uses 2×10^{-5} and accumulation 16.

E Evaluation and Statistical Protocol

Published baselines in Table 1 retain benchmark-compatible HR/SR, whereas Independent GPT-4o and E2A retain their run-native HR/SR. These rows therefore provide protocol-stratified evidence rather than a single matched leaderboard. The GPT-4o independent/E2A comparison is paired at seed 103. The DeepSeek E2A row assembles frozen task-level outputs from seed-93 workflow runs and seed-95 Evolution Policy runs; it is not a single-seed matched fourteen-task cohort. The groups retain their respective task constraints, property sources, and evaluation labels (Appendix A.1). For benchmark-compatible evaluation, stability requires energy above hull at most 0.1 eV/atom and, when formation energy is available, formation energy at most -1 eV/atom; when only formation energy is available, its bound determines stability. Properties follow the benchmark's Materials Project-first route with surrogate fallback. With N_B evaluated occurrences, including repeats, and

benchmark predicates H_B and S_B, the reported percentages are

$$\mathrm{HR}_B = 100\frac{\sum_{x=1}^{N_B}\mathbf{1}[H_B(x)]}{N_B}, \qquad \mathrm{SR}_B = 100\frac{\sum_{x=1}^{N_B}\mathbf{1}[S_B(x)]}{N_B}. \tag{12}$$

An incomplete task–method evaluation has no aggregate score. The matched run-native and Uniform-CIF analyses below use their own frozen evaluation outputs. Materials Project and pymatgen provide database and structure-analysis infrastructure (Jain et al., 2013; Ong et al., 2013); the property models include CGCNN and ALIGNN (Xie & Grossman, 2018; Choudhary & DeCost, 2021).

Let $\mathcal{E}$ be the multiset of run-native evaluated candidate occurrences and let $N_{\mathrm{eval}} = |\mathcal{E}|$. Repeated occurrences remain in this denominator. With $H(x)$ denoting satisfaction of the run-native task constraints, $S_\delta(x)$ requires $H(x)$ and the recorded stability test: $E_h(x) \le \delta$ and $E_f(x) \le -1$ eV/atom when both are available, or the corresponding bound when only one is available; stability is unknown when neither is available. The Legacy endpoints are

$$\mathrm{HR} = \frac{\sum_{x\in\mathcal{E}}\mathbf{1}[H(x)]}{N_{\mathrm{eval}}}, \qquad \mathrm{SR}_\delta = \frac{\sum_{x\in\mathcal{E}}\mathbf{1}[S_\delta(x)]}{N_{\mathrm{eval}}}. \tag{13}$$

Tables express HR/SR as percentages unless labeled as fractions. The primary hull-energy threshold is $\delta = 0.1$ eV/atom; the Uniform-CIF supplement also reports sensitivity at $\delta = 0.2$. If $f_L(x)$ is the declared formula string with whitespace removed, the unique stable-success rate used in workflow utility is

$$\mathrm{USR}_\delta = \frac{|\{f_L(x) : x \in \mathcal{E},\ S_\delta(x)\}|}{N_{\mathrm{eval}}}. \tag{14}$$

Thus Legacy USR deduplicates declared formulas, not CIF or structure identities.

For complete-case Uniform-CIF evaluation, let $\mathcal{E}_U \subseteq \mathcal{E}$ contain occurrences with every task-required property available and $N_U = |\mathcal{E}_U|$. Then

$$\mathrm{uHR} = \frac{\sum_{x\in\mathcal{E}_U}\mathbf{1}[H_U(x)]}{N_U}, \qquad \mathrm{uSR}_\delta = \frac{\sum_{x\in\mathcal{E}_U}\mathbf{1}[H_U(x)\wedge E_h^U(x)\le\delta]}{N_U}. \tag{15}$$

The matched seed-103 NUHY analysis in Table 8 retains its frozen Evaluation V2 identity implementation. Let $f_O(x)$ be the formula identity recorded after successful parsing of the candidate's declared formula, and let $\mathcal{F}^O_{\mathrm{ref}}$ be the identities marked present by the frozen reference built from 103 corpus chunks. Its normalized unique novel-hit yield is

$$\mathrm{NUHY}_O = 100\,\frac{\left|\{f_O(x) : x \in \mathcal{E}_U,\ H_U(x),\ f_O(x) \notin \mathcal{F}^O_{\mathrm{ref}}\}\right|}{N_U}. \tag{16}$$

Missing required properties exclude an occurrence from N_U and are reported through coverage; they are not replaced by run-native values. Invalid or unparseable declared formulas cannot enter the unique numerator. Repeated complete-case occurrences remain in N_U, while deduplication is applied only to the numerator. This frozen declared-formula identity is retained for Table 8 and is not replaced by a later CIF-composition identity.

