MATGYM: AN AGENT-CENTRIC ENVIRONMENT BRIDGING MATERIALS DISCOVERY AND REPRODUCIBLE SYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Materials discovery requires exploring how composition and processing affect performance when these relationships are only partly known. We introduce MatGym, an agent-centric environment for evaluating autonomous exploration and reproducible delivery across five simulated materials laboratories. Given a material specification and a resource budget, agents conduct open-ended exploration: they select candidates, control preparation steps, and choose computations and measurements to inform subsequent actions. Each task ends with a sample and a recipe, evaluated for service performance, agreement with the sample's preparation history, and replication on fresh batches. Across 52 tasks using the largest implemented candidate libraries, with three attempts per task, Codex and Claude Code achieve success rates of 43.6% and 41.0%, respectively. Similar overall rates mask different strengths across laboratories, and each system solves fewer than 30% of tasks successfully in all three attempts. Separate delivery checks distinguish material performance, process traceability, and batch reproducibility. In auxiliary transfer tests with an earlier Codex configuration, success on new ALD tasks increases from 30.6% without memory to 55.6% with prior campaign memory, while gains across material systems vary. MatGym provides a controlled setting for evaluating how agents explore unfamiliar material processes, deliver reproducible outcomes, and reuse experience on new tasks.

1 INTRODUCTION

Materials researchers investigate how composition and processing affect performance under the intended conditions of use. Existing knowledge guides this search, while experiments help resolve unfamiliar process responses. A successful experiment must also be captured in a recipe that can reproduce the result (Canty & Abolhasani, 2024). Scientific agents and autonomous laboratories are beginning to plan experiments and coordinate synthesis and characterization (Boiko et al., 2023; M. Bran et al., 2024; Szymanski et al., 2023; Canty & Abolhasani, 2026; Yuan et al., 2025). Evaluating these systems requires checking the outcomes of those decisions. Can an agent investigate unfamiliar material processes, identify a workable preparation route, and reproduce its performance within a limited experimental budget?

Existing benchmarks evaluate important parts of this process. Matbench and Matbench Discovery assess property prediction and stable-crystal screening (Dunn et al., 2020; Riebesell et al., 2025). CrystalGym and MADE extend evaluation to computational design and adaptive candidate search (Govindarajan et al., 2025; Malik et al., 2026). Olympus evaluates experiment planning, Alchemy-Bench assesses synthesis procedure prediction, and ChemGymRL supports interactive laboratory operations (Häse et al., 2021; Kim et al., 2025; Beeler et al., 2024). These efforts provide ways to study candidate selection, feedback use, and experimental workflows. However, the benchmarks compared in Table 1 do not jointly test whether a delivered sample meets its intended service requirements and whether its recipe succeeds on fresh batches. Evaluating both would connect an agent's discovery decisions to the performance and reproducibility of its final output.

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.

Table 1: Comparison with representative benchmarks. ✓: included; –: not evaluated; ∼: partially supported.

Benchmark	Open-ended workflow	In-task feedback	Synthesis execution	Service test	Fresh-batch replication
Matbench Discovery (Riebesell et al., 2025)	–	–	–	–	–
AlchemyBench (Kim et al., 2025)	–	–	–	–	–
CrystalGym (Govindarajan et al., 2025)	∼	–	–	–	–
Olympus (Häse et al., 2021)	✓	✓	–	–	–
MADE (Malik et al., 2026)	✓	✓	–	–	–
ChemGymRL (Beeler et al., 2024)	✓	✓	✓	–	–
MatGym	✓	✓	✓	✓	✓

We introduce MatGym, an agent-centric environment that connects materials discovery with reproducible synthesis (Figure 1). Each task gives the agent a customer brief describing the required properties, conditions of use, and available resources. The agent chooses which candidates to investigate, what to compute, how to synthesize samples, and which measurements to collect. It can revise these choices as results arrive. Tasks are open-ended in their solution paths: they specify a goal without prescribing an experimental sequence or a single correct recipe. MatGym includes five simulated laboratories covering atomic layer deposition (ALD) dielectrics, an age-hardening alloy, and garnet, liquid, and polymer electrolytes. The primary evaluation contains 52 tasks using the largest implemented candidate libraries; an earlier 250-task release supports auxiliary experiments. Hidden process responses give agents something to investigate, while batch variation and measurement noise require them to judge whether an observed improvement is reproducible.

Each task ends with two submissions: a sample and the recipe used to make it. The sample must meet all specified requirements after the prescribed service test. The submitted recipe must match the sample's candidate and process history, then succeed on at least two of three fresh batches, with all task budgets respected. This evaluation combines material performance, process traceability, and synthesis reproducibility. Agents are free to follow different workflows, while their outputs are checked against the same requirements. Simulation allows us to repeat tasks under controlled conditions and inspect how individual decisions affect the outcome.

Our experiments examine task completion, reproducibility, and reuse of experience. On the 52 primary tasks, Codex and Claude Code reach similar average success rates but differ in which laboratories they handle well and which tasks they solve consistently. Sample, recipe, and fresh-batch checks identify where a delivery fails. Auxiliary controls examine tool availability for an earlier agent configuration and feedback content for scripted strategies. Separate transfer experiments provide Codex with memory from prior Claude Code campaigns. Success is higher with memory on new ALD tasks, while gains vary across material systems.

Our contributions are:

- We introduce five simulated materials laboratories with open-ended tasks, partially known process responses, persistent samples, and resource limits. The evaluation includes 52 maximum-library tasks and an auxiliary release of 250 small-library tasks.
- We define an acceptance protocol that checks service performance, matches a submitted recipe to the sample's preparation history, and tests replication on fresh batches.
- We evaluate two coding-agent systems and six scripted baselines, analyze tool access, scripted feedback use, and repeated-attempt consistency, and test whether recorded campaign experience helps solve new tasks.

2 Related work

Scientific-agent benchmarks and experimental discovery. Scientific-agent evaluation now extends beyond domain knowledge to research activity. ChemBench, MaScQA, MatSciBench and ALDbench assess domain reasoning (Mirza et al., 2025; Zaki et al., 2024; Zhang et al., 2026; Yanguas-Gil et al., 2025), while MatTools tests the use of materials software (Liu et al., 2025). ScienceAgentBench, MLE-bench and CORE-Bench assess work with scientific data and code

(Chen et al., 2025b; Chan et al., 2025; Siegel et al., 2025); SciAgentArena adds interactive research tasks with stepwise verification across scientific domains (Liu et al., 2026). Other benchmarks make experimentation itself the unit of evaluation. BoxingGym tests experimental design and model discovery in probabilistic worlds (Gandhi et al., 2025), SciGym uses systems-biology models (Duan et al., 2025), NewtonBench tests law discovery under counterfactual physics (Zheng et al., 2026), and DiscoveryWorld places agents in persistent scientific scenarios (Jansen et al., 2024).

Closed-loop materials discovery and process optimization. Materials evaluation spans property and stability prediction (Dunn et al., 2020; Riebesell et al., 2025) and computational candidate search in crystal and molecular spaces (Govindarajan et al., 2025; Gao et al., 2022; Brown et al., 2019; Nigam et al., 2023). PSCG-Net uses multiscale crystal graphs for property prediction (Chen et al., 2025a). Sequential settings now evaluate how candidates are selected from feedback: MADE lets agents propose structures and query a formation-energy oracle under a budget (Malik et al., 2026), while Olympus and Summit compare experiment planners on reaction surfaces (Häse et al., 2021; Felton et al., 2021). ChemGymRL and PC-Gym extend simulation to laboratory operations and process control (Beeler et al., 2024; Bloor et al., 2026). In atomic layer deposition (ALD), optimization studies compare strategies under noisy growth measurements (Paulson et al., 2021); more recent work tests language-model agents choosing dose times in a simulated reactor (Yanguas-Gil, 2026a) and probing an unknown atomic-layer-processing chemistry (Werbrouck et al., 2025).

Autonomous laboratories and digital sandboxes. Physical platforms have closed the loop between planning, synthesis and measurement: Coscientist and the A-Lab operate chemistry and inorganic-materials workflows (Boiko et al., 2023; Szymanski et al., 2023), and a self-driving deposition system adjusts conditions for individual thin-film samples using in situ optical feedback (Zheng et al., 2025). Language-model tool use has been demonstrated in chemistry (M. Bran et al., 2024), including an agent-compatible interface to a physical ALD reactor (Yanguas-Gil et al., 2026). EMSeek coordinates microscopy analysis, crystal reconstruction, and property prediction through multiple agents (Chen et al., 2026). In parallel, MULTITASK models interacting agents, instruments and laboratory resources (Kusne & McDannald, 2023); digital sandboxes have been proposed for studying broader research and lab-management decisions (Kusne & McDannald, 2026). Recent evaluation work also distinguishes agent knowledge, tool use and closed-loop synthesis tasks (Yanguas-Gil, 2026b), while self-driving-lab benchmarking examines how experimental campaigns should be compared by their acceleration and improvement over reference strategies (Adesiji et al., 2026).

3 The MatGym Benchmark

3.1 Task Definition and Agent Interaction

Each task asks an agent to produce a sample and a synthesis recipe under a fixed resource budget. Figure 1 shows how candidate search, tool use, and feedback lead to delivery and outcome checks. We represent a task as

$$\tau = (\ell, g, \mathcal{C}, \mathbf{b}, \theta), \tag{1}$$

where ℓ specifies the laboratory, g contains the required properties and conditions of use, $\mathcal{C}$ is the public design space, $\mathbf{b}$ gives the resource limits, and θ contains the hidden process parameters. The agent knows the available materials, operations, targets, and budgets, but must explore which compositions and process settings meet the requirements. Unknown responses are task-specific and hidden from the agent: the interface does not return θ or internal sample states. Appendix A provides an example brief and the agent prompt.

For example, the earlier garnet task illustrated in Figure 1 requires a pellet to retain ionic conductivity of at least 0.642 mS/cm, electronic conductivity of at most 10^{-4} mS/cm, and relative density of at least 0.92 after 30 days of simulated air storage. Hotter sintering can improve density but increase lithium loss, so the agent must investigate the tradeoff through its choice of preparation and measurements. The service-test protocol retires the tested pellet, leaving fewer samples for further trials and delivery. The agent submits an unconsumed pellet and its synthesis recipe for fresh-batch replication; Appendix A gives a complete brief from the primary maximum-library release.

