EvoMaster: A Foundational Evolving Agent Framework for Agentic Science at Scale

Anonymous Authors[1]

Abstract

The convergence of large language models and agents is catalyzing a new era of scientific discovery: **Agentic Science**. While the scientific method is inherently iterative, existing agent frameworks are predominantly static, narrowly scoped, and lack the capacity to learn from trial and error. To bridge this gap, we present **EvoMaster**, a **foundational** evolving agent framework engineered specifically for **Agentic Science at Scale**. Driven by the core principle of **continuous self-evolution**, EvoMaster empowers agents to iteratively refine hypotheses, self-critique, and progressively accumulate knowledge across experimental cycles, faithfully mirroring human scientific inquiry. Crucially, as a domain-agnostic base harness, EvoMaster is exceptionally **easy to scale up**—enabling developers to build and deploy highly capable, self-evolving scientific agents for arbitrary disciplines in approximately 100 lines of code. Built upon EvoMaster, we incubated the SciMaster ecosystem across domains such as machine learning, physics, and general science. Evaluations on four authoritative benchmarks (Humanity's Last Exam, MLE-Bench Lite, BrowseComp, and FrontierScience) demonstrate that EvoMaster achieves state-of-the-art scores of 41.1%, 75.8%, 73.3%, and 53.3%, respectively. It comprehensively outperforms the general-purpose baseline OpenClaw with relative improvements ranging from +159% to +316%, robustly validating its efficacy and generality as the premier foundational framework for the next generation of autonomous scientific discovery.

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

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

1. Introduction

1.1. The Dawn of Agentic Science

Large language models with strong reasoning, planning, and tool-use capabilities (OpenAI, 2025; Anthropic, 2025a) are transforming the landscape of scientific research. A growing community consensus recognizes that AI agents represent the next frontier for accelerating scientific discovery (Gao et al., 2025). The 2024 Nobel Prizes in Physics and Chemistry, awarded to pioneers of artificial neural networks and computational protein design (The Royal Swedish Academy of Sciences, 2024b;a), signal that AI has moved to the center of scientific achievement. AlphaFold (Jumper et al., 2021) and AlphaFold 3 (Abramson et al., 2024) have revolutionized structural biology; GNoME (Merchant et al., 2023) discovered 2.2 million new crystal structures in a single sweep, equivalent to 800 years of human experimental work. Yet these achievements still position AI as a *tool* that answers specific questions posed by human researchers. The emergence of autonomous agents that can independently drive the full cycle of scientific research, from literature review and hypothesis generation, through experimental design and code execution, to result analysis and paper writing, marks the arrival of ***Agentic Science***. Early demonstrations such as Coscientist (Boiko et al., 2023) and The AI Scientist (Lu et al., 2024) confirm that this paradigm is imminently feasible.

If a single AI agent can conduct a complete research cycle in hours rather than months, the next imperative is to deploy such agents across dozens of scientific disciplines simultaneously. Scaling up not only extends coverage to more domains but also enables parallel execution that further accelerates discovery. We call this prospect ***Agentic Science at Scale***: the pace of scientific discovery is no longer bottlenecked by human bandwidth but by the quality of our AI agent architectures. This paper takes a foundational step toward that vision.

1.2. Challenges of Agentic Science at Scale

Despite its transformative promise, scaling Agentic Science from isolated demonstrations to a broadly applicable paradigm faces two critical challenges:

