WHAT MAKES AGENTS EFFECTIVE FOR SCIENTIFIC DESIGN? A CONTROLLED EVALUATION WITH SCIENTIFIC FOUNDATION MODEL TOOLS

Anonymous authors
Paper under double-blind review

ABSTRACT

Scientific agents have shown promise for automating complex computational workflows, but their usefulness for scientific design hinge on reliably completing a task *and* finding high-quality solutions. Here, we evaluate how agent design choices—including the agentic harness, Large Language Model (LLM), and tool-calling mechanism—affect these two outcomes. We introduce Nomad, a platform for serving Scientific Foundation Models (SciFMs) through either direct tool calling or Programmatic Tool Calling (PTC) via the Model Context Protocol, allowing us to vary the tool-calling mechanism without modifying the tools or agentic harnesses. Our evaluation includes 845 agent rollouts across two multi-objective design tasks: designing a multi-material system for Richtmyer–Meshkov Instability formation and screening molecular space for liquid-electrolyte solvent candidates. We measure design quality using Pareto hypervolume and fit a Bayesian hurdle model to separately estimate the probability of viable designs and the performance of the proposed designs conditional on success. We find that PTC most clearly improves reliability and performance for the multi-material design task, while the effects of harness and LLM are generally smaller and depend on the task and their pairing. Providing historical designs increases conditional hypervolume by 106.4% for the multi-material task, with agent traces suggesting that these examples helped to focus the search near the surrogate model's training distribution. Finally, higher-fidelity multi-physics simulations confirm that selected agent-proposed designs increased the historical Pareto-front hypervolume by $+44.7\%$ despite imperfect surrogate calibration. Collectively, our results demonstrate the importance of tool-calling mechanisms and scientific context when connecting SciFMs to agents for design tasks.

1 INTRODUCTION

For Artificial Intelligence (AI) agents to meaningfully complete scientific design tasks, they must successfully execute scientific workflows *and* produce useful scientific output (Kim & Romero, 2026; Lu et al., 2026). Agents must be able to identify useful candidates from large, complex design spaces found across scientific domains (Butler et al., 2018; Shachar et al., 2025). Evaluating agents on these tasks thus requires measuring both their reliability—the probability of successfully producing a design—and the expected quality of their output conditional on success. Evaluations that measure only task completion cannot distinguish between these two dimensions of effectiveness.

Tools backed by Scientific Foundation Models (SciFMs) and related surrogate models can provide agents with predictions of scientific properties or simulation results. By approximating expensive computational or experimental measurements, these tools can make repeated candidate evaluations computationally tractable, supporting iterative search over scientific design spaces (Wadell et al., 2025; Herde et al., 2024; Bodnar et al., 2025). Additionally, they provide a practical setting for comparing how agent configurations translate scientific feedback into useful designs, without the high computational cost of giving agents access to high-fidelity and computationally costly simulations. However, improvements in surrogate-predicted objectives need not translate into improvements in the underlying scientific system (Trabucco et al., 2021), motivating a distinction between surrogate-evaluated design quality and higher-fidelity validation.

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.

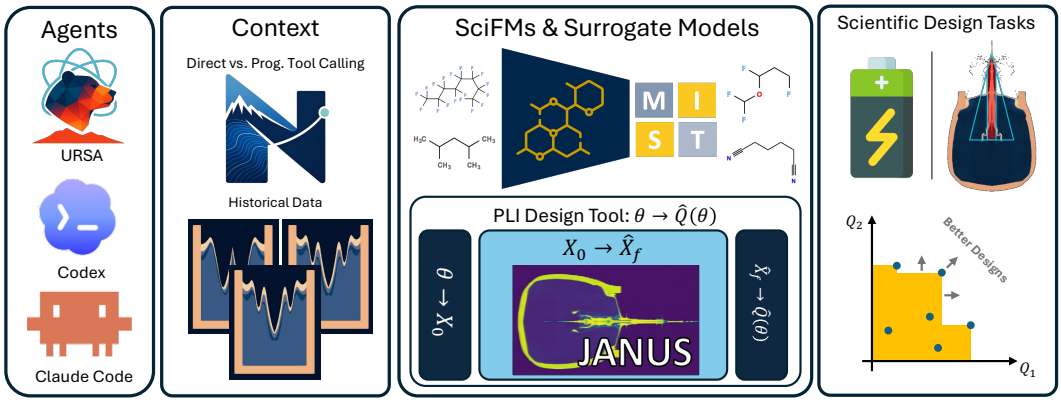

Figure 1: We benchmark how agent design, tool access, and access to historical data influences both the reliability and performance of agents on scientific design tasks.

General-purpose agents such as Codex and Claude Code, alongside scientific agents (Grosskopf et al., 2025; Shachar et al., 2025; Wang et al., 2025c; M. Bran et al., 2024), use a *harness* that enables the Large Language Model (LLM) to act autonomously through tools in pursuit of a goal (Yao et al., 2022). We examine how the backing LLM, harness, tool interface, and access to historical designs affect agents' reliability and design quality. Evaluating these choices jointly is important as the effect of one choice may depend on the others. Our analysis therefore accounts for interactions between the harness and backing LLM.

Tool access is a particularly relevant design choice for workflows that require repeated candidate evaluations. In direct tool calling, the LLM emits structured tool calls, and the harness returns their results to the conversation (Abdelaziz et al., 2024). In Programmatic Tool Calling (PTC), the LLM instead generates code that invokes and composes tools within an execution environment (Wang et al., 2024; Adam Jones & Conor Kelly, 2025; Bin Wu et al., 2025). This allows multiple evaluations and intermediate processing to occur within a single program execution without exposing the LLM to every intermediate result.

We evaluate agents on two multi-objective scientific design tasks: continuous Perturbed Layer Interface (PLI) design and discrete molecular screening. For each task, we conducted replicated rollouts across agent configurations and separately evaluated the probability of successfully proposing a design and the quality of the submitted designs conditional on success. We used Nomad, introduced here, to connect the agents to relevant SciFM-backed or other surrogate tools using either direct tool calling or PTC via the Model Context Protocol (MCP), enabling us to examine the effect of the tool interface across harnesses without modifying the harnesses. We further evaluated selected PLI designs using high-fidelity simulations.

Our findings highlight the importance of PTC for scientific design, with significant improvements in both performance and reliability on the multi-material design task (fig. 2). Although agents can propose PLI designs that materially advance the historical Pareto front (fig. 3c), their high overall failure rate (59.3%) underscores the need for greater reliability. Nonetheless, our results suggest that, when successful, agents can effectively navigate scientific design spaces, even when guided by surrogate models.

2 Related Work

Evaluating Agents for Scientific Research. Scientific agents increasingly connect an LLM to simulation tools, experimental facilities, or self-driving labs (M. Bran et al., 2024; Ghafarollahi & Buehler, 2025; Wang et al., 2025a;c; Vriza et al., 2026; Hellert et al., 2026). Evaluations to date have focused on capability demonstrations (Wang et al., 2025a;c; Vriza et al., 2026; Ghafarollahi & Buehler, 2025; Hellert et al., 2026) or agent-specific evaluations (M. Bran et al., 2024; Ansari & Moosavi, 2024; Qu et al., 2025). Generalist benchmarks of scientific research aptitude evaluate agents across multiple stages of scientific workflows (Bragg et al., 2026; Gu et al., 2024; Chen

et al., 2025). However, existing benchmarks primarily evaluate task completion, correctness, or adherence to acceptable scientific practices, rather than the domain-specific utility of otherwise valid results (Gu et al., 2024; Chen et al., 2025). Conversely, we separately evaluate task completion and design quality, measuring the latter using the Pareto hypervolume of the proposed design set.

Mechanisms for Tool Calling. Augmenting LLMs with external tools has proven to be a simple and effective way to extend their capabilities (Schick et al., 2023; Abdelaziz et al., 2024; Nakano et al., 2022; Patil et al., 2024). Using direct tool calling, LLM have driven scientific workflows (Ghafarollahi & Buehler, 2025; M. Bran et al., 2024; Wang et al., 2025b). Prior work has shown that PTC, code-mode execution, or code-as-action can improve upon direct tool calling for complex multi-step tasks by leveraging the strong code-writing abilities of LLMs (Wang et al., 2024; Surís et al., 2023; Gao et al., 2023; Adam Jones & Conor Kelly, 2025). Current implementations of PTC (Kenton Varda & Sunil Pai, 2025; Bin Wu et al., 2025; Roucher et al., 2025) rely on containerized execution environments or off-site execution services, which are often incompatible with High-Performance Computing (HPC) systems or tightly controlled computing environments. Further, they often do not permit the reuse of code modules across PTC invocations or the use of third-party libraries (Bin Wu et al., 2025; Roucher et al., 2025; Kenton Varda & Sunil Pai, 2025). Conversely, Nomad's PTC gateway executes agent-provided Python programs on-device and within the agent's execution environment. This approach enables agents to reuse code across invocations and leverage existing third-party software libraries, both core features of PTC highlighted in prior work (Wang et al., 2024; Chen et al., 2023), but which are often absent from current implementations (Kenton Varda & Sunil Pai, 2025; Bin Wu et al., 2025; Roucher et al., 2025).

Scientific foundation models for design. Prior SciFM-guided design workflows have primarily employed high-throughput screening (Wadell et al., 2025) or Bayesian optimization (Wang et al., 2025a). We instead study how general-purpose agents explore design spaces using SciFMs as evaluators, while validating improvements against higher-fidelity simulations.

3 METHODS

3.1 SCIENTIFIC TASKS

Perturbed Layer Interface Design We tasked agents with improving the design of a PLI from the High-Explosives and Affected Targets (HEAT) dataset (Kaiser et al., 2026) as measured by two Quantity of Interests (QoIs). The two QoIs were jet length and jet mass concentration in the final simulation frame (fig. 3b). Jet length is the vertical extent of the copper, measured in pixels. Jet mass concentration is the fraction of mass in the inner region of the instability (highlighted in red in fig. 3b). Agents had access to a tool for computing the QoIs from the 28 design parameters controlling the shape of the PLI (table 1). The tool used JANUS, a 10.7M-parameter UNet diffusion model for sampling the final state of the HEAT PLI simulation from the initial state (Chu et al., 2026). The tool generates the initial simulation state from the 28 parameters; JANUS then predicts the final state, from which the tool computes the QoIs. In some configurations, agents received the design parameters and their corresponding QoIs computed from the simulation outputs in the HEAT PLI dataset (Kaiser et al., 2026).