The agent chooses its next action from the task description and its interaction history. It can query candidates, run computations, create and select samples, apply process steps, collect measurements, and submit a result. Each laboratory defines its own process steps and instruments through the

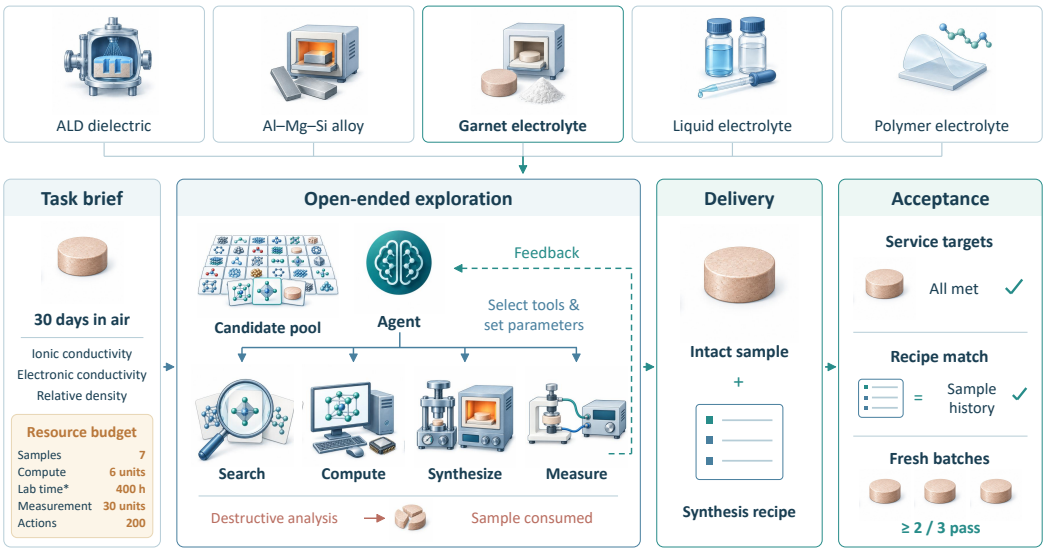


Figure 1: Overview of MatGym. Across five simulated laboratories, agents explore candidates and adapt tool use to feedback under resource budgets, then deliver a sample and its synthesis recipe. Acceptance requires service performance, recipe–history consistency, and success on at least two of three fresh batches. An earlier garnet task is shown. *Lab time excludes the simulated storage wait.

same interface. Tool descriptions specify arguments, costs, and known limitations. Computations provide nominal predictions or public data that may differ from the properties of the current batch. Tasks prescribe neither an action sequence nor a unique recipe. We call the workflow *open-ended* because the agent can choose and revise its actions; the available design space remains bounded by the laboratory.

Actions change persistent sample states and consume resources. Each new sample draws a hidden batch, while measurements can be noisy, fail, or consume the sample. In particular, an agent can perform a destructive service test using the task's conditions of use, but must then prepare another sample for delivery. The budget covers samples, computation, laboratory time, measurement cost, and actions. Requests exceeding sample, computation-cost, or computation-time limits are refused; other budget violations are recorded and cause acceptance to fail. A valid delivery ends the episode with one unconsumed sample and a recipe specifying a candidate and at most eight process steps. The action interface and state handling are detailed in Appendix A.

3.2 Laboratories and Simulator Design

Table 2 summarizes the five laboratories and the largest implemented design libraries used in the primary evaluation. Their sizes count different design units: complete formulations in liquid electrolyte, and material–precursor combinations, compositions, or monomer designs in the other laboratories. These units describe each laboratory's design space. The listed sample budgets apply to these maximum-pool tasks. An earlier 250-task release offers 3–8 named starting materials per laboratory, alongside process and formulation choices, and is used for auxiliary experiments. Appendix B details the design spaces and model sources; Appendix C.2 specifies the evaluation cohorts and library comparisons.

We combine existing scientific software and public data with process simulators developed for MatGym. For the alloy, pycalphad supplies equilibrium phase fractions and phase boundaries (Otis & Liu, 2017). For garnet, pymatgen analyzes Materials Project entries to obtain phase-stability information (Ong et al., 2013; Jain et al., 2013), while OBELiX supplies measured conductivity records (Therrien et al., 2026). These calculations and records are prepared offline and exposed through lookup tools. Liquid and polymer conductivity models are fitted to public measurements (de Blasio et al., 2024; Bradford et al., 2023). Our simulators connect these inputs to processing, measurement, and service behavior.

To support studies of sequential scientific decision-making, the process models retain mechanisms such as precipitation and coarsening (Shercliff & Ashby, 1990; Deschamps & Brechet, 1998), ce-

Table 2: Design libraries and sample budgets for the primary maximum-pool evaluation. Parentheses indicate destructive measurement channels.

Laboratory	Key properties	Largest library	Measurement channels	Sample budget
ALD dielectric	Capacitance, leakage	22	5 (3)	8
Al–Mg–Si alloy	Hardness, conductivity	443	4 (2)	6
Garnet electrolyte	Ionic conductivity, leakage	1,055	6 (3)	7
Liquid electrolyte	Ionic conductivity, stability	2,301	7 (1)	10
Polymer electrolyte	Ionic conductivity, modulus	531	8 (2)	8

ramic densification (Su & Johnson, 1996), and precursor exposure in ALD trenches (Gordon et al., 2003). Candidate composition and processing jointly determine the modeled properties. ALD and alloy aging use literature-scale kinetic parameters. Simplified kinetics make repeated exploration computationally practical.

Processing, measurement, and service tests act on a shared sample state. A process changes quantities such as phase fraction, grain size, or residual solvent, which then determine multiple measured properties. For example, residual solvent can improve a polymer film's conductivity while reducing its modulus. Task parameters remain fixed within an episode, new batches introduce sample variation, and observation noise is applied when measurements are taken. Persistent states, coupled properties, and experimental costs give agents choices about what to investigate next.

Simulator checks assess calibration agreement, selected physical trends, and numerical consistency. Tests cover effects such as alloy overaging, lithium loss during sintering, and self-limiting ALD growth. We also compare polymer moment equations with a separate kinetic Monte Carlo implementation and check selected repeatable process steps for consistency when their durations are split. Appendix B.7 specifies the tested conditions and tolerances. Equations, parameter sources, and implementation checks are documented in Appendices B and B.7.

3.3 Task Construction and Quality Control

The primary maximum-pool evaluation contains 52 tasks, with ten per laboratory except twelve in liquid electrolyte. The earlier small-candidate release contains five task families per laboratory and ten admitted instances per family, giving 250 tasks. These are the small-candidate task versions, separate from the extended libraries of Table 2. Families vary application requirements, service conditions, or process limits. One family per laboratory changes a mechanism substantially from its nominal behavior without disclosing the change in the brief. These counterfactual tasks probe adaptation when familiar assumptions no longer hold. The generator also supports a specification grammar that combines service conditions, property limits, and budgets into new briefs. These grammar-generated tasks use the same admission checks and form separate releases.

Task generation draws hidden process parameters and uses a privileged reference search to set an achievable target. The designer uses the hidden world to establish feasibility; neither that world nor the reference recipe is provided to the agent. For the earlier 250-task release, it searches 256–2,240 grid recipes, depending on the laboratory and task family, and checks feasibility on a nominal batch and at least three of four adverse batches. The target is set to a laboratory-specific fraction, 93–98.5%, of the best qualifying grid recipe's minimum performance over the nominal batch and the adverse batches that satisfy non-target constraints. Local refinement improves the reference recipe without raising the target beyond that derived from the grid. The resulting target stays below the best qualifying grid score, subject to the laboratory's rounding precision. Appendix A.2 gives the generation algorithm and thresholds.

A task is admitted only if the reference recipe succeeds on three laboratory seeds and the open-loop handbook recipe fails on all three. These gates select feasible tasks that challenge a fixed nominal recipe. They define the evaluated task distribution; controlled experiments then test feedback dependence.

Release checks also test consistency when a process duration is split into consecutive steps and search for recipes that exploit gaps in the reference grid or simulator. During evaluation, unusually high performance relative to the reference search triggers an audit. The audit compares performance

with a numerical reference search and leaves the acceptance score unchanged. When an audit leads to a model or target correction, affected tasks are regenerated and reevaluated. The checks and observed issues are described in Appendix B.7.

3.4 Evaluation Protocol

Evaluation checks the delivered sample, the submitted recipe, and their process consistency. Let $Q_\tau(s)$ equal one when sample s meets every requirement in g under the prescribed conditions of use, including any service treatment, and zero otherwise. The evaluator executes the submitted recipe r exactly as written on three fresh batches, producing samples s_1, s_2, s_3. These executions keep the task's hidden parameters fixed and use new random seeds for the batches. Replication is performed after the episode and does not spend the agent's exploration budget.

Let $V(s, r)$ indicate that the submission contains an unconsumed sample and a valid recipe, $M(s, r)$ indicate that the recipe has the same candidate and ordered process steps as the delivered sample, and $\mathbf{c}(h)$ be the resources spent over interaction history h. A run is accepted if

$$A_\tau = V(s,r)\,\mathbf{1}[\mathbf{c}(h) \preceq \mathbf{b}]\; Q_\tau(s)\, M(s,r)\, \mathbf{1}\left[\sum_{j=1}^{3} Q_\tau(s_j) \geq 2\right]. \tag{2}$$

The budget comparison is componentwise. Missing submissions and failed recipe executions count as failures. Invalid recipes are refused at submission, allowing the agent to correct them before the episode ends. The consistency check compares system-validated arguments, so defaults, numeric notation, and normalized formulation fractions can express the same step; it does not equate reordered, split, or merged steps. Both the original sample and the replicas are checked against the same requirements. Acceptance combines process traceability with sample performance and fresh-batch replication.

The acceptance rate measures whether exploration yields a traceable, reproducible solution under the simulated conditions. For repeated runs, we first average acceptance over seeds within each task and then average over tasks, giving each task equal weight. We also report sample and recipe outcomes and consistency across attempts to explain performance. Action logs support these analyses and replay, but no particular action sequence or intermediate answer is required for success. The exact checks and replication seeds are specified in Appendix A.

4 Experiments

4.1 Experimental setup

Our primary evaluation uses the largest implemented candidate libraries in the five laboratories (Table 2). It contains 52 tasks: ten each in alloy, garnet, polymer, and ALD, and twelve in liquid electrolyte. Both agent systems complete three attempts per task. Table 3 uses alloy, garnet, and liquid max_v2 and polymer and ALD max_v1, and includes all valid completed runs. Earlier small-candidate experiments provide auxiliary evidence: scripted strategies cover 250 tasks (Table 9), and a 25-task agent pilot supports auxiliary diagnostics and tool controls. These task versions are distinct from the primary evaluation.

We compare six non-privileged scripted strategies: a handbook recipe (open loop), compute-guided selection, single-metric feedback, random search, Gaussian-process Bayesian optimization, and test-and-switch. A privileged reference accesses the hidden world and serves as an attainable-performance reference. We evaluate Codex with GPT-5.6-Sol and Claude Code with Opus 5 as complete agent systems, with shell access to the public task and laboratory client. The task interface exposes briefs, candidate information, budgets, and tool descriptions, while withholding hidden process parameters and internal sample states. Maximum-pool agent runs have a 3,600 s wall-clock limit per task; all methods use the sample budgets in Table 2 and other resource limits specified in each task. The auxiliary pilot, library-size comparisons, and memory-transfer experiments retain the earlier Codex configuration with GPT-5.6-Luna. The pilot uses a 3,000 s limit and the earlier small-library task budgets; Table 8 summarizes evaluation sizes and limits. Its two Codex controls disable computation and measurement together or withhold candidate names and compositions; feedback controls for two scripted strategies use the primary maximum-pool tasks. Appendix C gives CLI settings and resource limits.