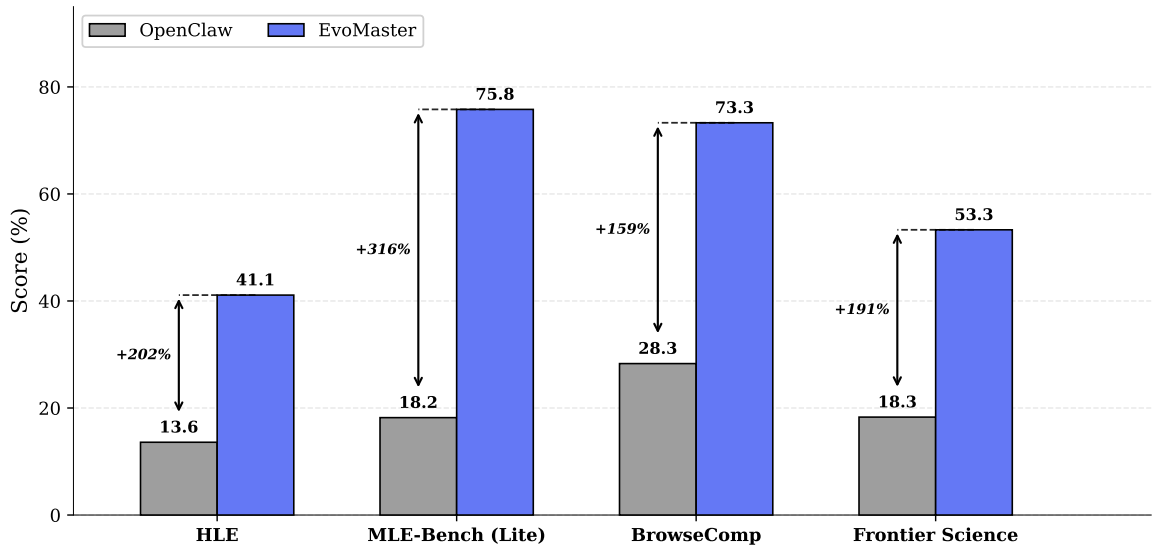

Figure 1. Performance comparison between EvoMaster and OpenClaw across four authoritative benchmarks. EvoMaster consistently and substantially outperforms the general-purpose baseline, with relative improvements ranging from +159% (BrowseComp) to +316% (MLE-Bench Lite). Both methods use GPT-5.4 as the backend model.

Fragmented, siloed development. Scientific research spans diverse disciplines. From chemistry and biology to materials science, each characterizes unique toolchains, data modalities, and evaluation protocols. Currently, most scientific agents are developed as monolithic, domain-specific systems: ChemCrow (Bran et al., 2024) is custom-tailored for chemical synthesis, while systems like MLAgentBench (Huang et al., 2023) are strictly coupled to machine learning pipelines. Such "bottom-up" engineering leads to a massive duplication of effort in common harness such as tool orchestration, trajectory management, and error recovery. This lack of a **foundational** abstraction layer ensures that advances in one domain remain non-transferable to others. Consequently, the marginal cost of supporting a new discipline remains prohibitively high, preventing the horizontal scaling of agentic capabilities across the broader scientific landscape.

Absence of evolution. Scientific discovery is inherently a long-horizon, trial-and-error process that aligns with the cycle of hypothesis, experimentation, and refinement. However, existing agent frameworks predominantly follow a single-pass paradigm: they execute a task once and terminate, lacking the mechanism to learn from failures or accumulate insights over successive attempts. This "stateless" execution contradicts the essence of the scientific method, where understanding is deepened through repeated iterations. Without the capacity for **continuous evolution**, agents remain static tools rather than self-improving researchers. They struggle to navigate complex, open-ended scientific frontiers that require the agent to progressively refine its strategies and "evolve" its cognitive boundaries through experimental feedback.

1.3. EvoMaster: A Foundational Evolving Agent Framework for Agentic Science at Scale

To address these challenges, we propose **EvoMaster**, a foundational evolving agent framework engineered specifically for *Agentic Science at Scale*. EvoMaster is driven by two core tenets: serving as a **foundational** harness that unifies agent development across diverse disciplines, and driving **continuous evolution**, allowing agents to iteratively refine strategies and accumulate experience just as human researchers do. EvoMaster translates them directly into four architectural design principles:

- **Modular composability**: To provide a truly foundational platform, the system is decoupled into independently replaceable components managed by unified registries. This standardizes harness, enabling the seamless onboarding of new scientific domains with minimal marginal cost and fostering organic ecosystem growth.
- **Experiment-ready harness**: Agent behavior is governed by declarative configurations paired with comprehensive trajectory tracking. This guarantees the parameter agility and execution traceability essential for rigorous, cross-domain scientific experimentation.
- **Iterative self-evolution**: Breaking away from single-pass execution, EvoMaster agents operate within multi-turn reactive loops. Equipped with intelligent context management, agents continuously observe, self-critique, and refine their hypotheses over long horizons, faithfully mirroring the iterative scientific method.
- **Multi-agent collaborative evolution**: The orchestration layer supports flexible collaboration patterns, enabling agent teams to collectively debate and optimize solutions, simulating the peer-driven dynamics of real-world scien-

tific discovery.