Electrolyte Screening We tasked agents with finding electrolyte solvent candidates predicted to be non-toxic, have a HOMO–LUMO gap $\geq 5\,\mathrm{eV}$, and remain liquid from 0 to $75\,^{\circ}\mathrm{C}$. We used the HOMO–LUMO gap as a physically motivated heuristic for electrochemical stability (Korth, 2014; Husch et al., 2015; Peljo & Girault, 2018; Pande & Viswanathan, 2019). Molecules with larger HOMO–LUMO gaps or a wider liquid-temperature range were scored higher, with a score of zero for molecules that just met the thresholds. Agents had access to MIST-backed surrogate models (Wadell et al., 2025) for all relevant properties, plus a collection of tools unrelated to the target objectives (table 1). We evaluated toxicity using MIST-28M-Clintox (Wadell et al., 2025) as a generic safety proxy. In addition, Agent had access to a tool for searching PubChem and mapping common molecular names to the SMILES encoding used by MIST.

3.2 AGENTIC ROLLOUTS

Across the two design tasks, we varied four factors: agent harness, LLM, tool-calling mechanism, and access to historical data (PLI-only). We evaluated both coding-focused agentic harnesses from OpenAI and Anthropic, as well as, science-focused harnesses from URSA (Grosskopf et al., 2025). To keep token costs tractable, we used Generalized Subset Design (Surowiec et al., 2017) to prune the experimental design space. Pruning was cost-aware, with more expensive LLMs receiving a higher reduction factor of up to $3\times$. For each configuration, we launched at least five replicate rollouts to capture the stochastic nature of agentic systems. In total, we evaluated configurations spanning the following axes:

Harness: Codex, Claude Code, URSA Execution, URSA Plan-Execute

LLM: GPT-5.4, GPT-5.5, Claude Sonnet 4.5, Gemma 4 31B IT, GPT-OSS 120B

Tool-calling mechanism: Direct and Programmatic Tool Calling

Context (PLI-only): With and without historical designs

For each agentic rollout, the agent was given a suitable task description for one of two tasks (PLI design or liquid-electrolyte screening) and a collection of MCP tools. MCP servers were attached to each agent using the agentic harness's documented mechanism, providing access to task-specific surrogate models. SciFM-backed tools were served using Nomad, with all models running on a single NVIDIA A100 GPU. For the PLI design task, some rollouts additionally received a CSV file containing historical PLI designs and their corresponding QoI. For rollouts using PTC, a Nomad PTC gateway was inserted between the upstream surrogate model MCP server and the agent. Identical task-specific prompts describing the provided tools were given to each agent. For direct tool calling, agents received the individual tool interfaces served via MCP. For PTC, Nomad exposed `execute_mcp_script` for executing code and `search_mcp_tools` for discovering tools available within the PTC execution environment.

Agents executed within a Charliecloud container (Priedhorsky & Randles, 2017) for isolation, provisioned with dual-socket Xeon Gold 6152 22-core processors and more than 300 GB of memory. Agents were given up to four hours to complete their task. For each task, agents were required to write their candidate solutions to a designated CSV file. We additionally logged wall times, token usage, and tool usage for each rollout.

Rollouts were included in our analysis using the following ordered assessment. We first retained all rollouts in which the harness exited with code 0. Next, we retained all rollouts that ran for more than 10 minutes and reported at least one candidate. For rollouts not retained by either criterion, we excluded those whose errors were attributable to infrastructure or external services, including node failures, storage failures, and inference-provider connection or budget failures. All other rollouts were retained, including those that failed due to incompatibilities between the LLM and agentic harness. Eligibility for analysis was distinct from statistical success, defined as $H > 0$, and was used to filter out rollouts that were not representative of agent performance. Overall, our analysis included 845 runs: 621 PLI and 224 electrolyte-screening rollouts.

3.3 STATISTICAL ANALYSIS

Each rollout was scored by how well its proposed design candidates performed across the quantities of interest (QoIs). We read the candidates $C = \{c_0, \dots\}$ directly from agent-produced CSV files and reevaluated them using the relevant surrogate models. We converted the resulting predictions into task-specific maximization scores $y_{i,j}$ for candidate c_i and objective j, incorporating feasibility thresholds. All scores were formulated so that larger values indicate better performance. To make their scales comparable, we divided each score by an objective-specific scale s_j, chosen to represent the median score for that objective.

We then summarized each rollout's performance using the hypervolume H of its normalized objective values (Van Moffaert & Nowé, 2014; Zitzler & Thiele, 1999). Hypervolume measures the volume of objective space dominated by the candidates and bounded by the origin (eq. 4). The Pareto front is the set of candidates for which no other proposed candidate performs at least as well on every objective and strictly better on at least one. Hypervolume therefore captures both candidate performance and coverage of trade-offs among objectives, with larger values indicating better

overall results. Under this normalization, a rollout proposing a single candidate with median performance on every objective ($y_{i,j} = s_j$ for all j) receives $H = 1$. Rollouts that submitted no design candidates received $H = 0$.

As a generative model, evaluating JANUS produces a possible final state from the initial state of the HEAT PLI simulation (Chu et al., 2026). As such the QoIs predicted are stochastic and vary evaluation to evaluation. To account for this, we averaged the estimated QoI over 10 evaluations for each design when computing candidate scores. We did not average QoIs for the electrolyte screening task, as the MIST models are not stochastic (Wadell et al., 2025).

Because the majority of rollouts (59.3%) had zero hypervolume, the distribution of H contains an excess of zeros. We therefore use a hurdle model for H (Cragg, 1971), with separate fixed-effects regressions for success $\mathbb{I}(H_i > 0)$ and performance conditional on success $H_i \mid H_i > 0$:

$$\mathbb{I}(H_i > 0) \sim \text{Bernoulli}\big(\text{logit}^{-1}(\mathbf{x}_i^\top \boldsymbol{\beta}_s)\big) \tag{1}$$

$$\log H_i \mid H_i > 0 \sim \text{Normal}\big(\mathbf{x}_i^\top \boldsymbol{\beta}_+, \sigma^2\big) \tag{2}$$

Here, $\mathbf{x}$ is the model matrix specified by the fixed-effects model `(llm_model * agent_cls + has_data + code_mode) * task`; expanded linear predictors are provided in appendix C. All predictors were treatment-coded with reference levels of GPT-5.4, URSA Execution, no historical data, and direct tool calling. Parameters ($\boldsymbol{\beta}_s,\ \boldsymbol{\beta}_+, \sigma$) were estimated with Markov-Chain Monte Carlo (MCMC) using No U-Turn Sampler (NUTS) as implemented in `Turing.jl` (Fjelde et al., 2025). Priors, sampling procedure, convergence diagnostics, model comparisons, and tabulated effect sizes are presented in appendix C.

We report posterior expected effect sizes relative to the baseline, computed from posterior draws. For the success rate ($\mathbb{I}(H > 0)$), effects are reported as the relative risk for a particular treatment E relative to the baseline (eq. 3a). For performance ($H \mid H > 0$), effects are reported as a percentage change relative to the baseline treatment (eq. 3b). For both measures, $\mathbf{x}_E$ indicates the covariate setting with treatment E applied and all other factors held at the baseline reference setting $\mathbf{x}_0$.

$$\text{RR}_{\text{success}}(E) = \frac{p(H > 0 \mid \mathbf{x}_E)}{p(H > 0 \mid \mathbf{x}_0)}, \tag{3a}$$

$$\%\Delta(E) = 100 \left[\frac{\mathbb{E}[H \mid H > 0, \mathbf{x}_E]}{\mathbb{E}[H \mid H > 0, \mathbf{x}_0]} - 1 \right] \tag{3b}$$

Notably, eq. 1 models the expected performance of a rollout as $\mathbb{E}[H \mid \mathbf{x}] = p(H > 0 \mid \mathbf{x})\mathbb{E}[H \mid H > 0, \mathbf{x}]$. This formulation enables us to separate the effects of a treatment on reliability $p(H > 0 \mid \mathbf{x})$ and performance $\mathbb{E}[H \mid H > 0, \mathbf{x}]$.

3.4 Validation

For the PLI design task, we validated a subset of agent-proposed designs using PAGOSA (Subramanian et al., 2020), the high-fidelity simulator used to develop the HEAT PLI dataset (Kaiser et al., 2026). PAGOSA is an Eulerian finite-difference code developed for simulating high-speed compressible flows and high-rate material deformation (Subramanian et al., 2020; Kaiser et al., 2026). Simulating a PLI single design with PAGOSA takes ≈ 45 minutes on 112 CPUs across two nodes, approximately $450\times$ slower in latency than the JANUS predictions on an NVIDIA A100 GPU.

The PLI designs for validation with PAGOSA (Subramanian et al., 2020) were selected to cover both the Pareto front and a diverse subset of designs. Designs were first filtered to ensure the PLI remained within the prescribed geometric bounds; this removed 30 of 2,694 designs. We then selected 15 designs approximately evenly distributed along the Pareto front. Next, we used a coreset sampler to select 70 designs from the pool of agent designs. The coreset was initialized using nontrivial DBSCAN (Ester et al., 1996) cluster representatives, followed by greedy farthest-first sampling (Indyk et al., 2014; Sener & Savarese, 2018; Théo Viel & Ahmet Erdem, 2024). PAGOSA successfully simulated 84 of the 85 selected agent designs; one design failed during simulation setup because the geometry engine rejected its discontinuous layers.

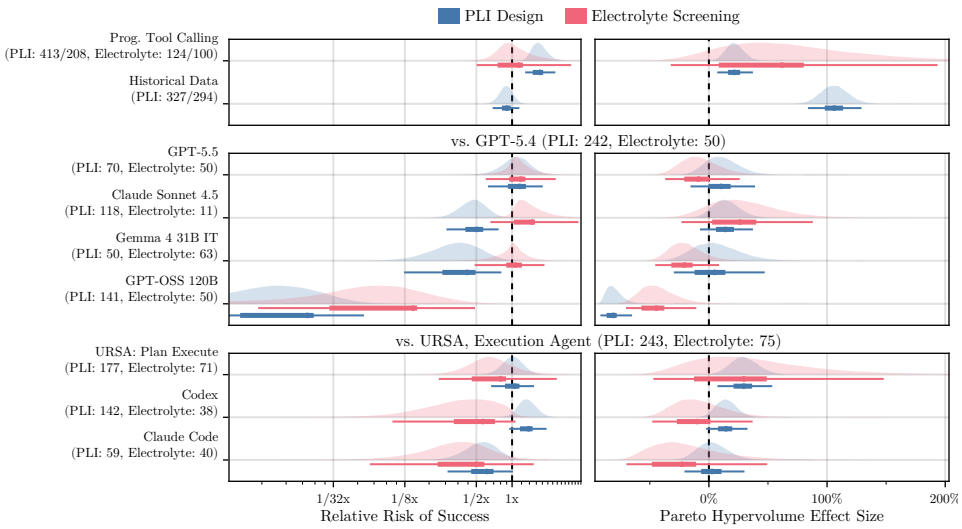

Figure 2: Posterior expected relative risk of success (left, eq. 3a) and percentage change in Pareto-front hypervolume (right, eq. 3b) from the fitted hurdle models (eq. 1); higher is better for both. The thick and thin intervals represent the 50% and 95% Highest Density Interval, respectively. These values were computed using 10,000 retained draws from each of six independent MCMC chains. Task-specific observation counts for each treatment level are reported. For binary treatments (e.g., PTC), counts are reported as Y/N, indicating the numbers of observations with and without the treatment.