Table 3: Success rates (%) on the final maximum-pool tasks, reported as mean ± sample standard deviation across three complete repeats. Rates average tasks equally. Random search and BO use a fixed public recipe pool; the privileged reference uses hidden information.

Method	Alloy	Garnet	Liquid	Polymer	ALD	Overall
Open loop	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Compute / prior data	23.3 ± 5.8	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	4.5 ± 1.1
Single-metric feedback	10.0 ± 10.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.9 ± 1.9
Random search	3.3 ± 5.8	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.6 ± 1.1
Bayesian optimization	13.3 ± 15.3	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	2.6 ± 2.9
Test-and-switch	26.7 ± 5.8	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	5.1 ± 1.1
Codex	56.7 ± 15.3	6.7 ± 5.8	75.0 ± 8.3	13.3 ± 5.8	60.0 ± 0.0	43.6 ± 4.8
Claude Code	30.0 ± 10.0	13.3 ± 15.3	80.6 ± 9.6	36.7 ± 11.5	36.7 ± 5.8	41.0 ± 6.8
Privileged reference	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0

We score final acceptance using the protocol in Section 3.4. The primary table reports task-averaged success rates (%) and sample standard deviations across three complete repeats of the same tasks. Each task receives equal weight after averaging its available runs. Provider interruptions are excluded. Repeated attempts vary both the agent trajectory and the laboratory seed, and are distinct from the three fresh batches evaluated within each delivery. Primary scripted conditions and feedback controls use three laboratory seeds per task; auxiliary scripted conditions use five; the pilot has three attempts for Codex and one for Claude Code and each control. Single-attempt pilot intervals use Wilson's method. All reported acceptance rates use the final protocol, including strict matching between the submitted recipe and the delivered sample's preparation history. Appendix A gives the agent prompt and exact matching rules. Appendix C.2 gives cohort details and statistical methods.

4.2 End-to-end task performance

The six scripted baselines test different ways to choose and revise a recipe (Table 3). Open loop delivers a handbook recipe without measurements; compute/prior data uses calculations or public reports to select a candidate or process setting, with an additional thickness-calibration measurement in ALD. Single-metric feedback selects recipes mainly by a local property rather than the full service requirements. Random search and Bayesian optimization test recipes from a public pool and remake the best observed recipe; Bayesian optimization uses a Gaussian process with expected improvement to choose trials. Test-and-switch checks service performance and follows predefined alternatives when a trial fails. Under these implementations, all six succeed only in alloy; test-and-switch has the highest overall success rate, 5.1%. The scripted methods use fixed rules or a finite recipe pool, while agents can construct other process sequences and parameter settings. Open-loop failure is expected because task admission requires the handbook recipe to fail on the admission seeds.

Both agent systems solve tasks in all five laboratories, with performance varying by material system. Codex achieves $43.6 \pm 4.8\%$ overall success and Claude Code $41.0 \pm 6.8\%$, but their relative performance varies by laboratory. Codex has higher observed rates in alloy (56.7% versus 30.0%) and ALD (60.0% versus 36.7%). Claude Code performs better in liquid electrolyte, polymer, and garnet, although garnet remains difficult for both systems (6.7% and 13.3%). Similar overall rates hide different strengths across laboratories. The comparison covers complete agent systems, with each laboratory defining its own specifications and design space.

The privileged reference passes every task and repeat by executing a reference recipe selected using hidden information. Its success confirms feasibility in the full simulated environment; this check does not establish feasibility within the public recipe pool. We next separate sample performance, process consistency, and fresh-batch replication to locate the remaining delivery failures.

4.3 Further analysis

Process consistency and fresh-batch replication impose distinct delivery requirements. After submission and budget checks, 51.9% of Codex runs and 60.9% of Claude Code runs meet service targets. Rates fall to 48.1% and 44.9% after recipe-history matching, then to 43.6% and 41.0% after fresh-batch checks (Figure 2, left). For Claude Code, the larger drop occurs at history matching.

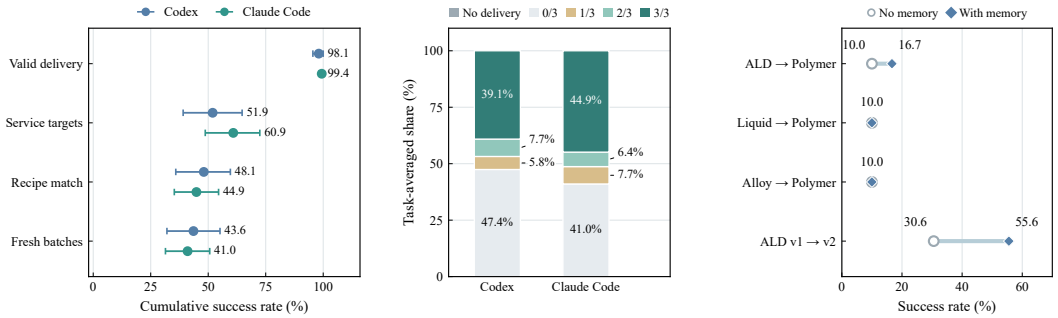


Figure 2: Delivery and memory transfer. Left: cumulative acceptance on 52 tasks, with task-bootstrap 95% intervals. Middle: fresh-batch outcomes, independent of other checks. Right: transfer using the earlier Codex configuration; connected points compare no memory with source-campaign memory on the same target tasks. All panels weight tasks equally.

These cumulative rates depend on earlier checks and show that final acceptance evaluates process traceability as well as material performance.

Submitted recipes vary in how consistently they reproduce across fresh batches. All three batches pass in 39.1% of Codex runs and 44.9% of Claude Code runs, whereas all three fail in 47.4% and 41.0%, respectively (Figure 2, middle). Recipes in the intermediate categories pass on only one or two batches, showing why a single successful remake can miss batch sensitivity. These outcomes are scored independently of the other delivery checks and thus differ from final task success.

Solving a task once is more common than solving it consistently across attempts. Following Yao et al. (2025), we distinguish pass@3 (at least one success in three attempts) from pass^3 (success on all three). Codex achieves 61.5% pass@3 and 26.9% pass^3; Claude Code reaches 69.2% and 19.2% (Figure 3). Claude Code covers more tasks at least once, whereas Codex solves more tasks on all three attempts. The two systems therefore differ in task coverage and consistency, even though their average success rates are close. These attempts vary both the agent trajectory and laboratory seed.

Success with prior campaign memory is higher on new ALD tasks, with smaller gains across materials. We provide the earlier Codex configuration with a fixed memory file from successful Claude Code campaigns (Figure 2, right). Source campaigns satisfy the same final acceptance protocol, including strict recipe-history matching. On twelve new ALD tasks with different hidden worlds, success is 30.6% without memory and 55.6% with ALD-source memory; three-attempt consistency is 8.3% and 25.0%, respectively. On ten polymer tasks, success is 10.0% without memory and 16.7% with ALD-source memory, while the liquid-electrolyte and alloy memory conditions both achieve 10.0%. Model weights remain unchanged: the agent reuses prior experience through the memory file. The observed gain is larger within ALD and varies by source on the polymer tasks. Appendix C.3 describes the memory construction and paired comparisons.

Codex has lower observed success without computation and measurement on the pilot tasks. On the auxiliary 25-task pilot, success falls from 20% with all tools to 4% when both computation and measurement are disabled (Figure 4, left). The observed trend favors tool access in the earlier agent configuration. Withholding candidate names and compositions instead yields 40% success, indicating that these identifiers are not required to solve the pilot tasks. The tool control disables computation and measurement together. Appendix C.2 gives the pilot cohorts and paired comparisons.

Both scripted strategies perform better with task-relevant feedback on the alloy tasks. We compare real and shuffled feedback on ten final alloy tasks with three seeds each, following related dry-lab evaluations (Gupta et al., 2025; Wainrib et al., 2026). Shuffling substitutes readings from an unrelated sample while preserving actual sample outcomes, costs, and destructive effects. Test-and-switch success falls from 26.7% to 10.0%, and Bayesian optimization from 13.3% to 0% (Figure 4, middle). Both tested scripted strategies have higher observed success with readings from their own experiments. Appendix C.2 reports paired effect intervals and the full controls.

Performance changes with library expansion differ across laboratories and agent systems. Using the earlier Codex configuration and Claude Code, success is lower on maximum-library tasks in six of eight comparisons across alloy, garnet, polymer, and ALD (Figure 4, right). Claude Code's

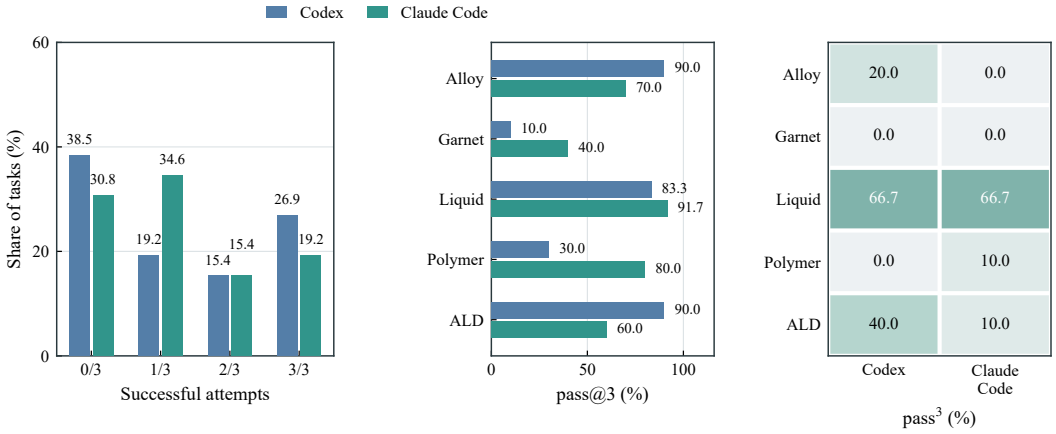


Figure 3: Repeated-attempt success on the final 52 tasks. Left: distribution of successful attempts per task. Middle: pass@3 (at least one success) by laboratory. Right: pass^3 (all three succeed); darker cells indicate higher rates. Values are observed task percentages. Agent attempts are distinct from fresh-batch replication.