Archive novelty N_t^{arc} in the training reward is a different quantity from NUHY_O: it measures formulas new to the evolving run archive and is not a frozen-reference discovery metric.

The matched controller comparison fixes the task, seed, 300-iteration budget, M3 workflow, generator, retriever, evaluator, action vocabulary, and three-iteration dwell; only the action selector differs. The reported controller endpoints are matched outcomes from that protocol.

F Reproducibility

The reproducibility record consists of the task and panel contracts above, the policy interfaces, the evaluation definitions, and the frozen run artifacts containing full prompts, manifests, configurations, and candidate-level outcomes. The executable sequence is:

```
select workflow once from task semantics and budget
for each search iteration:
    at a three-iteration boundary in the full-evidence workflow:
        select EXPLORE, REFINE, or ESCAPE from prior evidence
    apply held action and refresh retrieval as needed
    generate and evaluate candidates
    update archive, observation, and enabled evidence
```

F.1 RECORDED EXECUTION EXCERPT

The frozen seed-95 Structural Aerospace controller artifact provides a directly checkable block excerpt. After search iteration 60, the Learned Policy history contained 23 cumulative stable-success occurrences, including four from the preceding twelve iterations. It executed REFINE for iterations 61–63 and reached 24 cumulative occurrences. The separate Rule-based Controller history contained 24 occurrences at its corresponding boundary, executed ESCAPE, and reached 26 after iteration 63. This excerpt documents two realized histories rather than a counterfactual effect of either action; no larger end-to-end trace is assembled from different artifacts.

G SUPPLEMENTARY RESULTS

G.1 MATCHED SYSTEM RESULTS

Table 8: Matched GPT-4o normalized unique novel-hit yield (NUHY) across fourteen tasks at seed 103 and 300 iterations.

Task	Independent GPT-4o NUHY	E2A NUHY
Acousto-optic hybrids	0.00	**15.75**
Insulating dielectrics	0.00	**6.28**
Hard coatings	0.00	0.00
Hard, stiff ceramics	0.58	**36.54**
High-k dielectrics	0.00	**11.95**
Low-density structural	0.84	**2.30**
Photovoltaic absorbers	0.18	**6.88**
Piezo harvesters	11.87	**63.99**
SAW/BAW substrates	0.43	**18.91**
Solid-state electrolytes	0.00	**74.43**
Wide-bandgap semiconductors	0.00	**4.10**
Structural aerospace	0.38	**8.87**
Toxic-free perovskite oxide	0.43	**0.87**
Transparent conductors	5.81	**9.39**

G.2 FIXED EVIDENCE WORKFLOWS AND WORKFLOW ROUTING

Table 9: Historical fixed-workflow ablation across fourteen tasks, reported as frozen run-native HR/SR (%).

Task	M1		M2		M3	
	HR	SR	HR	SR	HR	SR
Acousto-optic hybrids	16.52	**3.39**	14.66	2.63	**20.97**	1.06
Insulating dielectrics	24.53	0.00	**29.81**	0.00	12.25	0.00
Hard coatings	**2.57**	**2.57**	1.43	1.43	0.23	0.23
Hard, stiff ceramics	**85.90**	17.49	80.80	**46.38**	77.80	25.30
High-k dielectrics	**13.02**	0.87	11.48	**3.09**	8.79	1.76
Low-density structural	37.50	5.53	56.56	**5.81**	**57.77**	4.37
Photovoltaic absorbers	14.44	0.53	14.93	1.41	**15.13**	**6.41**
Piezo harvesters	**60.33**	6.65	59.00	8.06	46.42	**17.77**
SAW/BAW substrates	**44.34**	14.39	37.83	**15.65**	39.23	12.68
Solid-state electrolytes	37.76	32.40	60.25	38.52	**90.66**	**79.61**
Wide-bandgap semiconductors	**41.32**	**33.88**	22.75	20.90	25.45	22.65
Structural aerospace	60.31	16.19	61.66	**22.87**	**70.97**	20.43
Toxic-free perovskite oxide	**10.63**	0.54	4.49	1.00	4.68	**1.17**
Transparent conductors	**31.28**	5.53	13.28	**7.05**	10.19	4.16

G.3 MATCHED CONTROLLER RESULTS

Table 10: Learned Policy and matched Rule-based Controller results. HR/SR are run-native fractions, and ΔU is their utility difference (Learned minus Rule-based).

	HR		SR		Utility		
Task	Learned	Rule-based	Learned	Rule-based	Learned	Rule-based	ΔU
Solid-state electrolytes	**.8813**	.8301	**.7018**	.6452	**.7274**	.7266	+.0008
Photovoltaic absorbers	**.1880**	.1792	**.0075**	.0022	**.0937**	.0925	+.0012
Structural aerospace	**.5934**	.5931	**.2298**	.1912	**.3530**	.3175	+.0356