To contextualize agent performance, we compared the proposed designs with those obtained using the NSGA-II multi-objective optimizer. NSGA-II is an elitist multi-objective evolutionary algorithm that uses nondominated sorting and crowding distance to maintain a diverse set of nondominated solutions (Deb et al., 2002; Blank & Deb, 2020). We used NSGA-II to optimize the PLI QoIs predicted by the JANUS surrogate model (fig. 3). The baseline NSGA-II configuration used a population of 10 for 1,000 generations, with each design's QoIs computed by averaging 10 JANUS evaluations. As a sensitivity study, we repeated the optimization using different random seeds, population sizes (10 or 100), and numbers of generations (100, 500, or 1,000). We pooled candidates from all runs when constructing the NSGA-II Pareto front. We further validated 15 NSGA-II designs evenly distributed along the Pareto front by simulating the designs with PAGOSA.

4 Experiments and Results

4.1 Agent-design choices affect reliability and performance

We estimated the posterior expected effects of agent design choices using posterior samples of eq. 1 after fitting with NUTS (fig. 2); expected effect sizes and expectations are tabulated in appendix C.2. For an equal-weight mixture of the two tasks, the posterior mean indicated that PTC increased reliability ($1.5\times$) and conditional hypervolume by 42% (table 4). Use of GPT-OSS 120B delivered notably lower reliability and performance (fig. 2), but otherwise both flagship (e.g. GPT-5.5, Claude Sonnet 4.5) and non-frontier LLM (i.e. Gemma 4 31B IT) performed similarly. There were task-specific outliers for success (Gemma 4 31B IT for PLI design) or performance (URSA's Plan-Execute for PLI design). However, the overall trend was of LLM and harness selection having a limited effect on outcomes and there was no consistently superior configuration (fig. 2). Including harness, LLM, and harness–LLM interaction effects improved predictive fit, likely due to improving the allocation of effects to treatments. Removing all harness and LLM terms increased Watanabe–Akaike information criterion (WAIC) by 368.2 ± 35.6, while retaining their main effects, but removing their interaction increased WAIC by 67.6 ± 18.4 (table 3). WAIC (eq. 10) quantifies the pointwise posterior out-of-sample prediction accuracy of a Bayesian model, with lower values suggesting better generalization performance (Vehtari et al., 2017).

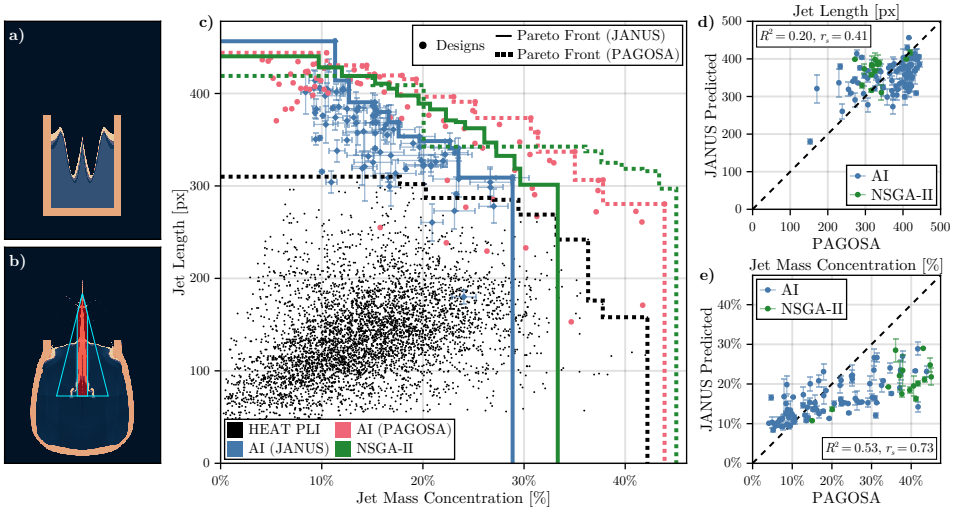

Figure 3: a) Agents were tasked with designing a PLI parameterized by 28 parameters (Kaiser et al., 2026) to maximize two QoIs of the resulting Richtmyer–Meshkov Instability (RMI). b) The RMI QoIs measured the length and mass concentration of the resulting jet. c) Pareto fronts of AI-proposed designs based on JANUS estimates (Chu, 2026) and PAGOSA-simulated performance. Under both estimates, the AI designs extended beyond the historical PLI Pareto front. The JANUS estimate primarily advanced the jet length metric, but produced a hypervolume of 5.4, slightly below the historical value of 5.53. d&e) Parity plots of the JANUS-estimated vs. PAGOSA-simulated QoIs. JANUS was better calibrated for mass concentration than for jet length, but retained positive rank correlations for both.

For the PLI design task, access to historical designs proved to be critical to achieving good performance (fig. 2); it increased the expected hypervolume conditional on success by 106.4% (95% Highest Density Interval (HDI): 84%–130%) but did not materially affect the probability of success ($0.91\times$, 95% HDI: 0.69–$1.2\times$; fig. 2). Without access to prior designs, agents overwhelmingly selected points within the 28-dimensional unit hypercube ($[0, 1]^{28}$), which for this simulation was non-physical. Conversely, when provided with prior designs, agents would sample a region slightly larger than that covered by the historical designs. Manual examination of agent traces revealed that agents attempted to avoid moving too far outside the supplied design space, which further kept their candidates near JANUS's training distribution. Notably, the supplied PLI design tool provided a reference PLI design (average design parameters over all design in HEAT PLI) in it's description. As such, the observed performance uplift can not be merely attributed to providing omitted design space context.

4.2 Higher Fidelity Validation of Agentic PLI Design

Both the JANUS-predicted and PAGOSA-simulated Pareto fronts for the agent designs extended beyond the historical PLI fronts (fig. 3). The final PAGOSA results confirmed that agents increased the Pareto-front hypervolume from the historical value of 5.53 to 8.0 ($+44.7\%$); the JANUS-predicted hypervolume was 5.4. While the performance of some designs was overstated, on the whole the JANUS-predicted QoIs underestimated the PAGOSA-simulated QoIs.

AI-designs remained competitive when compared to designs produced by NSGA-II, a multi-objective optimizer (Deb et al., 2002). The two search methods resulted in comparable hypervolumes: 8.0 for the agent designs and 7.95 for the NSGA-II designs. The reported PAGOSA hypervolumes were computed from 84 successfully simulated agent designs and 15 NSGA-II designs, respectively. Randomly resampling 15 agent designs from the 84 evaluated designs yielded an agent Pareto-front hypervolume of 6.93 (95% HDI: 5.85–7.81), suggesting that the agent search strategy remained competitive with NSGA-II. Among rollouts reporting at least one design, agents reported a median of 9 and at most 82 designs, compared with the 10,000-design evaluation budget of the

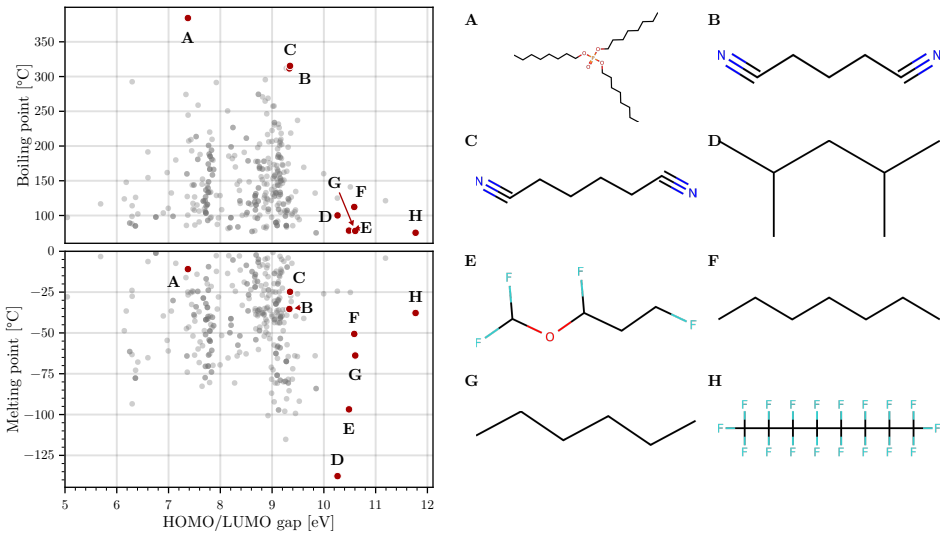

Figure 4: Left: Melting and boiling points versus HOMO–LUMO gap, as computed using fine-tuned MIST models (Wadell et al., 2025) for AI-proposed molecules. Right: Subset of Pareto-front molecules, over all runs, proposed by AI agents seeking non-toxic molecules that are liquid from 0 to 75 °C and have a HOMO–LUMO gap ≥ 5 eV.

baseline NSGA-II configuration; reported-design counts do not measure the total number of agent evaluations.

4.3 Liquid Electrolyte Screening

To further assess the effects of agentic design choices and PTC for scientific design tasks, we tasked agents with a liquid electrolyte screening task using predictions from fine-tuned MIST surrogate models (Wadell et al., 2025). As a surrogate-only evaluation, this task was not validated with wet-lab experimentation, nor did it evaluate all relevant chemical properties for a true *in silico* discovery pipeline. Instead, our focus was to explore the capacity of agentic systems to navigate molecular space guided by surrogate predictions. Unlike the PLI design task, molecular discovery requires navigating a discrete, combinatorial, and highly nonlinear design space rather than the bounded and continuous design space of PLI designs. The space of synthetically feasible small molecules below 500 Da has been estimated to contain more than 10^{60} structures, making exhaustive enumeration impractical and motivating targeted search strategies (Kirkpatrick & Ellis, 2004; Virshup et al., 2013).