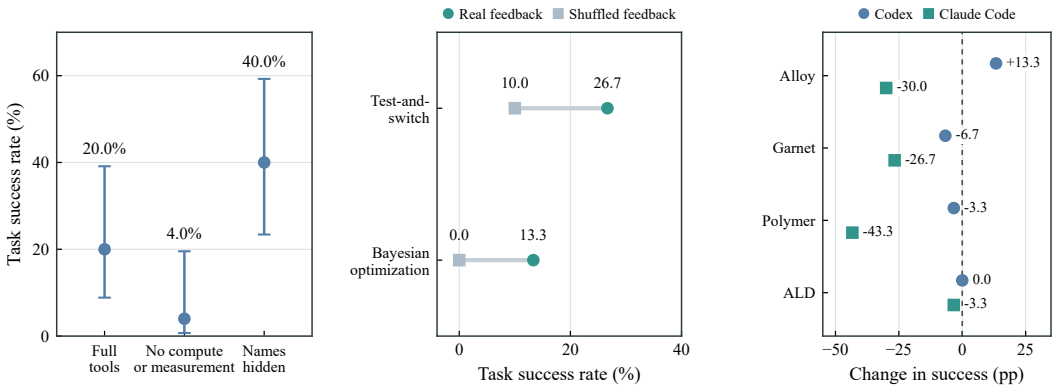


Figure 4: Tools, feedback, and candidate libraries. Left: Codex tool controls on the 25-task pilot, with Wilson 95% intervals. Middle: real versus shuffled feedback on ten alloy tasks, three seeds each. Right: change from small to maximum libraries (percentage points). Codex uses the earlier configuration in the left and right panels. Cohort details are in Appendix C.2.

largest drop is in polymer, from 80.0% to 36.7%, while the earlier Codex cohort's alloy success rises from 33.3% to 46.7%, reversing the alloy ranking. Performance on small libraries thus does not carry over uniformly to expanded design spaces. The comparison spans releases with different specifications, screening budgets, and repeat counts; it measures performance across these releases. Appendix Table 10 gives library sizes and absolute rates.

5 Conclusion

We introduced MatGym, a controlled setting for studying how agents explore partially known material processes and turn their findings into reproducible outcomes. Across five simulated laboratories, agents choose candidates, control preparation, and submit a sample and its recipe. The evaluations distinguish sample performance, process traceability, fresh-batch replication, and consistency across attempts. Auxiliary controls show lower pilot-agent success without laboratory tools and higher scripted-strategy success with task-relevant feedback. Transfer experiments show higher success with prior campaign memory on new ALD tasks and smaller or no gains across the tested material systems. More open synthesis data, including process histories and batch outcomes, can further improve these environments.

AI USE STATEMENT

Generative AI tools were used to assist with drafting parts of the manuscript, polishing and improving the clarity of the writing, retrieving and discovering relevant literature, and generating or refining code. All AI-generated or AI-assisted text, references, and code were carefully reviewed, verified, and, when necessary, revised by the authors before inclusion in the paper or implementation. The authors take full responsibility for the final manuscript, references, experimental code, and reported results.

REPRODUCIBILITY STATEMENT

Upon acceptance, we will publicly release all benchmark datasets and code.

REFERENCES

Adedire D. Adesiji, Jiashuo Wang, Cheng-Shu Kuo, and Keith A. Brown. Benchmarking self-driving labs. *Digital Discovery*, 5:14–27, 2026. doi: 10.1039/D5DD00337G.

Chris Beeler, Sriram Ganapathi Subramanian, Kyle Sprague, Mark Baula, Nouha Chatti, Amanuel Dawit, Xinkai Li, Nicholas Paquin, Mitchell Shahen, Zihan Yang, Colin Bellinger, Mark Crowley, and Isaac Tamblyn. ChemGymRL: A customizable interactive framework for reinforcement learning for digital chemistry. *Digital Discovery*, 3(4):742–758, 2024. ISSN 2635-098X. doi: 10.1039/d3dd00183k. URL http://dx.doi.org/10.1039/D3DD00183K.

Maximilian Bloor, José Torraca, Ilya Orson Sandoval, Akhil Ahmed, Martha White, Mehmet Mercangöz, Calvin Tsay, Ehecatl Antonio Del Rio Chanona, and Max Mowbray. PC-Gym: Benchmark environments for process control problems. *Computers & Chemical Engineering*, 204: 109363, 2026. doi: 10.1016/j.compchemeng.2025.109363. URL https://doi.org/10.1016/j.compchemeng.2025.109363.

Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06792-0. URL http://dx.doi.org/10.1038/s41586-023-06792-0.

Gabriel Bradford, Jeffrey Lopez, Jurgis Ruza, Michael A. Stolberg, Richard Osterude, Jeremiah A. Johnson, Rafael Gomez-Bombarelli, and Yang Shao-Horn. Chemistry-Informed Machine Learning for Polymer Electrolyte Discovery. *ACS Central Science*, 9(2):206–216, 2023. ISSN 2374-7951. doi: 10.1021/acscentsci.2c01123. URL http://dx.doi.org/10.1021/acscentsci.2c01123.

Nathan Brown, Marco Fiscato, Marwin H.S. Segler, and Alain C. Vaucher. GuacaMol: Benchmarking Models for de Novo Molecular Design. *Journal of Chemical Information and Modeling*, 59(3):1096–1108, 2019. ISSN 1549-960X. doi: 10.1021/acs.jcim.8b00839. URL http://dx.doi.org/10.1021/acs.jcim.8b00839.

Richard B. Canty and Milad Abolhasani. Reproducibility in automated chemistry laboratories using computer science abstractions. *Nature Synthesis*, 3:1327–1339, 2024. doi: 10.1038/s44160-024-00649-8.

Richard B. Canty and Milad Abolhasani. The past, present and future of self-driving laboratories. *Nature Reviews Chemistry*, 10:523–537, 2026. doi: 10.1038/s41570-026-00847-2.

Jerry F. Casteel and Edward S. Amis. Specific conductance of concentrated solutions of magnesium salts in water-ethanol system. *Journal of Chemical & Engineering Data*, 17(1):55–59, 1972. doi: 10.1021/je60052a029.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, Aleksander Madry, and Lilian Weng. MLE-bench: Evaluating Machine Learning Agents on Machine Learning Engineering. In *International Conference on Learning Representations*, pp. 50466–50494, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/7e3767db483c942b883eb4f8cfb74e31-Paper-Conference.pdf.

Guangyao Chen, Zhilong Wang, and Fengqi You. PSCG-Net: A Multiscale Crystal Graph Neural Network for Accelerated Materials Discovery. *Journal of Chemical Information and Modeling*, 65(20):10871–10884, 2025a. doi: 10.1021/acs.jcim.5c01460.

Guangyao Chen, Wenhao Yuan, and Fengqi You. Bridging electron microscopy and materials analysis with an autonomous agentic platform. *Science Advances*, 12(14):eaed0583, 2026. doi: 10.1126/sciadv.aed0583.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, Vishal Dey, Mingyi Xue, Frazier N. Baker, Benjamin Burns, Daniel Adu-Ampratwum, Xuhui Huang, Xia Ning, Song Gao, Yu Su, and Huan Sun. ScienceAgentBench: Toward Rigorous Assessment of Language Agents for Data-Driven Scientific Discovery. In *International Conference on Learning Representations*, pp. 96934–96990, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/f12b4df26344f3be803c06b555252efe-Paper-Conference.pdf.

Paolo de Blasio, Jonas Elsborg, Tejs Vegge, Eibar Flores, and Arghya Bhowmik. CALiSol-23: Experimental electrolyte conductivity data for various Li-salts and solvent combinations. *Scientific Data*, 11(1):750, July 2024. ISSN 2052-4463. doi: 10.1038/s41597-024-03575-8. URL http://dx.doi.org/10.1038/s41597-024-03575-8.

A. Deschamps and Y. Brechet. Influence of predeformation and ageing of an Al–Zn–Mg alloy—II. Modeling of precipitation kinetics and yield stress. *Acta Materialia*, 47(1):293–305, 1998. ISSN 1359-6454. doi: 10.1016/s1359-6454(98)00296-1. URL http://dx.doi.org/10.1016/S1359-6454(98)00296-1.

Haonan Duan, Stephen Zhewen Lu, Caitlin Fiona Harrigan, Nishkrit Desai, Jiarui Lu, Michał Koziarski, Leonardo Cotta, and Chris J. Maddison. Measuring Scientific Capabilities of Language Models with a Systems Biology Dry Lab. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-3733. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/a23760ba51036d530a2656c3835f826c-Paper-Datasets_and_Benchmarks_Track.pdf.

Alexander Dunn, Qi Wang, Alex Ganose, Daniel Dopp, and Anubhav Jain. Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Computational Materials*, 6(1):138, Sept 2020. ISSN 2057-3960. doi: 10.1038/s41524-020-00406-3. URL http://dx.doi.org/10.1038/s41524-020-00406-3.

Kobi C. Felton, Jan G. Rittig, and Alexei A. Lapkin. Summit: Benchmarking Machine Learning Methods for Reaction Optimisation. *Chemistry–Methods*, 1(2):116–122, 2021. ISSN 2628-9725. doi: 10.1002/cmtd.202000051. URL http://dx.doi.org/10.1002/cmtd.202000051.

Paul J. Flory. Molecular Size Distribution in Three Dimensional Polymers. I. Gelation. *Journal of the American Chemical Society*, 63(11):3083–3090, 1941. doi: 10.1021/ja01856a061.

Thomas G. Fox and Paul J. Flory. Second-Order Transition Temperatures and Related Properties of Polystyrene. I. Influence of Molecular Weight. *Journal of Applied Physics*, 21(6):581–591, 1950. doi: 10.1063/1.1699711.

Kanishk Gandhi, Michael Y. Li, Lyle Goodyear, Agam Bhatia, Louise Li, Aditi Bhaskar, Mohammed Zaman, and Noah D. Goodman. BoxingGym: Benchmarking Progress in Automated Experimental Design and Model Discovery, 2025. URL https://arxiv.org/abs/2501.01540.

Wenhao Gao, Tianfan Fu, Jimeng Sun, and Connor W. Coley. Sample Efficiency Matters: A Benchmark for Practical Molecular Optimization. In *Advances in Neural Information Processing Systems*, volume 35, pp. 21342–21357, 2022. doi: 10.52202/068431-1551. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/8644353f7d307baaf29bc1e56fe8e0ec-Paper-Datasets_and_Benchmarks.pdf.

R.G. Gordon, D. Hausmann, E. Kim, and J. Shepard. A Kinetic Model for Step Coverage by Atomic Layer Deposition in Narrow Holes or Trenches. *Chemical Vapor Deposition*, 9(2):73–78, 2003. ISSN 1521-3862. doi: 10.1002/cvde.200390005. URL http://dx.doi.org/10.1002/cvde.200390005.

Prashant Govindarajan, Mathieu Reymond, Antoine Clavaud, Mariano Phielipp, Santiago Miret, and Sarath Chandar. CrystalGym: A New Benchmark for Materials Discovery Using Reinforcement Learning, 2025. URL https://arxiv.org/abs/2509.23156.

Rushil Gupta, Jason Hartford, and Bang Liu. LLMs for Bayesian Optimization in Scientific Domains: Are We There Yet? In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 15482–15510, 2025. doi: 10.18653/v1/2025.findings-emnlp.838. URL https://aclanthology.org/2025.findings-emnlp.838/.

Florian Häse, Matteo Aldeghi, Riley J Hickman, Loïc M Roch, Melodie Christensen, Elena Liles, Jason E Hein, and Alán Aspuru-Guzik. Olympus: a benchmarking framework for noisy optimization and experiment planning. *Machine Learning: Science and Technology*, 2(3):035021, July 2021. ISSN 2632-2153. doi: 10.1088/2632-2153/abedc8. URL http://dx.doi.org/10.1088/2632-2153/abedc8.