Built upon these principles and EvoMaster, we have successfully incubated the **SciMaster** (Zhang et al., 2025) agent ecosystem, which spans autonomous machine learning (ML-Master 2.0) (Liu et al., 2025; Zhu et al., 2026), general scientific research (X-Master / X-Master 2.0) (Chai et al., 2025), web information retrieval (Browse-Master) (Pang et al., 2025), physics reasoning (PhysMaster) (Miao et al., 2025), and embodied intelligence training (EmboMaster) (Lei et al., 2026). By leveraging a shared foundational harness, we are rapidly expanding this ecosystem to additional scientific domains.

To comprehensively validate the scientific and evolutionary capabilities of EvoMaster, we conduct head-to-head comparisons against OpenClaw (OpenClaw Contributors, 2025), the fastest-growing open-source general AI agent. Rather than merely evaluating general intelligence, we deliberately selected four authoritative benchmarks that map directly to the core competencies required for Agentic Science:

- **BrowseComp** evaluates deep, multi-step information retrieval. Since comprehensive literature review and data sourcing are the bedrock of research, EvoMaster's superior performance (73.3%, **+159%** relative gain) demonstrates its ability to autonomously navigate the web to extract and synthesize high-quality scientific context.
- **MLE-Bench Lite** tests long-horizon machine learning engineering. Scientific discovery is inherently a prolonged, trial-and-error process. EvoMaster's state-of-the-art 75.8% medal rate (**+316%**) proves its structural capacity to sustain complex, multi-step experimental workflows over extended horizons without compounding errors.
- **HLE** assesses cross-disciplinary expert knowledge. Modern breakthroughs often occur at the intersection of fields; achieving 41.1% accuracy (**+202%**) confirms EvoMaster possesses the robust, multidisciplinary cognitive foundation necessary for cross-domain hypothesis generation.
- **FrontierScience** directly measures advanced reasoning across physics, chemistry, and biology. EvoMaster's strong performance (53.3%, **+191%**) serves as direct evidence that it can successfully resolve domain-specific, PhD-level scientific problems.

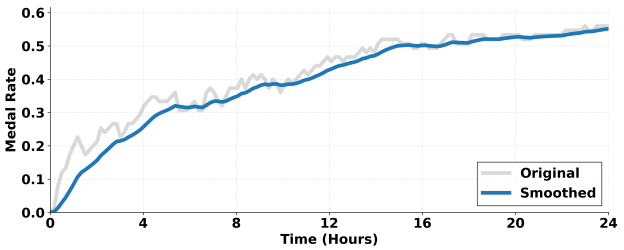

Figure 2. Evolving performance improvement of EvoMaster on MLE-Bench over time.

Specifically, as illustrated in Figure 2, EvoMaster demonstrates continuous self-evolution on the long-horizon MLE-Bench, progressively reaching superior accuracy. Together, these results systematically confirm that a domain-aware, evolution-driven framework comprehensively outperforms general-purpose platforms, effectively bridging the gap between isolated tools and fully autonomous scientific discovery.

1.4. Contributions

Our main contributions are:

- **The EvoMaster foundational agent framework.** We propose EvoMaster, a foundational evolving agent framework for agentic science at scale that enables new domain agents to be built in approximately 100 lines of code. EvoMaster also supports being invoked as a skill for other agents.
- **The SciMaster agent ecosystem.** We build 6+ autonomous research agents across diverse scientific domains on top of EvoMaster, facilitating a shared foundational harness enables rapid scaling to new disciplines.
- **Comprehensive experimental validation.** On four authoritative benchmarks, EvoMaster consistently outperforms OpenClaw with relative improvements of 159% to 316%, confirming the effectiveness of an evolution-driven foundational framework for scientific agents.