Agents proposed molecules (fig. 4) included several known Li-ion electrolyte solvents, additives, or diluents. Trioctyl phosphate (A) has been studied as a flame-retardant Li-ion electrolyte additive (Shim et al., 2009), whereas glutaronitrile (B) has been investigated as an electrolyte co-solvent for high-energy, high-power cells (Xu et al., 2017; Abu-Lebdeh & Davidson, 2009b). Adiponitrile (C) has been evaluated in both high-voltage liquid electrolytes (Wang et al., 2018; Abu-Lebdeh & Davidson, 2009a; Narayanan Kirshnamoorthy et al., 2018) and soft co-crystalline solid electrolytes (Prakash et al., 2023). The nonpolar hydrocarbons (n)-heptane (F) and (n)-hexane (G) have likewise been used as diluents in lithium-battery electrolyte formulations (Wang et al., 2023; Cheng et al., 2023; Kim et al., 2024). While not completely unexpected, the rediscovery of known electrolyte solvents, additives, and diluents suggests that the agents employed a reasonable search strategy. This result is likely reflective of LLM internal knowledge of common electrolytes, however this remains orthogonal to the agent's demonstrated ability to locate molecules meeting the screening criteria.

5 CONCLUSION

In this work, we evaluated how the agentic harness, backing LLM, tool-calling mechanism, and access to historical designs affect the reliability and performance of agents for scientific design. Using Nomad, we exposed the same SciFM-backed tools through either direct tool calling or PTC without modifying the tools or agentic harnesses. This enabled a controlled comparison across two distinct multi-objective tasks: continuous PLI design and discrete electrolyte screening. Rather than treating task completion as the sole outcome, we used a Bayesian hurdle model (eq. 1) to separate the probability of viable designs from the quality of the proposed designs, as measured by the Pareto hypervolume of their QoIs.

We found PTC to be the most consequential agent design choice considered here (fig. 2). In particular, PTC improved both reliability and conditional performance for the PLI task by allowing agents to programmatically batch model evaluations, process their outputs, and combine the provided tools with numerical libraries. Surpassingly, we found the effect of harness and LLM to be minimal, with no consistently superior combination.

Access to historical designs was similarly important for the PLI task. Providing these designs increased the expected hypervolume conditional on success by 106.4% without materially changing the probability of proposing a design (fig. 2). Without this context, agents often searched the unit hypercube, a region that was largely non-physical for the PLI simulation. With historical designs, agents instead searched near the supplied design space and, consequently, closer to the training distribution of JANUS (Chu, 2026). This result highlights that effective scientific search requires not only rapid model evaluations, but also sufficient context to define a scientifically meaningful design space.

Finally, we validated selected agent-proposed PLI designs using higher-fidelity PAGOSA simulations (fig. 3c). Despite imperfect calibration between JANUS and PAGOSA, the validated designs increased the historical Pareto-front hypervolume from 5.53 to 8.0 and achieved a front competitive with the NSGA-II hypervolume of 7.95 (fig. 3c). Because the agent and optimizer used different search and validation budgets, this comparison does not establish that agents outperform numerical optimization. Instead, it demonstrates that general-purpose agents can use an imperfect surrogate to identify designs that retain their improvement under higher-fidelity simulation. The molecular task provides a complementary surrogate-only result: agents recovered several molecules previously studied as electrolyte solvents, additives, or diluents, but these candidates require additional property evaluation and experimental validation (fig. 4).

Collectively, our results demonstrate the potential of combining agentic systems with SciFM-backed tools for scientific design, while showing that performance depends strongly on how the tools and design space are exposed to the agent. Using SciFM-backed tools, rather than high-fidelity simulations or wet-lab experiments, also provided a tractable setting for evaluating agents on scientific design tasks. Our results indicate an overall low success rate, suggesting that more robust agents are needed before they can be directly coupled to higher-fidelity simulations or experiments. However, their performance conditional on success suggests that, once their robustness improves, agents may be capable of autonomously conducting scientific design tasks. Future work should explore mechanisms for improving agent robustness and error recovery, as well as methods for incorporating surrogate uncertainty into the agent decision-making loop.

6 REPRODUCING THE RESULTS

We have deposited the source code and data need to replicate our analysis into Zenodo at doi:10.5281/zenodo.22818752. This release includes traces of agent rollouts, fitted hurdle models, MCP tools and Nomad configuration. Model weights will not be redistributed, instead pointers to the public releases are documented in table 1. The HEAT dataset can be found at `https://oceans11.lanl.gov/` as described in Kaiser et al. (2026). The PAGOSA input decks, source code and executables are proprietary, as such they will not be included within our release. However, the provided jet design tool can reproduce the initial state of the PLI simulation, for use with other solvers.

There are two additional consideration that may limited the reproducibility of our experiments. First, the majority of our experiments use API-based (closed-source) LLM, which may limit the long-term reproducibility of our work as models are retired. Second, all of our experiments involve LLM and thus are inherently stochastic. We have partially mitigated this by running replicated rollouts and by leveraging MCMC to fit our hurdle models to better estimate the uncertainty of our results.

7 AI USE STATEMENT

In this work, we used generative AI tools to create software code, summarize existing literature, brainstorm, copy-edit, translate code into mathematical prose, and draft parts of the paper. We did not use generative AI tools to propose or refine hypotheses; design or provide feedback on research methodology or experiments; provide critical ingredients for proving mathematical claims; assist with translation; support qualitative or thematic data analysis; or interpret results. Generating synthetic datasets, theoretical models or conceptual frameworks, and cleaning or reformatting datasets were not applicable to this work. We have closely reviewed all AI-assisted work, by reviewing all suggested literature to confirm relevance and reviewing generate code/tests/results for correctness. We take full responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Ibrahim Abdelaziz, Kinjal Basu, Mayank Agarwal, Sadhana Kumaravel, Matthew Stallone, Rameswar Panda, Yara Rizk, G. P. Bhargav, Maxwell Crouse, Chulaka Gunasekara, Shajith Ikbal, Sachin Joshi, Hima Karanam, Vineet Kumar, Asim Munawar, Sumit Neelam, Dinesh Raghu, Udit Sharma, Adriana Meza Soria, Dheeraj Sreedhar, Praveen Venkateswaran, Merve Unuvar, David Cox, Salim Roukos, Luis Lastras, and Pavan Kapanipathi. Granite-Function Calling Model: Introducing Function Calling Abilities via Multi-task Learning of Granular Tasks, June 2024. URL http://arxiv.org/abs/2407.00121.

Yaser Abu-Lebdeh and Isobel Davidson. High-Voltage Electrolytes Based on Adiponitrile for Li-Ion Batteries. *Journal of The Electrochemical Society*, 156(1):A60, 2009a. ISSN 00134651. doi:10.1149/1.3023084. URL https://iopscience.iop.org/article/10.1149/1.3023084.

Yaser Abu-Lebdeh and Isobel Davidson. New electrolytes based on glutaronitrile for high energy/power Li-ion batteries. *Journal of Power Sources*, 189(1):576–579, April 2009b. ISSN 03787753. doi:10.1016/j.jpowsour.2008.09.113. URL https://linkinghub.elsevier.com/retrieve/pii/S0378775308019125.

Adam Jones and Conor Kelly. Code execution with MCP: Building more efficient agents, November 2025. URL https://www.anthropic.com/engineering/code-execution-with-mcp.

Mehrad Ansari and Seyed Mohamad Moosavi. Agent-based learning of materials datasets from the scientific literature. *Digital Discovery*, 3(12):2607–2617, 2024. ISSN 2635-098X. doi:10.1039/D4DD00252K. URL https://pubs.rsc.org/dd/article/3/12/2607-2617/846152.

Bin Wu, Adam Jones, Artur Renault, Henry Tay, Jake Noble, Noah Picard, Sam Jiang, and the Claude Developer Platform team. Introducing advanced tool use on the Claude Developer Platform, November 2025. URL https://www.anthropic.com/engineering/advanced-tool-use.

Julian Blank and Kalyanmoy Deb. Pymoo: Multi-Objective Optimization in Python. *IEEE Access*, 8:89497–89509, 2020. ISSN 2169-3536. doi:10.1109/ACCESS.2020.2990567. URL https://ieeexplore.ieee.org/document/9078759/.

Cristian Bodnar, Wessel P. Bruinsma, Ana Lucic, Megan Stanley, Anna Allen, Johannes Brandstetter, Patrick Garvan, Maik Riechert, Jonathan A. Weyn, Haiyu Dong, Jayesh K. Gupta, Kit Thambiratnam, Alexander T. Archibald, Chun-Chieh Wu, Elizabeth Heider, Max Welling, Richard E.

Turner, and Paris Perdikaris. A foundation model for the Earth system. *Nature*, 641(8065): 1180–1187, May 2025. ISSN 0028-0836, 1476-4687. doi:10.1038/s41586-025-09005-y. URL https://www.nature.com/articles/s41586-025-09005-y.

Jonathan Bragg, Mike D'Arcy, Nishant Balepur, Dan Bareket, Bhavana Dalvi Mishra, Sergey Feldman, Dany Haddad, Jena D. Hwang, Peter Jansen, Varsha Kishore, Bodhisattwa Prasad Majumder, Aakanksha Naik, Sigal Rahamimov, Kyle Richardson, Amanpreet Singh, Harshit Surana, Aryeh Tiktinsky, Rosni Vasu, Guy Wiener, Chloe Anastasiades, Stefanus Candra, Jason Dunkelberger, Daniel Emery, Rob Evans, Malachi Hamada, Regan Huff, Rodney Kinney, Matt Latzke, Jaron Lochner, Ruben Lozano-Aguilera, Ngoc-Uyen Nguyen, Smita Rao, Amber Tanaka, Brooke Vlahos, Peter Clark, Doug Downey, Yoav Goldberg, Ashish Sabharwal, and Daniel S Weld. AstaBench: Rigorous benchmarking of AI agents with a scientific research suite. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=M7TNf5J26u.

Keith T. Butler, Daniel W. Davies, Hugh Cartwright, Olexandr Isayev, and Aron Walsh. Machine learning for molecular and materials science. *Nature*, 559(7715):547–555, July 2018. ISSN 0028-0836, 1476-4687. doi:10.1038/s41586-018-0337-2. URL https://www.nature.com/articles/s41586-018-0337-2.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023. doi:10.48550/arXiv.2211.12588. URL https://openreview.net/forum?id=YfZ4ZPt8zd.

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, March 2025. URL http://arxiv.org/abs/2410.05080.