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, July 2013. ISSN 2166-532X. doi: 10.1063/1.4812323. URL http://dx.doi.org/10.1063/1.4812323.

Peter Jansen, Marc-Alexandre Côté, Tushar Khot, Erin Bransom, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Oyvind Tafjord, and Peter Clark. DiscoveryWorld: A Virtual Environment for Developing and Evaluating Automated Scientific Discovery Agents. In *Advances in Neural Information Processing Systems*, volume 37, pp. 10088–10116, 2024. doi: 10.52202/079017-0324. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/13836f251823945316ae067350a5c366-Paper-Datasets_and_Benchmarks_Track.pdf.

Heegyu Kim, Taeyang Jeon, Seungtaek Choi, Ji Hoon Hong, Dong Won Jeon, Ga-Yeon Baek, Gyeong-Won Kwak, Dong-Hee Lee, Jisu Bae, Chihoon Lee, Yunseo Kim, Seon-Jin Choi, Jin-Seong Park, Sung Beom Cho, and Hyunsouk Cho. Towards Fully-Automated Materials Discovery via Large-Scale Synthesis Dataset and Expert-Level LLM-as-a-Judge, 2025. URL https://arxiv.org/abs/2502.16457.

A. Gilad Kusne and Austin McDannald. Scalable multi-agent lab framework for lab optimization. *Matter*, 6(6):1880–1893, 2023. doi: 10.1016/j.matt.2023.03.022.

A. Gilad Kusne and Austin McDannald. Managing autonomous materials labs with multi-agent AI and its implications for the science of science. *Communications Materials*, 7:173, 2026. doi: 10.1038/s43246-026-01219-5.

Siyu Liu, Bo Hu, Beilin Ye, Jiamin Xu, David J. Srolovitz, and Tongqi Wen. MatTools: Benchmarking Large Language Models for Materials Science Tools, 2025. URL https://arxiv.org/abs/2505.10852.

Tianyu Liu, Allen Xin Wang, Antonia Panescu, Lisa Xinyi Chen, Wenxin Long, et al. Benchmarking AI Agents for Addressing Scientific Challenges Across Scales, 2026. URL https://arxiv.org/abs/2606.12736.

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(5):525–535, 2024. ISSN 2522-5839. doi: 10.1038/s42256-024-00832-8. URL http://dx.doi.org/10.1038/s42256-024-00832-8.

Shreshth A Malik, Tiarnan Doherty, Panagiotis Tigas, Muhammed Razzak, Stephen J. Roberts, Aron Walsh, and Yarin Gal. MADE: Benchmark Environments for Closed-Loop Materials Discovery, 2026. URL https://arxiv.org/abs/2601.20996.

Adrian Mirza, Nawaf Alampara, Sreekanth Kunchapu, Martiño Ríos-García, Benedict Emoekabu, Aswanth Krishnan, Tanya Gupta, Mara Schilling-Wilhelmi, Macjonathan Okereke, Anagha Aneesh, Mehrdad Asgari, Juliane Eberhardt, Amir Mohammad Elahi, Hani M. Elbeheiry, María Victoria Gil, Christina Glaubitz, Maximilian Greiner, Caroline T. Holick, Tim Hoffmann, Abdelrahman Ibrahim, Lea C. Klepsch, Yannik Köster, Fabian Alexander Kreth, Jakob Meyer, Santiago Miret, Jan Matthias Peschel, Michael Ringleb, Nicole C. Roesner, Johanna Schreiber, Ulrich S. Schubert, Leanne M. Stafast, A. D. Dinga Wonanke, Michael Pieler, Philippe Schwaller, and Kevin Maik Jablonka. A framework for evaluating the chemical knowledge and reasoning abilities of large language models against the expertise of chemists. *Nature Chemistry*, 17(7):1027–1034, 2025. ISSN 1755-4349. doi: 10.1038/s41557-025-01815-x. URL http://dx.doi.org/10.1038/s41557-025-01815-x.

AkshatKumar Nigam, Robert Pollice, Gary Tom, Kjell Jorner, John Willes, Luca A. Thiede, Anshul Kundaje, and Alan Aspuru-Guzik. Tartarus: A Benchmarking Platform for Realistic And Practical Inverse Molecular Design. In *Advances in Neural Information Processing Systems*, volume 36, pp. 3263–3306, 2023. doi: 10.52202/075280-0145. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/09f8b2469a3d1089a7c60d9ef1983271-Paper-Datasets_and_Benchmarks.pdf.

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.

Richard Otis and Zi-Kui Liu. pycalphad: CALPHAD-based Computational Thermodynamics in Python. *Journal of Open Research Software*, 5(1):1, 2017. ISSN 2049-9647. doi: 10.5334/jors.140. URL http://dx.doi.org/10.5334/jors.140.

Noah H. Paulson, Angel Yanguas-Gil, Osama Y. Abuomar, and Jeffrey W. Elam. Intelligent Agents for the Optimization of Atomic Layer Deposition. *ACS Applied Materials & Interfaces*, 13(14):17022–17033, 2021. ISSN 1944-8252. doi: 10.1021/acsami.1c00649. URL http://dx.doi.org/10.1021/acsami.1c00649.

Raghunathan Ramakrishnan, Pavlo O. Dral, Matthias Rupp, and O. Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1(1):140022, 2014. ISSN 2052-4463. doi: 10.1038/sdata.2014.22. URL http://dx.doi.org/10.1038/sdata.2014.22.

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(6):836–847, June 2025. ISSN 2522-5839. doi: 10.1038/s42256-025-01055-1. URL http://dx.doi.org/10.1038/s42256-025-01055-1.

H.R. Shercliff and M.F. Ashby. A process model for age hardening of aluminium alloys—I. The model. *Acta Metallurgica et Materialia*, 38(10):1789–1802, 1990. ISSN 0956-7151. doi: 10.1016/0956-7151(90)90291-n. URL http://dx.doi.org/10.1016/0956-7151(90)90291-N.

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, pp. 8634–8652, 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1b44b878bb782e6954cd888628510e90-Paper-Conference.pdf.

Zachary S. Siegel, Sayash Kapoor, Nitya Nadgir, Benedikt Stroebl, and Arvind Narayanan. CORE-Bench: Fostering the Credibility of Published Research Through a Computational Reproducibility Agent Benchmark. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=BsMMc4MEGS.

Walter H. Stockmayer. Theory of Molecular Size Distribution and Gel Formation in Branched-Chain Polymers. *The Journal of Chemical Physics*, 11(2):45–55, 1943. doi: 10.1063/1.1723803.

Hunghai Su and D. Lynn Johnson. Master Sintering Curve: A Practical Approach to Sintering. *Journal of the American Ceramic Society*, 79(12):3211–3217, 1996. ISSN 1551-2916. doi: 10.1111/j.1151-2916.1996.tb08097.x. URL http://dx.doi.org/10.1111/j.1151-2916.1996.tb08097.x.

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 (7990):86–91, 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06734-w. URL http://dx.doi.org/10.1038/s41586-023-06734-w.

Félix Therrien, Jamal Abou Haibeh, Divya Sharma, Rhiannon Hendley, Leah Wairimu Mungai, Sun Sun, Alain Tchagang, Jiang Su, Samuel Huberman, Yoshua Bengio, Hongyu Guo, Alex Hernández-García, and Homin Shin. OBELiX: a curated dataset of crystal structures and experimentally measured ionic conductivities for lithium solid-state electrolytes. *Digital Discovery*, 5(2):910–918, 2026. doi: 10.1039/d5dd00441a. URL https://doi.org/10.1039/d5dd00441a.

Gilles Wainrib, Barbara Bodinier, Haitem Dakhli, Josep Monserrat, Almudena Espin Perez, Sabrina Carpentier, Roberta Codato, and John Klein. Can AI Scientist Agents Learn from Lab-in-the-Loop Feedback? Evidence from Iterative Perturbation Discovery, 2026. URL https://arxiv.org/abs/2603.26177.

Andreas Werbrouck, Marshall B. Lindsay, Matthew Maschmann, and Matthias J. Young. LLM Agents for Knowledge Discovery in Atomic Layer Processing, 2025. URL https://arxiv.org/abs/2509.26201. Presented at the AI for Accelerated Materials Design (AI4MAT) Workshop at NeurIPS 2025.

Angel Yanguas-Gil. Performance of AI agents based on reasoning language models on ALD process optimization tasks. *Journal of Vacuum Science & Technology A*, 44(4):043410, 2026a. doi: 10.1116/6.0005313.

Angel Yanguas-Gil. Evaluating LLM-based AI agents integrated with materials synthesis tools: the case of atomic layer deposition, 2026b. URL https://arxiv.org/abs/2608.29309.

Angel Yanguas-Gil, Matthew T. Dearing, Jeffrey W. Elam, Jessica C. Jones, Sungjoon Kim, Adnan Mohammad, Chi Thang Nguyen, and Bratin Sengupta. Benchmarking large language models for materials synthesis: The case of atomic layer deposition. *Journal of Vacuum Science & Technology A*, 43(3):032406, 2025. ISSN 1520-8559. doi: 10.1116/6.0004319. URL http://dx.doi.org/10.1116/6.0004319.

Angel Yanguas-Gil, Jessica C. Jones, Sungjoon Kim, Chi Thang Nguyen, and Jeffrey W. Elam. Design and performance of AI agents interfacing with an atomic layer deposition tool. *Review of Scientific Instruments*, 97(5):053903, 2026. doi: 10.1063/5.0318770.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains. In *International Conference on Learning Representations*, pp. 9965–10017, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/1b126cc38b8638e07bef37e7b2bb72bf-Paper-Conference.pdf.

Wenhao Yuan, Guangyao Chen, Zhilong Wang, and Fengqi You. Empowering Generalist Material Intelligence with Large Language Models. *Advanced Materials*, 37(32):2502771, 2025. doi: 10.1002/adma.202502771.

Mohd Zaki, Jayadeva, Mausam, and N. M. Anoop Krishnan. MaScQA: investigating materials science knowledge of large language models. *Digital Discovery*, 3(2):313–327, 2024. doi: 10.1039/d3dd00188a. URL https://doi.org/10.1039/d3dd00188a.

Junkai Zhang, Jingru Gan, Xiaoxuan Wang, Zian Jia, Changquan Gu, Jianpeng Chen, Yanqiao Zhu, Mingyu Derek Ma, Dawei Zhou, Ling Li, and Wei Wang. MatSciBench: Benchmarking the Reasoning Ability of Large Language Models in Materials Science. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pp. 12868–12879, 2026. doi: 10.1145/3770855.3818888. URL https://doi.org/10.1145/3770855.3818888.

Tianshi Zheng, Kelvin Kiu-Wai Tam, Newt Hue-Nam K. Nguyen, Baixuan Xu, Zhaowei Wang, Jiayang Cheng, Hong Ting Tsang, Weiqi Wang, Jiaxin Bai, Tianqing Fang, Yangqiu Song, Ginny Y. Wong, and Simon See. NewtonBench: Benchmarking Generalizable Scientific Law Discovery in LLM Agents. In *International Conference on Learning Representations*, pp. 99018–99088, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/a0e1c2c40fc245b5fe7251ea33fbb045-Paper-Conference.pdf.