2. Related Work

2.1. Agent Frameworks

The landscape of LLM agent development is currently dominated by two paradigms. General-purpose orchestration frameworks like LangChain (LangChain, 2025a) and LangGraph (LangChain, 2025b) provide foundational graph execution, while ecosystem-specific tools such as the OpenAI (OpenAI, 2025), Claude (Anthropic, 2025a), and Google ADK (Google, 2025) SDKs prioritize production-grade integration with proprietary models. In the open-source community, specialized frameworks have achieved significant traction: OpenHands (Wang et al., 2025a;b) optimizes software engineering via event-sourced state models, while OpenClaw (OpenClaw Contributors, 2025) leverages an extensive skill ecosystem for rapid adoption.

Despite their success, these frameworks are primarily tailored for software development or general automation. They lack systematic support for the unique demands of scientific research, such as long-running experiment management and domain-specific knowledge injection. **EvoMaster** addresses this gap as a *foundational* and *evolving* framework, providing a cross-disciplinary base that aligns with the iterative nature of scientific discovery and scales seamlessly across diverse research domains.

2.2. Scientific Agents

The evolution of scientific agents has transitioned from early automated systems like the Robot Scientist (King et al., 2009) to modern LLM-driven architectures. Domain-specific agents have shown remarkable results: ChemCrow (Bran et al., 2024) and Coscientist (Boiko et al., 2023) automate chemical synthesis and robotic control, while specialized systems like Virtual Lab (Swanson et al., 2024) and PaperQA2 (Skarlinski et al., 2024) excel in nanobody design and literature synthesis. Recent breakthroughs such as Google's AI Co-Scientist (Gottweis et al., 2025) and AI Scientist v2 (Yamada et al., 2025) further demonstrate the potential for agents to surpass human-level performance in hypothesis generation and peer-reviewed paper production. In machine learning, MLAgentBench (Huang et al., 2023) and ML-Master (Liu et al., 2025) pioneered autonomous experimentation, supported by benchmarks like ScienceAgentBench (Chen et al., 2025).

However, most existing scientific agents are implemented as end-to-end systems, leading to redundant engineering in tool and harness management. **EvoMaster** distinguishes itself by serving as the underlying framework layer. By abstracting essential scientific workflows, it enables developers to build self-evolving agents that iterate through experiments, effectively bridging the gap between bespoke scientific tools and a universal, scalable research harness.

3. Design Principles

EvoMaster's architecture is driven by the practical demands of scientific discovery. To realize our two core tenets—serving as a **foundational** framework and enabling continuous **evolution**—we establish four design principles. *Modular Composability* and *Experiment-Ready Harness* provide a foundational, scalable platform across disciplines, while *Iterative Self-evolution* and *Multi-Agent Collaborative Evolution* drive the evolutionary dynamics inherent to the scientific method.

3.1. Modular Composability

Monolithic agent systems couple decision logic with domain-specific tools, resisting cross-domain adaptation. EvoMaster achieves foundational flexibility by decoupling the architecture into orthogonal components across three layers: *Playground*, *Experiment* and *Agent*.

Through unified registries and adherence to industry standards like the Model Context Protocol (MCP) (Anthropic, 2025c) and Skill (Anthropic, 2025b), developers can seamlessly swap models, environments, or toolsets. This composability drastically reduces the barrier to entry, enabling the deployment of new domain-specific scientific agents in approximately 100 lines of code.

3.2. Experiment-Ready Harness

Scientific research demands rigorous parameter control and absolute reproducibility. EvoMaster treats experimentation as a first-class primitive through two core mechanisms. First, a *configuration-driven design* utilizes YAML-based manifests to manage agent parameters, prompts, and tool configurations dynamically, ensuring parameter agility without altering source code.

Second, *comprehensive observability* is achieved via a robust trajectory recording system. This system meticulously logs every conversational turn, tool invocation, and token statistic into thread-safe structured JSON. Just as researchers maintain strict laboratory notebooks, this guarantees that multi-turn agent experiments remain fully auditable and reproducible.

3.3. Iterative Self-Evolving

Unlike traditional agents that execute tasks linearly, scientific discovery relies on iterative cycles of hypothesis, observation, and revision. EvoMaster embeds this self-evolving loop directly into the Agent Engine.

Operating within a multi-turn reactive architecture, agents continuously execute tools, observe experimental outcomes, and explicitly self-critique before determining the next action. To sustain these long-horizon evolutionary loops, EvoMaster integrates an intelligent Context Manager featuring dynamic LLM-based summarization and context compression, preventing context degradation over hundreds of experimental turns.