Hao Cheng, Shichao Zhang, Bing Zhang, and Yingying Lu. *N* -Hexane Diluted Electrolyte with Ultralow Density enables Li–S Pouch Battery Toward >400 Wh kg^{-1}. *Small*, 19(9):2206375, March 2023. ISSN 1613-6810, 1613-6829. doi:10.1002/smll.202206375. URL https://onlinelibrary.wiley.com/doi/10.1002/smll.202206375.

Mengdi Chu. Janus-first2last, June 2026. URL https://huggingface.co/90879c/Janus-first2last.

Mengdi Chu, Jiaxin Yang, Angus G. Forbes, Nathan Debardeleben, Earl Lawrence, Ayan Biswas, and Han-Wei Shen. DiffUNet^2: Bidirectional Prediction, Probabilistic Generation and Collaborative Visual Discovery for Scientific Data, 2026. URL https://arxiv.org/abs/2606.03926.

John G. Cragg. Some Statistical Models for Limited Dependent Variables with Application to the Demand for Durable Goods. *Econometrica*, 39(5):829, September 1971. ISSN 00129682. doi:10.2307/1909582. URL https://www.jstor.org/stable/1909582?origin=crossref.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002. ISSN 1089778X. doi:10.1109/4235.996017. URL http://ieeexplore.ieee.org/document/996017/.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pp. 226–231, Portland, Oregon, August 1996. AAAI Press.

Tor Erlend Fjelde, Kai Xu, David Widmann, Mohamed Tarek, Cameron Pfiffer, Martin Trapp, Seth D. Axen, Xianda Sun, Markus Hauru, Penelope Yong, Will Tebbutt, Zoubin Ghahramani, and Hong Ge. Turing.jl: A General-Purpose Probabilistic Programming Language. *ACM*

Transactions on Probabilistic Machine Learning, 1(3):1–48, September 2025. ISSN 2836-8924. doi:10.1145/3711897. URL https://dl.acm.org/doi/10.1145/3711897.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. PAL: Program-aided Language Models, January 2023. URL http://arxiv.org/abs/2211.10435.

Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. Chapman and Hall/CRC, 3 edition, 2014. ISBN 978-1-4398-4095-5.

Alireza Ghafarollahi and Markus J. Buehler. Automating alloy design and discovery with physics-aware multimodal multiagent AI. *Proceedings of the National Academy of Sciences*, 122(4): e2414074122, January 2025. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.2414074122. URL https://pnas.org/doi/10.1073/pnas.2414074122.

Michael Grosskopf, Russell Bent, Rahul Somasundaram, Isaac Michaud, Arthur Lui, Nathan Debardeleben, and Earl Lawrence. URSA: The Universal Research and Scientific Agent, June 2025. URL http://arxiv.org/abs/2506.22653.

Ken Gu, Ruoxi Shang, Ruien Jiang, Keying Kuang, Richard-John Lin, Donghe Lyu, Yue Mao, Youran Pan, Teng Wu, Jiaqian Yu, Yikun Zhang, Tianmai M. Zhang, Lanyi Zhu, Mike A Merrill, Jeffrey Heer, and Tim Althoff. BLADE: Benchmarking Language Model Agents for Data-Driven Science. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 13936–13971, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi:10.18653/v1/2024.findings-emnlp.815. URL https://aclanthology.org/2024.findings-emnlp.815.

Thorsten Hellert, João Montenegro, and Antonin Sulc. Osprey: Production-ready agentic AI for safety-critical control systems. *APL Machine Learning*, 4(1):016103, March 2026. ISSN 2770-9019. doi:10.1063/5.0306302. URL https://pubs.aip.org/aml/article/4/1/016103/3378552/Osprey-Production-ready-agentic-AI-for-safety.

Maximilian Herde, Bogdan Raonić, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel de Bézenac, and Siddhartha Mishra. Poseidon: Efficient Foundation Models for PDEs, November 2024. URL http://arxiv.org/abs/2405.19101.

Tamara Husch, Nusret Duygu Yilmazer, Andrea Balducci, and Martin Korth. Large-scale virtual high-throughput screening for the identification of new battery electrolyte solvents: Computing infrastructure and collective properties. *Physical Chemistry Chemical Physics*, 17(5):3394–3401, 2015. ISSN 1463-9076, 1463-9084. doi:10.1039/C4CP04338C. URL https://pubs.rsc.org/cp/article/17/5/3394-3401/433520.

Piotr Indyk, Sepideh Mahabadi, Mohammad Mahdian, and Vahab S. Mirrokni. Composable core-sets for diversity and coverage maximization. In *Proceedings of the 33rd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 100–108, Snowbird Utah USA, June 2014. ACM. ISBN 978-1-4503-2375-8. doi:10.1145/2594538.2594560. URL https://dl.acm.org/doi/10.1145/2594538.2594560.

Bryan Kaiser, Kyle Hickmann, Sharmistha Chakrabarti, Soumi De, Sourabh Pandit, David Schodt, Jesus Pulido, Divya Banesh, and Christine Sweeney. The High Explosives and Affected Targets (HEAT) Dataset, April 2026. URL http://arxiv.org/abs/2604.18828.

Kenton Varda and Sunil Pai. Code Mode: The better way to use MCP, September 2025. URL https://blog.cloudflare.com/code-mode/.

Jeonghyeon Kim and Philip Romero. Benchmarking and behavioral characterization of LLM agents for protein design, May 2026. URL http://biorxiv.org/lookup/doi/10.64898/2026.05.06.723381.

Minkwan Kim, Jiwoo An, Seung-Jae Shin, Insu Hwang, Jimin Lee, Youngbin Park, Jinyoung Kim, Eunseok Park, Jisub Kim, Gyuleen Park, Sujin Kim, Ali Coskun, and Jang Wook

Choi. Anti-corrosive electrolyte design for extending the calendar life of lithium metal batteries. *Energy & Environmental Science*, 17(16):6079–6090, 2024. ISSN 1754-5692, 1754-5706. doi:10.1039/D4EE01436G. URL https://pubs.rsc.org/ee/article/17/16/6079-6090/848606.

Peter Kirkpatrick and Clare Ellis. Chemical space. *Nature*, 432(7019):823–823, December 2004. ISSN 0028-0836, 1476-4687. doi:10.1038/432823a. URL https://www.nature.com/articles/432823a.

Martin Korth. Large-scale virtual high-throughput screening for the identification of new battery electrolyte solvents: Evaluation of electronic structure theory methods. *Phys. Chem. Chem. Phys.*, 16(17):7919–7926, 2014. ISSN 1463-9076, 1463-9084. doi:10.1039/C4CP00547C. URL https://pubs.rsc.org/cp/article/16/17/7919-7926/431625.

Chris Lu, Cong Lu, Robert Tjarko Lange, Yutaro Yamada, Shengran Hu, Jakob Foerster, David Ha, and Jeff Clune. Towards end-to-end automation of AI research. *Nature*, 651(8107):914–919, March 2026. ISSN 0028-0836, 1476-4687. doi:10.1038/s41586-026-10265-5. URL https://www.nature.com/articles/s41586-026-10265-5.

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, May 2024. ISSN 2522-5839. doi:10.1038/s42256-024-00832-8. URL https://www.nature.com/articles/s42256-024-00832-8.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. WebGPT: Browser-assisted question-answering with human feedback, June 2022. URL http://arxiv.org/abs/2112.09332.

Anand Narayanan Kirshnamoorthy, Kristina Oldiges, Martin Winter, Andreas Heuer, Isidora Cekic-Laskovic, Christian Holm, and Jens Smiatek. Electrolyte solvents for high voltage lithium ion batteries: Ion correlation and specific anion effects in adiponitrile. *Physical Chemistry Chemical Physics*, 20(40):25701–25715, 2018. ISSN 1463-9076, 1463-9084. doi:10.1039/C8CP04102D. URL https://pubs.rsc.org/cp/article/20/40/25701-25715/599009.

Vikram Pande and Venkatasubramanian Viswanathan. Descriptors for Electrolyte-Renormalized Oxidative Stability of Solvents in Lithium-Ion Batteries. *The Journal of Physical Chemistry Letters*, 10(22):7031–7036, November 2019. ISSN 1948-7185, 1948-7185. doi:10.1021/acs.jpclett.9b02717. URL https://pubs.acs.org/doi/10.1021/acs.jpclett.9b02717.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large language model connected with massive apis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/e4c61f578ff07830f5c37378dd3ecb0d-Paper-Conference.pdf.

Pekka Peljo and Hubert H. Girault. Electrochemical potential window of battery electrolytes: The HOMO–LUMO misconception. *Energy & Environmental Science*, 11(9):2306–2309, 2018. ISSN 1754-5692, 1754-5706. doi:10.1039/C8EE01286E. URL https://pubs.rsc.org/ee/article/11/9/2306-2309/603850.

Prabhat Prakash, Birane Fall, Jordan Aguirre, Laura A. Sonnenberg, Parameswara Rao Chinnam, Sumanth Chereddy, Dmitriy A. Dikin, Arun Venkatnathan, Stephanie L. Wunder, and Michael J. Zdilla. A soft co-crystalline solid electrolyte for lithium-ion batteries. *Nature Materials*, 22(5):627–635, May 2023. ISSN 1476-1122, 1476-4660. doi:10.1038/s41563-023-01508-1. URL https://www.nature.com/articles/s41563-023-01508-1.

Reid Priedhorsky and Tim Randles. Charliecloud: Unprivileged containers for user-defined software stacks in HPC. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–10, Denver Colorado, November 2017. ACM. ISBN 978-1-4503-5114-0. doi:10.1145/3126908.3126925. URL https://dl.acm.org/doi/10.1145/3126908.3126925.

Yuanhao Qu, Kaixuan Huang, Ming Yin, Kanghong Zhan, Dyllan Liu, Di Yin, Henry C. Cousins, William A. Johnson, Xiaotong Wang, Mihir Shah, Russ B. Altman, Denny Zhou, Mengdi Wang, and Le Cong. CRISPR-GPT for agentic automation of gene-editing experiments. *Nature Biomedical Engineering*, 10(2):245–258, July 2025. ISSN 2157-846X. doi:10.1038/s41551-025-01463-z. URL https://www.nature.com/articles/s41551-025-01463-z.

Aymeric Roucher, Albert Villanova del Moral, Thomas Wolf, Leandro von Werra, and Erik Kaunismäki. 'smolagents': A smol library to build great agentic systems., 2025. URL https://github.com/huggingface/smolagents.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *arXiv preprint*, 2023. URL https://arxiv.org/abs/2302.04761.

Ozan Sener and Silvio Savarese. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *International Conference on Learning Representations*, 2018. doi:10.48550/arXiv.1708.00489. URL http://arxiv.org/abs/1708.00489.