Yuanlong Bill Zheng, Connor Blake, Layla Mravac, Fengxue Zhang, Yuxin Chen, and Shuolong Yang. A self-driving physical vapor deposition system making sample-specific decisions on the fly. *npj Computational Materials*, 11:327, 2025. doi: 10.1038/s41524-025-01805-0.

A Tasks and evaluation protocol

A.1 Task specification and agent interface

The following condensed brief is drawn from task SOL-2a68b7 in the primary garnet max_v2 release. It illustrates the information given to the agent; the air-storage workflow in Figure 1 uses a different, earlier task.

> **Goal.** Deliver a lithium-garnet solid-electrolyte pellet that meets the specifications after 200 hours of simulated lithium-metal contact.
> **Specifications.** Total ionic conductivity at 25 °C ≥ 0.6513 mS/cm; electronic conductivity $\leq 10^{-4}$ mS/cm; relative density ≥ 0.92.
> **Design space.** Choose from 1,055 garnet compositions and select the preparation steps and their parameters. Public reports and screening tools support candidate selection.
> **Limits.** Sintering temperature ≤ 1300 °C; furnace time ≤ 60 hours per pellet. The campaign allows seven samples, 12 computation-cost units, 72 simulated computation-hours, 400 laboratory-hours, 30 measurement-cost units, and 300 actions.
> **Delivery.** Submit one unconsumed pellet and its composition and ordered preparation steps. The customer checks the pellet and repeats the recipe on fresh batches.

The agent receives BRIEF.md, task.json, and the laboratory client tools.py. These files expose the specifications, candidate information, budgets, and tool descriptions. They do not expose hidden process parameters or internal material states. Available actions include candidate queries, computations, sample creation, processing, measurement, and delivery. In this example, the agent can screen compositions, assess phase stability, and choose mixing, milling, calcination, pressing, and sintering conditions. A service test consumes the tested pellet, so another pellet must remain available for delivery. Simulated service exposure is not counted as laboratory waiting time; computation and measurement units denote costs, not numbers of calls.

The base prompt used in the reported runs was: "You are a materials researcher working from this directory. Read BRIEF.md and task.json, then solve the customer's problem with the tools available through tools.py and hand in your result with lab_deliver. Analyse data with Python if you need to. Do not modify tools.py. Finish by writing notes.md." Memory-transfer conditions add the instruction described in Appendix C.3.

Each sample retains its candidate, batch draw, processing history, and consumption status. Processing changes a persistent internal state; measurements can be noisy, fail, or consume the sample. The public interface returns sample identities, histories, observations, and resource use. Repeated identical computations are cached. Requests exceeding sample, computation-cost, or computation-time limits are refused; other budget violations are recorded and cause acceptance to fail. Campaign state and random-generator state can be saved and restored for deterministic continuation.

A.2 TASK GENERATION AND ADMISSION

The generator draws a hidden world and initial specifications, then searches for a recipe that remains feasible under batch variation. Let b_0 denote the nominal batch and $b_1, \ldots, b_4$ the four laboratory-specific adverse batches. Let $F(r, b)$ indicate that recipe r executes successfully and satisfies the non-target constraints on batch b, and let $f(r, b)$ be its service-conditioned figure of merit: hardness for alloy, capacitance for ALD, and ionic conductivity for the three electrolyte laboratories. The qualifying grid and its recipe score are

$$\begin{aligned} \mathcal{R}_{\mathrm{ok}} &= \left\{ r \in \mathcal{R}_{\mathrm{grid}} : F(r, b_0) = 1,\ \sum_{j=1}^{4} F(r, b_j) \geq 3 \right\}, \\ u(r) &= \min_{b \in \{b_0\} \cup \{b_j : F(r, b_j) = 1,\ 1 \leq j \leq 4\}} f(r, b). \end{aligned} \tag{3}$$

Thus, a reference may fail a non-target constraint on one adverse batch. Its score takes the minimum over the nominal batch and the adverse batches that satisfy those constraints; it does not include the failed batch as a feasible outcome.

The generator selects the best qualifying grid recipe and sets the target as

$$r_{\mathrm{grid}} \in \arg\max_{r \in \mathcal{R}_{\mathrm{ok}}} u(r), \qquad t_\ell = \mathrm{round}_\ell(\alpha_\ell u(r_{\mathrm{grid}})), \tag{4}$$

where round_ℓ uses the laboratory's reported precision. In the order alloy, garnet, liquid, polymer, and ALD, the fractions α_ℓ are 0.985, 0.93, 0.95, 0.93, and 0.94. Empty qualifying grids and non-positive reference scores are rejected. Local refinement and adversarial search can improve the executable reference recipe and recorded audit ceiling, but do not raise the grid-based target.

Algorithm 1 Task generation and admission

1: Draw hidden world θ, initial specifications, and budgets from task seed z
2: Construct reference grid $\mathcal{R}_{\mathrm{grid}}$ and batches $b_0, \ldots, b_4$
3: Compute $\mathcal{R}_{\mathrm{ok}}$ and $u(r)$ using Eq. (3); reject if no positive qualifying score exists
4: Select r_{grid} and set target t_ℓ using Eq. (4)
5: Refine reference recipes and search adversarial alternatives; retain target t_ℓ
6: Reject if a feasible adversarial recipe improves the grid score by more than 20%
7: Form task $\tau = (\ell, g, \mathcal{C}, \mathbf{b}, \theta)$ with target t_ℓ included in g
8: Evaluate privileged and open-loop controllers at seeds $e_k = z + 5 + 7919k$, $k \in \{0, 1, 2\}$
9: Release iff $\sum_{k=0}^{2} A_\tau(\pi_{\mathrm{priv}}, e_k) = 3$ and $\sum_{k=0}^{2} A_\tau(\pi_{\mathrm{open}}, e_k) = 0$

Here $A_\tau(\pi, e)$ is the final acceptance indicator in Eq. (2) for controller π and laboratory seed e. The gates select executable tasks that the tested fixed handbook recipe fails on the admission seeds.

The primary maximum-library tasks and earlier 250-task release share this generation and admission procedure, but use different candidate libraries, task specifications, budgets, and reference grids. The earlier release searches 256–2,240 recipes per task; maximum-library searches use laboratory-specific screening and process grids. Reference refinement defaults to three starting recipes and 60 evaluations per start; adversarial search defaults to 150 evaluations. These settings are implementation defaults, and the released task worlds and generation records identify the frozen references used for evaluation. The reference is the best executable solution found by the configured search. Delivered performance more than 10% above its recorded favorable-batch ceiling triggers inspection without changing acceptance. Simulator checks and resulting release corrections are described in Appendix B.7; evaluation cohorts are identified in Appendix C.2.

A.3 ACCEPTANCE AND FRESH-BATCH REPLICATION

Evaluation applies the following checks in the order shown in Eq. (2). First, the submission must contain an unconsumed sample and a valid recipe, and the original campaign must satisfy every resource limit. Second, the delivered sample must satisfy all specified properties under the prescribed service conditions. Third, the submitted candidate and ordered process steps must match the sample's preparation history. Finally, the recipe must produce acceptable samples on at least two of three fresh batches. Acceptance uses customer-side simulated metrics, rather than noisy readings purchased by the agent.

History matching compares system-validated steps: defaults are filled, numeric arguments converted or clipped to allowed ranges, and formulation fractions normalized. Equivalent numeric notation can therefore match the same step. Split, merged, or reordered operations are not treated as equivalent, even if their modeled physical outcomes agree. Matching measures agreement with the recorded preparation history; a mismatch does not by itself identify a transcription error or establish physical inequivalence. A recipe contains at most eight steps. Invalid recipes are refused and can be corrected before delivery; a valid recipe with a mismatched history is accepted as a submission but fails evaluation.

The evaluator executes the recipe in three new campaigns with seeds $e + 1000 + i$, $i \in \{0, 1, 2\}$, where e is the original campaign's laboratory seed. The hidden world and specifications remain fixed, while each campaign draws a new batch. Replication receives sufficient execution budgets to test the material outcome; budget compliance is assessed on the original agent campaign. A failed recipe execution counts as a failed batch, and every successful execution is checked against the same service requirements as the delivered sample.

All reported acceptance results use the final protocol, including strict recipe-history matching, sample acceptance, budget compliance, and fresh-batch replication. The same matching rules apply across methods and experimental conditions. Scoring evaluates the resulting samples and recipes; it does not require an explicit model of the hidden mechanisms or assess similarity to human reasoning.

B Simulated laboratories

B.1 Shared construction

Each laboratory supports exploration of partially known process responses by connecting agent-controlled preparation to persistent sample states, measurable properties, and service performance. The implementation separates world generation, batch draws, step validation, processing, observation, public computation, and acceptance. A process updates the state; measurements and customer-side checks read that same state under their specified conditions. Existing software and datasets provide equilibrium tables, conductivity fits, or molecular descriptors. We implement the process dynamics and their coupling to material properties (Table 4).

Three sources of uncertainty have different roles. Hidden world parameters are drawn once per task and remain fixed across its attempts. They define task-specific process responses or candidate interactions that are unknown to the agent but fixed during exploration. Observations provide evidence about these relationships. Each sample draws batch variables, such as furnace offset, chemical assay, or inhibitor level, which change the outcome of the same recipe. Measurement noise is added when the agent purchases an observation. Public calculations return nominal predictions, fitted estimates, or source records; they do not reveal the full sample state. Thus, screening can narrow a search, but preparing and measuring samples can expose effects omitted from public predictions.

Models retain tradeoffs, sample histories, and resource costs relevant to planning experiments and revising preparation routes. Simplified kinetics support repeated interaction. Most updates are closed-form; alloy aging and polymerization use numerical substeps. Saved worlds and model tables specify the parameters for replay. The sections below identify the data-supported property layers, literature-informed mechanisms, and implemented process rules. Results describe performance in these simulated laboratories; experimental validation of a recipe requires physical synthesis and measurement.

B.2 Atomic layer deposition

The agent chooses a material–precursor combination and controls temperature, precursor and co-reactant exposure, purge durations, cycle count, and annealing. The sample tracks top and trench-bottom thickness, carbon contamination, parasitic growth, annealing temperature, and process time. For normalized exposures d_1, d_2, the saturation and penetration factors are

$$\chi = (1 - e^{-d_1})(1 - e^{-d_2}), \qquad \eta = \min\left(1, \sqrt{\frac{\min(d_1, d_2)}{3g(a)}}\right), \tag{5}$$

Table 4: Data and computational support for the laboratories. ✓: used; –: not used. Process-model references are given in B.2–B.6.