3.4. Multi-Agent Collaborative Evolution

Complex scientific problems frequently exceed the capacity of a single agent. EvoMaster's Playground orchestration layer simulates real-world interdisciplinary collaboration by supporting dynamic multi-agent topologies.

By declaratively defining Agent Slots, developers can orchestrate specialized agent teams. For example, when a critic agent identifies flaws in a solver's reasoning, the resulting revision represents a collaborative co-evolution, elevating the system's collective problem-solving capability beyond individual limits.

4. Architecture

EvoMaster's architecture translates our design principles into a concrete system built specifically for the demands of scientific discovery. To break down domain silos and facilitate rapid **scale-up**, the framework is decoupled across execution and capability layers, ensuring a **foundational** harness that is seamlessly reusable across disciplines. To

mirror the scientific method, the core engine and orchestration layers are fundamentally designed around **continuous evolution**, enabling agents to iteratively refine hypotheses, learn from feedback, and collaborate. As illustrated in Figure 3, this architecture empowers developers to deploy self-evolving scientific agents with minimal overhead, maintaining strict correspondence between identified research bottlenecks and our structural solutions.

4.1. Execution Architecture: Scaling Through Decoupling

To overcome fragmented, siloed development and enable horizontal scaling across domains, EvoMaster structurally realizes *Modular Composability* by separating execution into three orthogonal layers:

- **Playground (Orchestration):** Coordinates multi-agent collaboration patterns and domain-specific scientific workflows.
- **Exp (Experiment Execution):** Manages the single-experiment lifecycle, including task instantiation and trajectory recording.
- **Agent (Intelligence):** Drives the iterative reasoning and tool-use loop.

This separation guarantees that foundational improvements such as enhanced context management in the Agent Engine benefit all scientific domains simultaneously. Expanding to a new discipline simply requires defining a new Playground, leaving the underlying execution and reasoning logic untouched.

4.2. Agent Engine: The Evolution Core

Traditional agents execute tasks in a single pass, contradicting the trial-and-error nature of scientific discovery. Addressing this, the Agent Engine serves as the realization of *Iterative Self-Evolving*. Driven by the `BaseAgent` abstraction, it executes a multi-turn reactive loop: *reason* $\rightarrow$ *invoke tools* $\rightarrow$ *observe* $\rightarrow$ *self-critique*.

Since scientific tasks often span hundreds of interaction turns, maintaining this evolutionary loop requires robust memory. The engine integrates an intelligent `ContextManager` that prevents context degradation using dynamic LLM-based summarization and sliding windows. This ensures agents can accumulate insights and refine strategies over long experimental horizons without exceeding context limits.

4.3. Capability Layer: Universal Tools and Skills

This layer is the architectural realization of *Modular Composability* at the capability level, providing the extensible mechanisms through which agents interact with scientific tools, domain knowledge, and language models. It is this layer that makes EvoMaster foundational: by providing a universal capability interface, it ensures that domain-specific extensions developed for one scientific discipline are immediately available to all others.

- **Tool System:** Utilizing an Action-Execution-Observation pattern, this system natively integrates with the Model Context Protocol (MCP) (Anthropic, 2025c). External MCP-compatible scientific tools are instantly converted into standard EvoMaster tools, unlocking a massive, cross-disciplinary tool ecosystem.
- **Skill System:** Adopting emerging skill specifications (Anthropic, 2025b), domain knowledge is injected hierarchically. Metadata remains in-context for agent awareness, while detailed executable instructions are loaded on-demand to optimize context efficiency.
- **LLM Abstraction:** A unified interface standardizes responses across 100+ models (BerriAI, 2024), allowing researchers to seamlessly swap backend models for controlled experimentation without altering agent logic.

4.4. Playground Orchestrator: Collaborative Evolution

Complex scientific problems frequently exceed single-agent capacities. The Playground orchestrator addresses this by materializing *Multi-Agent Collaborative Evolution*. Developers use the `AgentSlots` mechanism to declaratively assign specialized roles (e.g., *solver, critic, rewriter*), each maintaining independent LLM and tool configurations.