Meir H. Shachar, Dane M. Sterbentz, Harshitha Menon, Charles F. Jekel, M. Giselle Fernández-Godino, Nathan K. Brown, Ismael D. Boureima, Yue Hao, Kevin Korner, Robert Rieben, Daniel A. White, William J. Schill, and Jonathan L. Belof. Multi-Agent Design Assistant for the Simulation of Inertial Fusion Energy, October 2025. URL http://arxiv.org/abs/2510.17830.

Eun-Gi Shim, Tae-Heum Nam, Jung-Gu Kim, Hyun-Soo Kim, and Seong-In Moon. Effects of Trioctyl Phosphate and Cresyl Diphenyl Phosphate as flame-retarding additives for Li-Ion battery electrolytes. *Metals and Materials International*, 15(4):615–621, August 2009. ISSN 1598-9623, 2005-4149. doi:10.1007/s12540-009-0615-5. URL http://link.springer.com/10.1007/s12540-009-0615-5.

Gopinath Subramanian, Wayne Weseloh, Sean Clancy, and James Painter. PAGOSA Theory Manual. Technical Report LA-UR-20-29881, Los Alamos National Laboratory, December 2020. URL https://www.osti.gov/biblio/1734678.

Dídac Surís, Sachit Menon, and Carl Vondrick. ViperGPT: Visual Inference via Python Execution for Reasoning, March 2023. URL http://arxiv.org/abs/2303.08128.

Izabella Surowiec, Ludvig Vikström, Gustaf Hector, Erik Johansson, Conny Vikström, and Johan Trygg. Generalized Subset Designs in Analytical Chemistry. *Analytical Chemistry*, 89(12): 6491–6497, June 2017. ISSN 0003-2700, 1520-6882. doi:10.1021/acs.analchem.7b00506. URL https://pubs.acs.org/doi/10.1021/acs.analchem.7b00506.

Théo Viel and Ahmet Erdem. Diversity sampling: GPU-accelerated coreset sampling. NVIDIA, January 2024. URL https://github.com/NVIDIA/Diversity-Sampling.

Brandon Trabucco, Aviral Kumar, Xinyang Geng, and Sergey Levine. Conservative objective models for effective offline model-based optimization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10358–10368. PMLR, 2021. URL https://proceedings.mlr.press/v139/trabucco21a.html.

Kristof Van Moffaert and Ann Nowé. Multi-objective reinforcement learning using sets of pareto dominating policies. *Journal of Machine Learning Research*, 15(107):3663–3692, 2014. URL https://jmlr.org/papers/v15/vanmoffaert14a.html.

Aki Vehtari, Andrew Gelman, and Jonah Gabry. Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27(5):1413–1432, September 2017. ISSN 0960-3174, 1573-1375. doi:10.1007/s11222-016-9696-4. URL http://arxiv.org/abs/1507.04544.

Aki Vehtari, Andrew Gelman, Daniel Simpson, Bob Carpenter, and Paul-Christian Bürkner. Rank-Normalization, Folding, and Localization: An Improved R-hat for Assessing Convergence of MCMC. *Bayesian Analysis*, 16(2):667–718, June 2021. ISSN 1936-0975. doi:10.1214/20-BA1221.

Aaron M. Virshup, Julia Contreras-García, Peter Wipf, Weitao Yang, and David N. Beratan. Stochastic Voyages into Uncharted Chemical Space Produce a Representative Library of All Possible Drug-Like Compounds. *Journal of the American Chemical Society*, 135(19):7296–7303, May 2013. ISSN 0002-7863, 1520-5126. doi:10.1021/ja401184g. URL https://pubs.acs.org/doi/10.1021/ja401184g.

Aikaterini Vriza, Michael H. Prince, Tao Zhou, Henry Chan, and Mathew J. Cherukara. Operating advanced scientific instruments with AI agents that learn on the job. *npj Computational Materials*, 12(1):160, March 2026. ISSN 2057-3960. doi:10.1038/s41524-026-02005-0. URL https://www.nature.com/articles/s41524-026-02005-0.

Alexius Wadell, Anoushka Bhutani, Victor Azumah, Austin R. Ellis-Mohr, Celia Kelly, Hancheng Zhao, Anuj K. Nayak, Kareem Hegazy, Alexander Brace, Hongyi Lin, Murali Emani, Venkatram Vishwanath, Kevin Gering, Melisa Alkan, Tom Gibbs, Jack Wells, Lav R. Varshney, Bharath Ramsundar, Karthik Duraisamy, Michael W. Mahoney, Arvind Ramanathan, and Venkatasubramanian Viswanathan. Foundation Models for Discovery and Exploration in Chemical Space, October 2025. URL http://arxiv.org/abs/2510.18900.

Chengshi Wang, Yeon-Ju Kim, Aikaterini Vriza, Rohit Batra, Arun Baskaran, Naisong Shan, Nan Li, Pierre Darancet, Logan Ward, Yuzi Liu, Maria K. Y. Chan, Subramanian K.R.S. Sankaranarayanan, H. Christopher Fry, C. Suzanne Miller, Henry Chan, and Jie Xu. Autonomous platform for solution processing of electronic polymers. *Nature Communications*, 16(1):1498, February 2025a. ISSN 2041-1723. doi:10.1038/s41467-024-55655-3. URL https://www.nature.com/articles/s41467-024-55655-3.

Cunshi Wang, Yu Zhang, Yuyang Li, Xinjie Hu, Yiming Mao, Xunhao Chen, Pengliang Du, Rui Wang, Ying Wu, Hang Yang, Yansong Li, Beichuan Wang, Haiyang Mu, Xiaohan Chen, Shunxuan He, Hao Mo, Liyue Zhang, Lin Du, Yunning Zhao, Jianfeng Tian, Liang Ge, Yongna Mao, Shengming Li, Zheng Wang, Xiaomeng Lu, Jinhang Zou, Yang Huang, Ningchen Sun, Jie Zheng, Min He, Yu Bai, Junjie Jin, Hong Wu, and Jifeng Liu. StarWhisper Telescope: An AI framework for automating end-to-end astronomical observations. *Communications Engineering*, 4(1):184, November 2025b. ISSN 2731-3395. doi:10.1038/s44172-025-00520-4. URL https://www.nature.com/articles/s44172-025-00520-4.

Mengying Wang, Tao Liu, Xiaofan Du, Youlong Sun, Luohao Wang, Zengqi Zhang, Xiaogang Wang, and Guanglei Cui. Weakening solvent polarity enables shuttle-effect-free and temperature-independent lithium-organic batteries. *Energy Storage Materials*, 60:102816, June 2023. ISSN 24058297. doi:10.1016/j.ensm.2023.102816. URL https://linkinghub.elsevier.com/retrieve/pii/S2405829723001952.

Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. Executable Code Actions Elicit Better LLM Agents, June 2024. URL http://arxiv.org/abs/2402.01030.

Xingyu Wang, Wendong Xue, Kai Hu, Yan Li, Yong Li, and Rongyi Huang. Adiponitrile as Lithium-Ion Battery Electrolyte Additive: A Positive and Peculiar Effect on High-Voltage Systems. *ACS Applied Energy Materials*, pp. acsaem.8b00968, September 2018. ISSN 2574-0962, 2574-0962. doi:10.1021/acsaem.8b00968. URL https://pubs.acs.org/doi/10.1021/acsaem.8b00968.

Ziqi Wang, Hongshuo Huang, Hancheng Zhao, Changwen Xu, Shang Zhu, Jan Janssen, and Venkatasubramanian Viswanathan. DREAMS: Density Functional Theory Based Research Engine for Agentic Materials Simulation, July 2025c. URL http://arxiv.org/abs/2507.14267.

Yun Xu, Lingyu Wan, Jiali Liu, Lecai Zeng, and Zuoguo Yang. γ-butyrolactone and glutaronitrile as 5 V electrolyte additive and its electrochemical performance for LiN i0.5 Mn 1.5 O 4. *Journal of Alloys and Compounds*, 698:207–214, March 2017. ISSN 09258388. doi:10.1016/j.jallcom.2016.11.381. URL https://linkinghub.elsevier.com/retrieve/pii/S0925838816338646.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2210.03629.

E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, November 1999. ISSN 1089778X. doi:10.1109/4235.797969. URL http://ieeexplore.ieee.org/document/797969/.

A SciFM-Backed Tools

Table 1: Agents were provided a range of SciFM-backed tools for their task. Summary information for the attached SciFMs is provided below; for complete training details, see the linked model information on Hugging Face. MIST models were pretrained on Enamine REAL Space and fine-tuned on the indicated datasets (Wadell et al., 2025).

Model	Input	Output summary	Training dataset
🗃 `DiffUNet2 HEAT PLI jet design`	28 spline parameters $\mathbb{R}^{28}$	Jet quality metrics $\mathbb{R}^2$	HEAT PLI (Kaiser et al., 2026), see `JANUS first2last`, for details.
🤗 `MIST Tox21`	SMILES	12 toxicity endpoint classifications	Tox21
🤗 `MIST ClinTox`	SMILES	CT_TOX and FDA approval classification	ClinTox
🤗 `MIST QM8`	SMILES	12 electronic transition regression targets	QM8
🤗 `MIST QM9`	SMILES	12 quantum-mechanical property regressions	QM9
🤗 `MIST tmQM`	SMILES	8 quantum-mechanical property regressions	tmQM
🤗 `MIST BBBP`	SMILES	Blood-brain barrier penetration classification	BBBP
🤗 `MIST HIV`	SMILES	HIV replication activity classification	MoleculeNet HIV
🤗 `MIST BACE`	SMILES	BACE-1 inhibitor classification	BACE
🤗 `MIST SIDER`	SMILES	27 adverse-reaction classifications	SIDER
🤗 `MIST ESOL`	SMILES	Aqueous solubility regression	ESOL
🤗 `MIST FreeSolv`	SMILES	Hydration free energy regression	FreeSolv
🤗 `MIST Lipophilicity`	SMILES	logD lipophilicity regression	MoleculeNet Lipophilicity
🤗 `MIST ToxCast`	SMILES	617 toxicity assay classifications	ToxCast
🤗 `MIST MUV`	SMILES	17 bioassay target classifications	MUV

Continued on next page

Model	Input	Output summary	Training dataset
🤗 `MIST pKa`	SMILES	pKa regression	pKa in DMSO dataset from IUPAC technical report sources
🤗 `MIST Solvent Properties`	SMILES	Solvent property regression, including ETN	solvent-property datasets including Catalan and Kamlet–Taft parameters
🗄️🤗 `Janus first2last / DiffUNet2 HEAT PLI f2l`	HEAT-PLI `av_density` field at $t = 0.0\mu$s	HEAT-PLI `av_density` field at $t = 25.0\mu$s	HEAT-PLI `av_density` pairs from the HEAT dataset (Kaiser et al., 2026)

B METRICS AND STATISTICAL MEASURES

This section defines the metrics and statistical measures used throughout the manuscript.