Laboratory	Property fit	Scientific inputs	Measured reports	Sources
ALD	–	–	–	–
Alloy	–	✓	–	pycalphad / MatCalc
Garnet	–	✓	✓	Materials Project / OBELiX
Liquid	✓	✓	–	CALiSol-23 / QM9
Polymer	✓	–	–	PEO-electrolyte data

Property fits calibrate conductivity. Scientific inputs are equilibrium/phase-stability calculations or molecular descriptors; measured reports provide candidate conductivity records. Generated reports are not counted as additional measured data.

where a is trench aspect ratio and $g(a)$ is the implemented exposure factor. The simplified penetration rule is informed by the trench-coverage model of Gordon et al. (2003). Exposures are normalized by chemistry-specific saturation times, with source strength and batch dose variation applied to the precursor. Temperature sets growth per cycle, while insufficient purge and precursor decomposition add parasitic growth. These effects can increase top thickness without achieving adequate trench-bottom coverage. Contamination is thickness-weighted when deposition stages are combined.

Capacitance depends on dielectric constant, thickness, and coverage; leakage depends on contamination, crystallization, thin regions, and incomplete coverage. Annealing reduces contamination and can improve electrical properties, but exceeding a hidden crystallization temperature increases leakage. Each anneal also incurs a 20-minute ramp cost. The agent therefore balances exposure and purge time against coverage, film quality, and the process-time limit. Flat-pad measurements can miss poor trench coverage; trench measurements help distinguish these outcomes. Batch variables perturb dose and temperature, and screening tools provide imperfect estimates of chemistry-specific windows. The ALD process parameters and electrical penalties use literature-scale values and implemented rules.

B.3 Al–Mg–Si alloy

The agent selects a composition and controls solution treatment, quench medium, room-temperature holding, and artificial aging. Stored pycalphad calculations using the MatCalc aluminum database provide equilibrium Mg_2Si and liquid fractions (Otis & Liu, 2017). They constrain solute dissolution and incipient melting. Our kinetic layer tracks dissolved solute, precipitate fraction f, precipitate radius r, coarse material, clusters, and furnace time, following the structure of precipitation-strengthening models (Shercliff & Ashby, 1990; Deschamps & Brechet, 1998).

For an aging substep Δt, precipitation and cubic coarsening use

$$\Delta f = (f_{\mathrm{eq}} - f)(1 - e^{-k_p \Delta t}), \qquad r_{\mathrm{new}}^3 = r^3 + K\Delta t, \tag{6}$$

where f_{eq} is the metastable equilibrium fraction and k_p and K are temperature-dependent Arrhenius rates. Aging uses 8–400 substeps depending on duration, with time expressed in hours and temperature in kelvin in the rate calculations. Hardness combines solid-solution strengthening with the smaller of precipitate-shearing and bypass contributions. Resistivity increases with dissolved solute and very fine precipitates. The same state therefore determines both hardness and conductivity, while quench history and natural aging change the later precipitation response.

Aging can raise hardness toward a peak and then reduce it through coarsening. Hot service applies further aging, so the best as-made hardness need not give the best service-conditioned combination of properties. Equilibrium calculations help locate a treatment window, while hardness, conductivity, and service measurements constrain the unknown kinetic response. Batch draws perturb solute content and furnace temperature. CALPHAD supplies the equilibrium layer; literature-informed kinetics determine hardness evolution.

B.4 GARNET SOLID ELECTROLYTE

The agent chooses a garnet composition and controls lithium excess, source materials, milling, calcination, pressing, and sintering. Samples progress through powder, green-pellet, and sintered stages. Milling reduces particle size but can add contamination. Calcination advances precursor conversion through first-order kinetics with an inverse-square particle-size factor, while lithium retention decreases through an Arrhenius loss rate. A mother-powder bed reduces that loss.

Sintering accumulates thermal work and computes relative density using a master-sintering-curve construction (Su & Johnson, 1996):

$$\begin{aligned} \Theta &= \sum_j \frac{t_j}{T_j} e^{-Q_s/(RT_j)} \left[\max(d_j, 0.05)\right]^{-0.6}, \\ \rho &= \rho_g + (\rho_{\max} - \rho_g) \left[1 + e^{-(\ln\Theta - \ln\Theta_0)/w_\rho}\right]^{-1}. \end{aligned} \tag{7}$$

Here t_j is stage duration in seconds, T_j is batch-adjusted temperature in kelvin, and d_j is the implemented particle-size value in micrometers. Q_s is the sintering activation energy, R is the gas constant, ρ_g is green density, and Θ_0, w_ρ set the density transition. The particle-size factor and density penalties are our model extensions. The ceiling $\rho_{\max}$ decreases with trapped carbonate and lithium deficit. Grain size follows cubic growth; abnormal growth is a persistent state change. Accumulating thermal work avoids repeatedly granting a fresh densification gain when a sinter is split into stages.

Higher temperature or longer sintering can improve density while increasing lithium loss and grain growth. Conductivity combines bulk and grain-boundary resistances with a density factor; lithium deficit and vacancy concentration alter the conducting phases. Air-storage effects depend on open porosity, while lithium-contact effects depend on reducibility, reactive boundaries, and connected pores. Phase-stability calculations, diffraction, density, and impedance measurements offer different evidence about these hidden states. Materials Project and pymatgen support phase analysis, and OBELiX supplies public conductivity reports (Jain et al., 2013; Ong et al., 2013; Therrien et al., 2026). Extended compositions use dopant and defect rules with hidden candidate variation. The process and service layers use the rules described above.

B.5 LIQUID ELECTROLYTE

The agent chooses a salt, solvent weight fractions, molality, additive, and drying duration. The conductivity layer fits a composition- and temperature-dependent Casteel–Amis form (Casteel & Amis, 1972) to CALiSol-23 measurements (de Blasio et al., 2024):

$$\ln\kappa = \ln\kappa_{\max}(\mathbf{w}, T) + a_c \ln(c/\mu) - a_c(c-\mu)/\mu + b_c(c-\mu)^2, \tag{8}$$

$$\ln\kappa_{\max} = \sum_i w_i A_i + \sum_{i<j} w_i w_j B_{ij} - E(\mathbf{w})(10^3/T - 10^3/298.15). \tag{9}$$

Here κ is expressed numerically in mS/cm, c and the peak concentration μ in mol/kg, T in kelvin, and w_i as normalized solvent weight fractions. The peak concentration varies with composition and temperature and is clipped to 0.2–4.0 mol/kg. Fits use robust least squares with a soft-L1 loss and regularization on solvent-pair coefficients. Cyclic–linear carbonate pairs share a regularized interaction tendency; other pair coefficients shrink toward zero. Maximum-library worlds add hidden solvent-pair, salt–solvent, and peak-concentration effects, so the public fit is informative but incomplete.

Drying approaches a hidden water floor. Blending consumes water and produces salt-dependent acid or insoluble products. Additives can improve the modeled interphase while lowering conductivity. Ideal-solution freezing estimates constrain cold operation; service rules check freezing, thermal acid generation, and graphite retention. High conductivity alone therefore does not identify an acceptable formulation. Water, acid, conductivity, and service measurements help separate preparation quality from nominal composition. The process and service layers use the implemented drying, reaction, and interphase rules. QM9 supplies gas-phase frontier-orbital descriptors for covered molecules (Ramakrishnan et al., 2014).

Calibration retains records with conductivity above 0.01 mS/cm, concentration above 0.05 mol/kg, solvent weight fractions, supported solvents, and temperatures between 240 and 335 K. The resulting salt-specific coverage and residuals appear in Table 5. The reported residuals are computed on the fitting records.

B.6 Polymer electrolyte

The agent selects a monomer design and controls inhibitor removal, initiator loading, temperature, duration, solvent fraction, chain transfer, crosslinking, purification, salt ratio, and film casting/drying. Polymerization uses 500 substeps under a radical quasi-steady-state approximation. Initiator decomposition, inhibitor consumption, propagation, conversion-dependent termination, and chain transfer update conversion and dead-chain moments μ_0, μ_1, μ_2. Before the crosslink correction, the molecular weights are $M_n = M_0\mu_1/\mu_0$ and $M_w = M_0\mu_2/\mu_1$, where M_0 is monomer molar mass. A simplified Flory–Stockmayer-inspired criterion marks gelled material that cannot be cast (Flory, 1941; Stockmayer, 1943).

Purification lowers residual monomer and slightly increases M_n; casting sets salt concentration and residual solvent. A Fox–Flory-type dependence links molecular weight to glass-transition temperature (Fox & Flory, 1950). Conductivity uses a salt-specific Vogel–Tammann–Fulcher form fitted to measured PEO-family electrolytes (Bradford et al., 2023):

$$\log_{10}\sigma = A + p\log_{10} m - qm - \frac{B}{\max(T - T_{00} - t_1 m - \Delta T_g, 25\,\mathrm{K})}. \tag{10}$$

Here σ is expressed numerically in S/cm, m is salt molality in mol/kg polymer, and $T, T_{00}, \Delta T_g$ are in kelvin. ΔT_g is the salt-free modeled glass-transition temperature relative to $-60\,^\circ$C, accounting for monomer design, molecular weight, and plasticization; the fitted $t_1 m$ term separately accounts for salt concentration. Hidden architecture and salt factors further scale conductivity. The modulus depends on molecular-weight-dependent entanglement, salt, residuals, temperature, and crystallinity. Residual monomer and solvent can improve conductivity while weakening the film; residual monomer also penalizes modeled lithium-cycling performance. Molecular-weight, thermal, residual-content, electrical, and mechanical measurements therefore provide complementary feedback.

Calibration selects PEO-family, single-salt records without a second solvent, second salt, or inorganic filler, using 0.1–5 mol/kg, approximately 60–127 °C, and $\log_{10}\sigma$ between -8 and -1.5. Robust least squares fits the amorphous-temperature regime. Table 5 reports residuals on these fitting records. Predictions at lower task temperatures and for generated monomer architectures are model extensions. Polymerization kinetics, modulus, and cycling penalties use the separate rules described above.

B.7 Calibration and implementation checks

We distinguish calibration agreement from checks of mechanisms, numerical implementation, and interaction rules. Table 5 measures agreement with the records used to fit conductivity. Trend tests check qualitative responses, such as overaging, lithium loss, and ALD saturation. A separate stochastic polymer engine provides a cross-implementation comparison under a specified common kinetic setting. Replay, budget, sample-consumption, and extreme-input tests check the environment protocol. Table 6 lists representative assertions for the selected test conditions.

The splitting test compares selected recipes with consecutive half-duration operations, using relative tolerances of 3×10^{-3} for alloy and 10^{-6} for the other laboratories, with absolute tolerance 10^{-9}. It skips operations that cannot be repeated and samples task families outside the largest-pool families; some laboratories have no applicable time-like operation in the sampled grid. Fixed overheads, such as the ALD anneal ramp, remain charged per operation, so physical equivalence need not preserve resource use. Extreme-input checks accept either finite outputs or explicit refusals.

Earlier garnet runs exposed a missing post-calcination milling route in the reference grid and a staged-sinter defect. Reference recipes were expanded and density updates changed to cumulative thermal work before the corrected releases reported here. Liquid max_v2 audit ceilings include the corrected reference search. These audits led to revised reference searches and cumulative process updates in the reported releases.

Table 5: In-sample conductivity calibration residuals and data coverage. Liquid errors use natural-log conductivity; polymer errors use $\log_{10}$ conductivity.