The orchestrator natively supports diverse collaboration patterns, including sequential handoffs, parallel exploration, and iterative peer-review cycles. Through the `@register_playground` decorator, custom workflow logic is automatically discovered at runtime. By reusing the shared foundational harness, developing a complex multi-agent scientific team is reduced to approximately 100 lines of orchestration code.

4.5. Experiment Harness: Rigor and Reproducibility

Scientific research demands absolute parameter control and auditability. Embodying the *Experiment-Ready Harness* principle, EvoMaster treats every agent execution as a rigorous, reproducible experiment.

All agent behaviors, prompts, and environment parameters are managed dynamically via YAML-based **Configuration Manifests**, allowing researchers to share complete experimental setups just like laboratory protocols. Furthermore, the **Trajectory System** acts as an automated lab notebook, meticulously logging every multi-turn dialogue, tool invocation, and token statistic into thread-safe, structured JSON. This guarantees that long-horizon evolutionary loops remain fully auditable and reproducible.

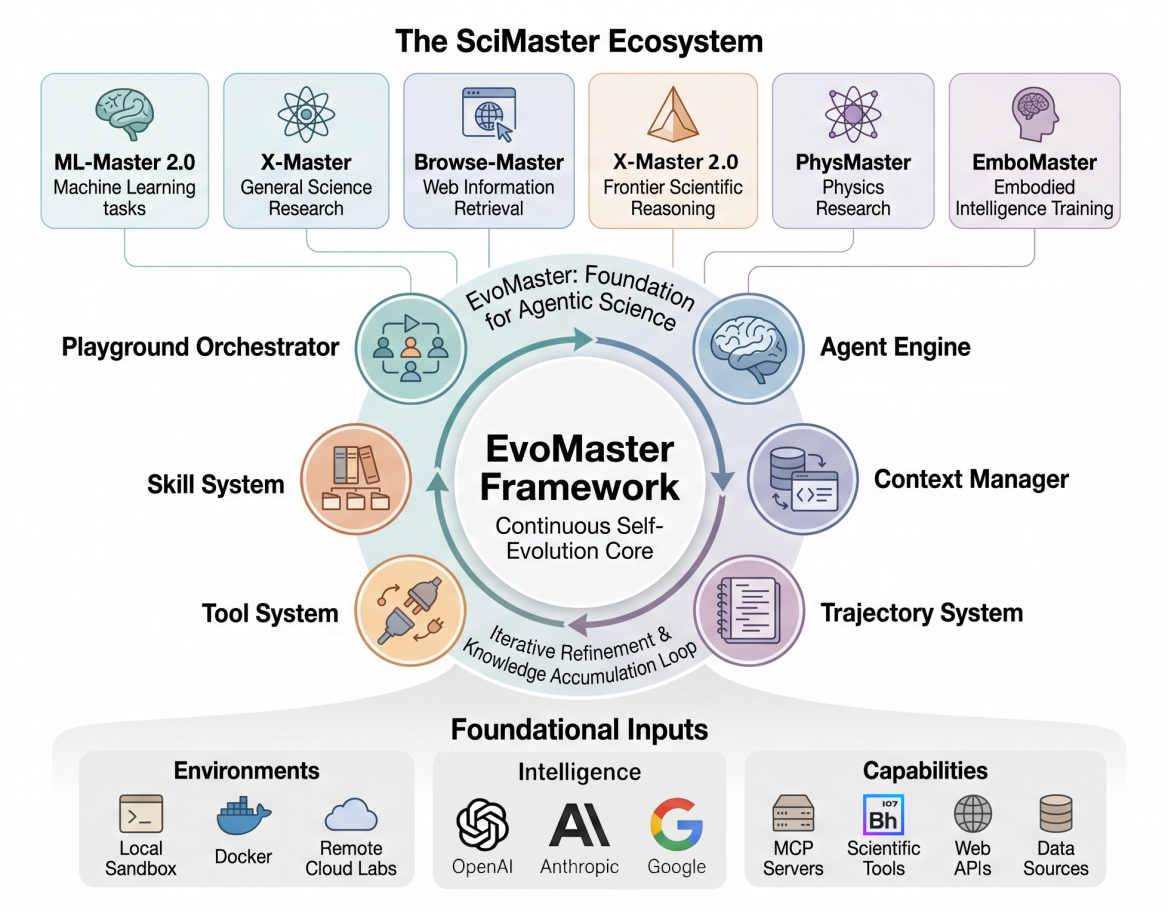

Figure 3. Overall architecture of EvoMaster.