Pareto hypervolume. For candidate objective vectors $\mathcal{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$, the dominated Pareto hypervolume is the measure of objective space dominated by at least one candidate and bounded by a reference point $\mathbf{r}$:

$$H(\mathcal{Y}; \mathbf{r}) = \lambda_d \left(\bigcup_{\mathbf{y} \in \mathcal{Y}_{\mathrm{ND}}} [r_1, y_1] \times \cdots \times [r_d, y_d] \right), \tag{4}$$

where $\mathcal{Y}_{\mathrm{ND}}$ is the nondominated subset and λ_d is the d-dimensional Lebesgue measure. Each raw objective x_j is normalized by a task-specific scale s_j and expressed so that larger values are preferred: $y_j = x_j / s_j$ for maximization objectives and $y_j = s_j / x_j$ for minimization objectives. We use the origin, $\mathbf{r} = \mathbf{0}$, as the reference point and discard candidates that do not dominate it. Thus, a single candidate with $x_j = s_j$ for every objective has $H = 1$, and larger H indicates better joint performance or broader Pareto-front coverage. Runs that return no valid candidate are assigned $H = 0$.

Relative risk. For treatment E, we report the posterior relative risk of rollout success against the baseline treatment:

$$\mathrm{RR}_{\mathrm{success}}(E) = \frac{p(H > 0 \mid \mathbf{x}_E)}{p(H > 0 \mid \mathbf{x}_0)}, \tag{5}$$

where $\mathbf{x}_E$ applies treatment E while holding all other factors at their baseline settings $\mathbf{x}_0$. A value below one indicates a lower probability of success than the baseline, one indicates equal success probabilities, and a value above one indicates a higher probability of success. The ratio is computed for every posterior draw of the Bernoulli component of the hurdle model, so its reported interval propagates posterior uncertainty.

Coefficient of determination. The coefficient of determination, R^2, measures the fraction of observed variance explained by a regression model:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \widehat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}, \tag{6}$$

where y_i is an observed value, $\widehat{y}_i$ is its prediction, and $\bar{y}$ is the sample mean. Values closer to one indicate better agreement; R^2 can be negative when predictions perform worse than the constant predictor $\widehat{y} = \bar{y}$.

Watanabe–Akaike information criterion (WAIC). The Watanabe–Akaike (or widely applicable) information criterion estimates a fitted model's expected out-of-sample predictive accuracy while accounting for its effective complexity (Gelman et al., 2014). For observation i, posterior draw s, and pointwise log likelihood $\ell_{is} = \log p(y_i \mid \theta^s)$, we compute

$$\mathrm{lppd}_i = \log \left(\frac{1}{S} \sum_{s=1}^{S} \exp(\ell_{is}) \right), \tag{7}$$

$$p_{\mathrm{WAIC},i} = \mathrm{Var}_{s=1,\ldots,S}(\ell_{is}), \tag{8}$$

$$\mathrm{WAIC}_i = -2 \left(\mathrm{lppd}_i - p_{\mathrm{WAIC},i} \right), \tag{9}$$

$$\mathrm{WAIC} = \sum_{i=1}^{n} \mathrm{WAIC}_i \,. \tag{10}$$

Lower values indicate better expected predictive fit, but WAIC values are comparable only when models use the same observations and likelihood terms. For the hurdle model, the pointwise likelihood includes the Bernoulli probability of success or failure and, for successful runs, the positive-outcome density.

Standard error of WAIC. The pointwise standard error reported with WAIC is

$$\mathrm{SE}(\mathrm{WAIC}) = \sqrt{n \ \mathrm{Var}_{i=1,\ldots,n}(\mathrm{WAIC}_i)}. \tag{11}$$

This measures sampling variation among observations in their contributions to WAIC; it is not the Monte Carlo error of the posterior sampler. For difference between models evaluated on the same observations, we computed the standard error of the ΔWAIC as:

$$\mathrm{SE}(\Delta\,\mathrm{WAIC}) = \sqrt{n \ \mathrm{Var}_{i=1,\ldots,n}(\Delta\,\mathrm{WAIC}_i)}. \tag{12}$$

Potential scale reduction factor. We use rank-normalized split-$\widehat{R}$ to assess convergence among m independent Markov chains (Vehtari et al., 2021). For the classical form:

$$\widehat{R} = \sqrt{\frac{\widehat{\mathrm{var}}^{+}(\psi \mid y)}{W}}, \qquad \widehat{\mathrm{var}}^{+}(\psi \mid y) = \frac{n-1}{n}W + \frac{1}{n}B, \tag{13}$$

$$B = \frac{n}{m-1}\sum_{j=1}^{m}\left(\bar{\psi}_{\cdot j} - \bar{\psi}_{\cdot\cdot}\right)^2, \qquad W = \frac{1}{m}\sum_{j=1}^{m} s_j^2, \tag{14}$$

where n is the number of post-warmup draws per chain, $\bar{\psi}_{\cdot j}$ and s_j^2 are the mean and variance within chain j, and $\bar{\psi}_{\cdot\cdot}$ is the mean across chains. Values near one indicate agreement between between-chain and within-chain variation. We required $\widehat{R} < 1.01$ for every scalar parameter for convergence (Vehtari et al., 2021; Gelman et al., 2014).

Effective Sample Size (ESS). Autocorrelation reduces the information in correlated posterior draws relative to the same number of independent draws. For m chains with n post-warmup draws per chain, the effective sample size is approximately

$$n_{\mathrm{eff}} = \frac{mn}{1 + 2\sum_{t=1}^{T}\widehat{\rho}_t}, \tag{15}$$

where $\widehat{\rho}_t$ is the estimated autocorrelation at lag t and the sum is truncated using a stable paired-lag criterion (Gelman et al., 2014; Vehtari et al., 2021). We report rank-normalized bulk and tail ESS: bulk ESS diagnoses sampling efficiency in the center of the posterior, whereas tail ESS diagnoses the reliability of posterior quantiles. Larger values are preferable; our convergence tables report the minimum over scalar parameters. The sampling duration was set to ensure acESS > 400 as recommended by Vehtari et al. (2021); Gelman et al. (2014).

Pearson's correlation coefficient. Pearson's correlation coefficient measures the linear relationship between variables X and Y:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}. \tag{16}$$

It ranges from -1 to 1, with zero indicating no linear correlation.

Spearman's correlation coefficient. Spearman's rank correlation, r_s, measures monotonic association by applying Pearson's correlation to ranked observations. When there are no tied ranks, it can be written as

$$r_s = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)}, \tag{17}$$

where d_i is the difference between the ranks of x_i and y_i.

C Statistical Model Specifications

This section documents the statistical models fitted in the analysis, including the expanded linear predictors generated by the current analysis code. Sampling and predictive-fit diagnostics for the primary hurdle and conditional-cost fits are reported in table 2.

Notation and coding. For rollout i, let L_i, A_i, D_i, C_i and T_i denote the model-matrix rows for the backing LLM, agent harness, access to historical data, PTC, and task, respectively. The categorical variables use treatment coding with GPT-5.4, URSA Execution, no historical data, and direct tool calling as reference levels. The PLI task is the reference level when task is included as a predictor. The symbol $\otimes$ denotes all pairwise products between the columns of two model-matrix terms; its extension to three terms denotes their three-way interaction. Terms with only one observed level and linearly dependent columns are removed before fitting, so the realized design matrix is full rank.

C.1 HYPERVOLUME HURDLE MODELS

Let $H_i \geq 0$ be the Pareto hypervolume for rollout i and $Z_i = \mathbb{I}(H_i > 0)$ indicate success. Every hurdle specification uses two regressions with the same design matrix but distinct coefficient vectors:

$$Z_i \sim \text{Bernoulli}\big(\text{logit}^{-1}(\eta_{s,i})\big) \qquad \eta_{s,i} = \mathbf{x}_i^\mathsf{T}\boldsymbol{\beta}_s \tag{18}$$

$$H_i \mid Z_i = 1 \sim \text{LogNormal}(\eta_{+,i}, \sigma) \qquad \eta_{+,i} = \mathbf{x}_i^\mathsf{T}\boldsymbol{\beta}_+ \tag{19}$$

Equivalently, $\log H_i = \eta_{+,i} + \epsilon_i$ for successful rollouts, where $\epsilon_i \sim \text{Normal}(0, \sigma^2)$. The priors are independent $\beta_{+,j} \sim \text{Normal}(0, 1)$, $\beta_{s,j} \sim \text{Normal}(0, 3)$, and $\sigma \sim \text{Exponential}(1)$.

Primary task-interaction model. The primary pooled model allows the effects of LLM, harness, historical data, and tool-use mechanism to vary by task. For either hurdle component, its expanded predictor is

$$\begin{aligned}\eta_i = \beta_0 &+ L_i\boldsymbol{\beta}_L + A_i\boldsymbol{\beta}_A + D_i\beta_D + C_i\beta_C + T_i\beta_T \\ &+ (L_i \otimes A_i)\boldsymbol{\beta}_{LA} + (L_i \otimes T_i)\boldsymbol{\beta}_{LT} + (A_i \otimes T_i)\boldsymbol{\beta}_{AT} \\ &+ (L_i \otimes A_i \otimes T_i)\boldsymbol{\beta}_{LAT} + (D_i \otimes T_i)\boldsymbol{\beta}_{DT} + (C_i \otimes T_i)\boldsymbol{\beta}_{CT}\end{aligned} \tag{20}$$

Equation (20) is used once with the success coefficients and once with the conditional-positive coefficients in eqs. 18 and 19. Here, β_0 is the expected value on the component's link scale for the reference configuration. The $L_i\boldsymbol{\beta}_L$, $A_i\boldsymbol{\beta}_A$, $D_i\beta_D$, $C_i\beta_C$, and $T_i\beta_T$ terms are the main effects of LLM, harness, historical data, PTC, and task. The LA interaction captures whether the effect of an LLM depends on its harness. The LT and AT interactions allow LLM and harness effects to differ between tasks, while the LAT interaction allows the joint LLM–harness effect to differ by task. Finally, the DT and CT interactions allow the effects of historical data and PTC to differ by task.