Environment	Salt	Fit rows	Fit RMSE	Median abs. residual	Calibration range
Liquid	LiPF6	2968	0.162	4.8% rel.	0.20–2.37 mol/kg, 243–333 K
Liquid	LiBF4	2620	0.403	5.8% rel.	0.20–2.37 mol/kg, 244–333 K
Liquid	LiBOB	2961	0.246	5.5% rel.	0.05–1.12 mol/kg, 244–332 K
Polymer	LiTFSI	369	0.313	0.053 dex	0.1–5 mol/kg, 60–127 °C
Polymer	LiClO4	138	0.441	0.151 dex	0.1–5 mol/kg, 60–127 °C
Polymer	LiBF4	115	0.719	0.120 dex	0.1–5 mol/kg, 60–127 °C

Table 6: Implemented simulator test criteria under selected conditions.

Laboratory	Check	Criterion
ALD	Dose saturation	$\lvert g_{\mathrm{high}}/g_{\mathrm{sat}} - 1\rvert < 0.02$
	Trench penetration	$g_{\mathrm{bottom}}/g_{\mathrm{top}} < 0.5$
Alloy	Aging peak	$H_{\mathrm{mid}} > \max(H_{\mathrm{short}}, H_{\mathrm{long}})$
Garnet	Lithium retention	$\kappa_{\mathrm{covered}}/\kappa_{\mathrm{bare}} > 3$
	Undoped conductivity	$\kappa_{\mathrm{undoped}} < 0.02\,\mathrm{mS/cm}$
Liquid	Conductivity scale	$9 < \kappa < 14\,\mathrm{mS/cm}$
	Concentration peak	$\kappa_{\mathrm{mid}} > \max(\kappa_{\mathrm{low}}, \kappa_{\mathrm{high}})$
Polymer	Conversion: moments vs. KMC	$\lvert X_{\mathrm{mom}} - X_{\mathrm{KMC}}\rvert < 0.03$
	Molecular weight: moments vs. KMC	$\lvert M_{n,\mathrm{mom}}/M_{n,\mathrm{KMC}} - 1\rvert < 0.15$

g: growth per cycle; H: hardness; κ: ionic conductivity; X: conversion; M_n: number-average molecular weight. ALD saturation compares 10 s and 1 s precursor doses; alloy aging compares 4 h with 0.5 h and 200 h; liquid concentration compares 1.0 with 0.3 and 2.0 mol/kg. Polymer criteria compare the moment model with the stochastic engine in one common kinetic setting.

B.8 Data sources and licenses

CALiSol-23 is CC BY 4.0 and supplies 8,549 selected conductivity records for the liquid fits (de Blasio et al., 2024). The polymer-electrolyte repository is MIT-licensed; 622 selected PEO-family single-salt records cover the three salts used by the laboratory (Bradford et al., 2023). Selection rules and coverage are described above, and the calibration tables are frozen for replay.

Materials Project (CC BY 4.0, snapshot 2026.04.13) supplies stored Li–La–Zr–O–C phase entries; OBELiX (CC BY 4.0) supplies 58 garnet conductivity records (Jain et al., 2013; Therrien et al., 2026). MatCalc mc_al v2.037 is ODbL and supplies the cleaned aluminum thermodynamic database used for equilibrium tables. QM9 is CC0 and supplies the covered molecular descriptors (Ramakrishnan et al., 2014). Reports for generated candidates use the rules described for each laboratory; measured source records are identified separately. Frozen tables and task worlds remove live database dependencies. Reconstructing a reported run also requires the corresponding code and dependencies, recipe grids, fit coefficients, prompts and memory files, agent settings, seeds, and action trace, linked to its score and table or figure inputs. Upon acceptance, we will release the benchmark data and code with applicable source attributions and licenses.

C Baselines and evaluation settings

C.1 Scripted strategies

Section 4.2 describes the six non-privileged strategies. Random search and Bayesian optimization select complete recipes from a fixed public pool, reserve resources for delivery, and remake the best observed recipe. They do not propose new process sequences or optimize arguments outside that pool; agents choose actions and arguments through the laboratory interface. Their observation score combines target-normalized performance with penalties for unmet constraints. Random search samples without replacement. Bayesian optimization encodes candidates, process arguments, and public descriptors; it standardizes nonconstant features and fits a Gaussian process to standardized

scores. The squared-exponential kernel has lengthscale $0.6\sqrt{D}$ for feature dimension D, with diagonal noise 0.05. Three initial choices are random; later trials maximize expected improvement. Fixed strategies use laboratory-specific controllers; the ALD compute/prior controller additionally calibrates thickness on a flat pad. The privileged controller executes the hidden reference recipe.

Shuffled feedback returns a channel reading from an unrelated sample generated from the public recipe pool. The strategy's actual sample outcomes, measurement costs, and destructive effects remain unchanged. The generator tries up to six alternative recipes and falls back to the actual reading if none yields a valid observation.

C.2 Agent configuration and evaluation cohorts

Table 7 gives CLI settings, and Table 8 summarizes evaluation sizes and limits. Primary alloy, garnet, and liquid tasks use max_v2; polymer and ALD use max_v1. Scripted rates come from the released baseline matrices. Model configurations are specified in Section 4: auxiliary agent comparisons and memory transfer retain the earlier Codex configuration. Provider-interrupted runs are excluded and replaced. All reported outcomes use the final acceptance protocol in Appendix A, including strict recipe-history matching. One primary Codex ALD run retains a reference-ceiling audit flag, which does not alter acceptance.

Table 7: CLI settings for the agent evaluations.

Setting	Configuration
CLI	Codex 0.149.1, `exec`; Claude Code 2.1, `-p`
Access	Shell/file tools; no web search or sub-agents
Session	Ephemeral; user configuration ignored; approval policy `never`
Records	Agent events, notes, campaign ledger, evaluation score
Isolation	Public laboratory gateway; hidden task parameters and evaluator data isolated from agent access

Table 8: Evaluation cohorts. Repeats are per task and condition; time limits apply to agent attempts.

Evaluation	Tasks	Repeats	Limit (s)
Primary agents	52	3	3,600
Primary scripted strategies	52	3	–
Alloy feedback controls	10	3	–
Auxiliary scripted strategies	250	5	–
Pilot agents	25	3 / 1	3,000
Pilot tool/identity controls	25	1	3,000
Library comparison: small	20	3 / 1	3,000
Library comparison: maximum	40	3	3,600
Polymer memory conditions	10	3	3,600
ALD memory conditions	12	3	3,600

"3 / 1" denotes Codex / Claude Code repeats. The pilot covers five tasks per laboratory; library comparisons cover alloy, garnet, polymer, and ALD, with five or ten tasks each. Primary tasks include twelve liquid and ten tasks in each other laboratory. Resource budgets are specified in the released task files.

The auxiliary scripted results in Table 9 use the earlier small-library release and are separate from the primary comparison in Table 3.

C.3 Memory-transfer evaluation

The polymer comparison uses max_v1 tasks with no memory or with ALD, liquid-electrolyte, or alloy memory. The no-memory condition reuses the existing polymer cohort. The ALD comparison uses max_v2 with or without ALD max_v1 memory. ALD max_v2 retains 22 chemistries but changes the candidate-library version and hidden task worlds. Each target comparison shares task IDs and run indices; cohort sizes are in Table 8.

Each source memory contains up to eight successful Claude Code campaigns, keeping the shortest ledger per task. Source campaigns pass the final acceptance protocol, including strict recipe-history

Table 9: Auxiliary small-library acceptance: 50 tasks per laboratory, five seeds per task. Values are proportions; parentheses show shuffled feedback.

	Alloy	Garnet	Liquid	Polymer	ALD
Open loop	0.01	0.00	0.00	0.00	0.00
Compute / prior data only	0.22 (0.22)	0.01 (0.01)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
Single-metric feedback	0.23 (0.18)	0.04 (0.01)	0.29 (0.10)	0.00 (0.00)	0.00 (0.00)
Test-and-switch	0.30 (0.12)	0.10 (0.02)	0.13 (0.00)	0.14 (0.00)	0.03 (0.01)
Random search	0.20 (0.04)	0.08 (0.02)	0.06 (0.02)	0.15 (0.04)	0.10 (0.02)
Bayesian optimisation	0.18 (0.04)	0.08 (0.02)	0.05 (0.01)	0.15 (0.03)	0.14 (0.02)
Privileged	0.99	1.00	1.00	1.00	1.00

matching. The fixed `MEMORY.md` contains final notes, up to 70 summarized ledger entries per campaign, and common instructions for using the memory. Hidden-world values and source task files are excluded. The target agent starts a new session, reads the new brief, and can reuse relevant experience from this file. Model weights remain fixed. Each memory condition uses the same file across target tasks and repeats; the released memory files specify the full input. Related work such as Reflexion stores textual feedback for later attempts without updating model weights (Shinn et al., 2023).

C.4 Statistics and comparisons

Table 3 averages tasks equally and reports the mean and sample standard deviation across three complete repeats. Figure 2's left panel uses 2,000 task-bootstrap draws, preserving each task's three attempts; its middle-panel batch categories are independent of other acceptance checks. Figure 3 reports observed pass@3 and pass^3, without extrapolation. Attempts vary both agent behavior and laboratory seed and are distinct from fresh-batch replication.

Pilot intervals use Wilson's method. Paired exact tests use discordant task outcomes: full tools versus disabled computation/measurement has four versus zero discordant successes; full tools versus anonymous candidates has one versus six. Both two-sided tests give $p = 0.125$. Feedback and memory effects average paired repeat differences within tasks and bootstrap tasks with 5,000 draws. Alloy feedback effects are 16.7 percentage points ($[-10.0, 43.3]$) for test-and-switch and 13.3 ($[3.3, 30.0]$) for Bayesian optimization; both conditions have zero successes in the other laboratories. Memory effects are 6.7 ($[-6.7, 23.3]$), 0.0 ($[-20.0, 20.0]$), and 0.0 ($[-10.0, 10.0]$) for ALD-, liquid-, and alloy-to-polymer, and 25.0 ($[2.8, 47.2]$) for ALD v1-to-v2. All bootstrap procedures use seed 0; intervals are descriptive 95% intervals without multiple-comparison adjustment.

The library comparison uses the earlier Codex configuration at both sizes and the same strict recipe-history check. Table 10 gives absolute rates; cohort sizes are in Table 8. Releases differ in specifications, screening budgets, and time limits, so the reported changes compare complete task releases. Liquid is omitted because its libraries count salts versus complete formulations.

Table 10: Small-to-maximum library comparison using the earlier Codex configuration. Rates are task-averaged percentages, small → maximum.

Laboratory	Candidates	Codex	Claude Code
Alloy	$6 \to 443$	$33.3 \to 46.7$	$60.0 \to 30.0$
Garnet	$7 \to 1{,}055$	$20.0 \to 13.3$	$40.0 \to 13.3$
Polymer	$8 \to 531$	$13.3 \to 10.0$	$80.0 \to 36.7$
ALD	$4 \to 22$	$13.3 \to 13.3$	$40.0 \to 36.7$