Table 1. The SciMaster agent ecosystem built on EvoMaster.

Agent	Domain	Code Status
ML-Master 2.0 (Zhu et al., 2026)	Autonomous Machine Learning	Open-sourced
ML-Master (Liu et al., 2025)	Autonomous Machine Learning	Open-sourced
X-Master (Chai et al., 2025)	General Scientific Research	Open-sourced
Browse-Master (Pang et al., 2025)	Web Information Retrieval	Open-sourced
X-Master 2.0	Frontier Scientific Reasoning	Coming soon
PhysMaster (Miao et al., 2025)	Physics Research & Reasoning	Coming soon
EmboMaster (Lei et al., 2026)	Embodied Intelligence Training	Coming soon

5. The SciMaster Agent Ecosystem

Built on the EvoMaster framework, we have constructed a growing ecosystem of autonomous research agents, collectively known as the SciMaster series (Table 1). The ecosystem has already scaled to cover several distinct scientific domains, from autonomous machine learning to embodied intelligence training, and we are actively expanding to additional disciplines.

The rapid expansion of the SciMaster ecosystem underscores the profound advantages of building domain-specific agents atop the EvoMaster foundational harness. Rather than developing isolated systems from scratch, incubating agents within this framework yields three critical architectural benefits, which reduces the time required for developers to build specialized agents from several days to mere hours:

- **Drastic Reduction in Engineering Overhead.** By inheriting EvoMaster's unified Agent Engine, context management, and Playground orchestrator, developers avoid redundant harness coding.
- **Shared Evolutionary Upgrades.** Since the execution logic is completely decoupled from domain knowledge, core algorithmic enhancements immediately benefit the entire ecosystem.
- **Cross-Domain Tool Interoperability.** Leveraging the standardized Capability Layer, tools developed for one discipline become universally accessible. A specialized skill or script can be seamlessly invoked to another agent, fostering genuine interdisciplinary cross-pollination.

By centralizing the complex machinery of continuous evolution and orchestration, EvoMaster frees researchers to focus exclusively on domain-specific scientific logic, proving itself as the optimal and highly scalable substrate for the next generation of Agentic Science.

6. Experiments

6.1. Benchmarks

We evaluate EvoMaster on four authoritative benchmarks spanning diverse capability dimensions: **HLE** (Phan et al., 2025) for cross-disciplinary expert knowledge, **MLE-Bench (Lite)** (Chan et al., 2024) for ML engineering, **BrowseComp** (Wei et al., 2025) for deep web retrieval, and **FrontierScience** (Wang et al., 2026) for frontier scientific reasoning across physics, chemistry, and biology. Detailed benchmark descriptions are provided in Appendix A.

6.2. Experimental Setup

We compare EvoMaster against **OpenClaw** (OpenClaw Contributors, 2025), one of the fastest-growing open-source AI agent projects in 2025–2026. OpenClaw supports 100+ built-in skills and multi-platform integration, and serves as a representative example of recent general-purpose AI agent systems.

In our experiments, EvoMaster uses its standard configuration to run SciMaster series agents tailored to each benchmark. Specifically:

- **HLE**: X-Master (Chai et al., 2025), configured for cross-disciplinary scientific reasoning.
- **MLE-Bench (Lite)**: ML-Master 2.0 (Zhu et al., 2026), with its multi-phase iterative optimization pipeline featuring research-driven parallel improvement and hierarchical cognitive caching architecture.
- **BrowseComp**: Browse-Master (Pang et al., 2025), using a Planner-Executor iterative loop for progressive information retrieval.
- **FrontierScience**: X-Master 2.0, a single-agent solver enhanced with academic retrieval tools for frontier scientific reasoning.

To ensure a fair comparison, all experiments were conducted on an NVIDIA RTX 4090 GPU server. Both approaches utilize GPT-5.4 as the backend model and are equipped with identical tools and skills. Additionally, we imposed a strict 24-hour runtime limit for the MLE-Bench evaluation.