Pooled LLM–harness interaction model. The pooled validation model includes a common task effect but no interactions with task:

$$\eta_i = \beta_0 + L_i\boldsymbol{\beta}_L + A_i\boldsymbol{\beta}_A + (L_i \otimes A_i)\boldsymbol{\beta}_{LA} + D_i\beta_D + C_i\beta_C + T_i\beta_T \tag{21}$$

The intercept and five main-effect terms describe the reference configuration and the average shifts associated with LLM, harness, historical data, PTC, and task. The sole interaction, LA, captures dependence between LLM and harness choice; all other effects are constrained to be common across tasks and across levels of the remaining factors.

Task-specific LLM–harness interaction models. Separate models are fit to the PLI and molecular-design tasks. Within task t, the candidate predictor is

$$\eta_{i,t} = \beta_{0,t} + L_i\boldsymbol{\beta}_{L,t} + A_i\boldsymbol{\beta}_{A,t} + (L_i \otimes A_i)\boldsymbol{\beta}_{LA,t} + D_i\beta_{D,t} + C_i\beta_{C,t} \tag{22}$$

Within each task, $\beta_{0,t}$ is the reference prediction, the L, A, D, and C terms describe the effects of LLM, harness, historical data, and PTC. The LA term allows the effect of an LLM to depend on its harness. Because these models are fitted separately, every coefficient may differ between the two tasks even though task does not appear explicitly in the predictor.

Positive-only ablations. For each task, a positive-only model is fit to observations with $H_i > 0$ to test whether the Bernoulli hurdle changes inference for the conditional-positive distribution:

$$H_i \sim \text{LogNormal}(\eta_i, \sigma) \qquad \eta_i = \mathbf{x}_i^\mathsf{T}\boldsymbol{\beta} \tag{23}$$

where η_i has the task-specific interaction form in eq. 22, $\beta_j \sim \text{Normal}(0, 1)$, and $\sigma \sim \text{Exponential}(1)$. Thus, its L, A, D, C, P, and LA terms have the same interpretations as in the task-specific hurdle model, but describe only variation in positive hypervolume and do not model the probability of success.

Table 2: Sampling and predictive-fit diagnostics for the fitted Bayesian hurdle models.

Model	n	$p(H=0)$	$\mathrm{ESS}_{\min}$	$\widehat{R}_{\max}$	WAIC $\pm$ SE
Pooled hurdle, interaction	845	0.593	32736	1.000	1848.4 ± 74.9
Pooled hurdle, task interaction	845	0.593	36254	1.000	1822.4 ± 73.5
Pooled hurdls, task interaction, modes only	845	0.593	32297	1.000	2190.6 ± 73.3
Pooled hurdle, task interaction, additive LLM/harness)	845	0.593	27390	1.000	1890.1 ± 72.8
PLI hurdle, interaction	621	0.578	40799	1.000	1308.2 ± 59.8
Electrolyte Screening hurdle, interaction	224	0.634	25890	1.000	510.0 ± 39.4
Pooled hurdle, no interaction	845	0.593	25719	1.000	1910.5 ± 73.0
PLI hurdle, no interaction	621	0.578	29604	1.000	1367.6 ± 59.4
Electrolyte Screening hurdle, no interaction	224	0.634	24337	1.000	519.2 ± 39.7

n is the number of rollouts on which each model was fit. $p(H=0)$ is the observed fraction of rollouts with zero hypervolume. Reported ESS (eq. 15) is the minimum bulk effective sample size, and $\widehat{R}$ (eq. 14) is the maximum rank-normalized split $\widehat{R}$ across scalar parameters. Lower WAIC (eq. 10) indicates better expected predictive fit; reported uncertainty is the standard error computed from pointwise WAIC contributions (eq. 11). WAIC values are comparable only for models evaluated on the same observations using the same likelihood.

Table 3: Comparison of pooled hurdle model specifications.

Model	ΔWAIC $\pm$ SE
Pooled hurdle, task interaction	0.0 ± 0.0
Pooled hurdle, interaction	25.9 ± 15.6
Pooled hurdle, task interaction, additive LLM/harness)	67.6 ± 18.4
Pooled hurdle, no interaction	88.1 ± 23.6
Pooled hurdls, task interaction, modes only	368.2 ± 35.6

Reported ΔWAIC values are relative to the selected model (Pooled hurdle, task interaction). Positive values favor the selected model over the alternative. Standard errors are calculated from paired pointwise WAIC differences across the same observations.

C.2 Markov-Chain Monte Carlo Fitting and Model Checks

We fit several hurdle models over the course of our analysis (appendix C); posterior draws from all models are provided in our data release (doi:10.5281/zenodo.22818752). This section presents additional summary statistics for our primary model (section 3.3), along with diagnostics for all fitted models.

Table 2 provides both sampling and predictive-fit diagnostics for the fitted Bayesian hurdle models. All models were fit using six chains, each with a 2,000-draw warmup followed by 10,000 retained draws. All 60,000 posterior draws for each model are provided in our data release (doi:10.5281/zenodo.22818752). The ESS and $\widehat{R}$ diagnostics support adequate sampling of the posterior. The selected pooled hurdle model with task interactions has the lowest WAIC among the compared pooled specifications, indicating the best estimated predictive performance (Vehtari et al., 2017). Paired pointwise WAIC differences are reported in table 3.

Tables 4 and 5 report effect sizes and expectations under the primary model (section 3.3). Table 4 presents the effects shown in fig. 2 in tabular form. The reported summaries describe posterior uncertainty in treatment effect sizes and model expectations, not the posterior predictive distribution of future hypervolume observations H.

Table 4: Treatment effects by task: posterior mean [95% HDI].

Treatment	Relative Risk for Success $\frac{p(H>0\|\text{treatment})}{p(H>0\|\text{reference})}$			Conditional Pareto hypervolume change (%) $100\left(\frac{E[H\|H>0,\text{treatment}]}{E[H\|H>0,\text{reference}]}-1\right)$		
	All	PLI	Electrolyte Screening	All	PLI	Electrolyte Screening
Programmatic Tool Calling	1.5 [0.94, 2.3]	**1.8** **[1.3, 2.3]**	1.4 [0.50, 3.1]	42 [−18, 100]	**22** **[7.1, 37]**	72 [−32, 190]
Historical Data	0.92 [0.71, 1.1]	0.91 [0.69, 1.2]	—	**110** **[84, 130]**	**110** **[84, 130]**	—
Claude Sonnet 4.5	1.1 [0.63, 1.7]	**0.51** **[0.28, 0.77]**	1.8 [0.66, 3.6]	37 [−15, 95]	15 [−7.6, 37]	30 [−23, 88]
GPT-OSS 120B	**0.12** **[0.008, 0.32]**	**0.023** **[0.001, 0.057]**	**0.19** **[0.007, 0.49]**	**−42** **[−73, −5.5]**	**−80** **[−92, −65]**	**−42** **[−70, −11]**
GPT-5.5	1.2 [0.77, 1.8]	1.2 [0.63, 1.8]	1.3 [0.60, 2.3]	−0.39 [−26, 26]	11 [−16, 39]	−7.5 [−37, 26]
Gemma 4 31B IT	0.80 [0.43, 1.2]	**0.45** **[0.12, 0.81]**	1.1 [0.48, 1.9]	−9.0 [−35, 19]	6.7 [−30, 47]	−20 [−45, 8.6]
URSA: Plan Execute	0.97 [0.51, 1.6]	1.1 [0.67, 1.5]	1.0 [0.24, 2.4]	30 [−28, 92]	**30** **[7.2, 54]**	39 [−47, 150]
Codex	0.96 [0.61, 1.4]	1.4 [0.95, 2.0]	0.59 [0.098, 1.1]	−1.7 [−31, 28]	15 [−2.3, 33]	−7.4 [−48, 37]
Claude Code	0.59 [0.23, 1.0]	0.63 [0.29, 1.0]	0.63 [0.063, 1.5]	−11 [−52, 32]	3.6 [−21, 30]	−17 [−70, 49]

Estimates use the pooled hurdle model with task interactions (section 3.3). Each row changes only the named treatment, holding other factors at their reference settings: GPT-5.4, URSA Execution, direct tool calling, and no historical data. *All* denotes an equal-weight mixture of the two tasks. Values are rounded to at most two significant digits and at most three decimal places. Success probabilities are averaged across tasks; conditional-positive means are weighted by task success probabilities. A dash denotes an unavailable within-task setting (historical data for Electrolyte Screening). No effect corresponds to a success risk ratio of 1 and a 0% change in conditional Pareto hypervolume; larger values indicate improvements for both effects. Means and intervals are shown in bold when the interval strictly excludes the no-effect value.

Table 5: Treatment expectations by task: posterior mean [95% HDI].

Treatment	Probability of success $p(H > 0)$		Conditional expectation $E[H \mid H > 0]$	
	PLI	Electrolyte Screening	PLI	Electrolyte Screening
Baseline Treatment	0.43 [0.28, 0.59]	0.57 [0.18, 0.94]	1.5 [1.3, 1.7]	2.2 [0.88, 3.9]
Programmatic Tool Calling	0.74 [0.65, 0.83]	0.66 [0.45, 0.85]	1.8 [1.6, 2.0]	3.5 [2.6, 4.4]
Historical Data	0.39 [0.25, 0.54]	—	3.1 [2.6, 3.6]	—
Claude Sonnet 4.5	0.22 [0.11, 0.34]	0.86 [0.44, 1.0]	1.7 [1.4, 2.1]	2.9 [1.0, 5.3]
GPT-OSS 120B	0.010 [0.000, 0.026]	0.12 [0.001, 0.38]	0.31 [0.12, 0.53]	1.3 [0.39, 2.4]
GPT-5.5	0.50 [0.29, 0.73]	0.68 [0.28, 0.99]	1.7 [1.2, 2.1]	2.1 [0.79, 3.6]
Gemma 4 31B IT	0.19 [0.049, 0.36]	0.61 [0.21, 0.97]	1.6 [1.0, 2.2]	1.8 [0.67, 3.1]
URSA: Plan Execute	0.46 [0.30, 0.62]	0.46 [0.23, 0.69]	1.9 [1.6, 2.3]	2.8 [2.0, 3.6]
Codex	0.60 [0.45, 0.74]	0.35 [0.021, 0.77]	1.7 [1.4, 2.0]	2.1 [0.69, 3.8]
Claude Code	0.27 [0.12, 0.44]	0.29 [0.066, 0.55]	1.5 [1.2, 2.0]	1.7 [0.99, 2.4]

Estimates use the pooled hurdle model with task interactions (section 3.3). Each row changes only the named treatment, holding other factors at their reference settings: GPT-5.4, URSA Execution, direct tool calling, and no historical data. Values are rounded to at most two significant digits and at most three decimal places. These are absolute model expectations under each treatment, rather than the relative effect sizes reported in table 4.