6.3. Results and Analysis

Table 2 presents the comparison between EvoMaster and OpenClaw across four benchmarks.

MLE-Bench (+316%). EvoMaster achieves 75.8% compared to OpenClaw's 18.2%, the largest margin among all benchmarks. EvoMaster earns medals on approximately 17 of 22 Kaggle competitions, while OpenClaw manages only 4 despite submitting on 18. The gap stems from EvoMaster's multi-phase iterative workflow: knowledge prefetch retrieves past insights, a drafting stage generates an initial solution, then up to 20 rounds of research-driven parallel improvement progressively refine it. A hierarchical cognitive caching mechanism (spanning prefetch, round-level knowledge promotion, and run-level wisdom promotion) accumulates reusable experience across cycles, embodying the evolution philosophy at the framework's core.

Table 2. Benchmark comparison between EvoMaster and OpenClaw. EvoMaster consistently outperforms OpenClaw across all four benchmarks with relative improvements ranging from 159% to 316%.

Benchmark	OpenClaw	EvoMaster	Rel. Improvement
HLE	13.6	**41.1**	+202%
MLE-Bench (Lite)	18.2	**75.8**	+316%
BrowseComp	28.3	**73.3**	+159%
FrontierScience	18.3	**53.3**	+191%

BrowseComp (+159%). EvoMaster achieves 73.3% versus OpenClaw's 28.3% on the BrowseComp benchmark. Through a Planner-Executor iterative loop (up to 10 rounds), the planner formulates targeted search plans based on accumulated findings, while the executor retrieves information via web search, URL fetching, and PDF extraction. This iterative deepening allows progressive query refinement and cross-source validation, whereas OpenClaw settles for the first plausible answer. The impact of this pipeline is evident across all question categories. EvoMaster consistently outperforms OpenClaw, with the most striking gain in Map + Search tasks (100.0% vs 25.0%, +75.0%). This suggests that the iterative planning process effectively resolves the geographical reasoning bottlenecks inherent in complex retrieval. Furthermore, significant improvements in Niche Knowledge (+52.9%) and Multi-step Reasoning (+43.8%) highlight the framework's versatility, proving that structured agentic coordination can robustly bridge the gap between base model capability and expert-level information retrieval requirements.

FrontierScience (+191%). EvoMaster achieves 53.3% compared to OpenClaw's 18.3% on the Research track (open-ended subtasks). A single-agent architecture enhanced with academic retrieval tools (Google Scholar, Semantic Scholar API, PDF reader) enables iterative tool-driven reasoning: the agent retrieves relevant literature, reflects on findings, and refines its understanding before synthesizing answers grounded in verifiable sources.

HLE (+202%). EvoMaster achieves 41.1% accuracy vs. 13.6% for OpenClaw, driven by its four-phase parallel pipeline (Solve, Critique, Rewrite, Select), which enables structured multi-agent refinement. Solver agents generate candidates, critics identify errors, and rewriters iteratively improve outputs before selection. This process is especially effective on hard HLE instances where initial solutions are weak. EvoMaster outperforms OpenClaw across all domains, with the largest gain in Mathematics (48.16%, +33.10%), followed by Computer Science (+24.55%) and

Table 3. **Evaluation Results on MLE-Bench(Lite).** We compare EvoMaster against OpenClaw, MLE-STAR-Pro-1.5 (Nam et al., 2025), and R&D-Agent (Yang et al., 2025) on MLE-Bench Lite. The best results in each metric are highlighted in **bold**. EvoMaster and OpenClaw use GPT-5.4 as the backend model and have a maximum runtime of 24 hours. We use the results of MLE-STAR-Pro-1.5 and R&D-Agent reported by official MLE-Bench.

Metric	OpenClaw (Baseline)	MLE-STAR	R&D-Agent	EvoMaster (Ours)	Δ (Gain)
Valid	81.82	**100.00**	77.27	**100.00**	+18.18
Above Median	31.82	75.76	74.24	**84.85**	+53.03
Bronze	4.55	**16.67**	12.12	13.64	+9.09
Silver	13.64	16.67	22.73	**31.82**	+18.18
Gold	0.00	**34.85**	33.33	30.30	+30.30
Any Medal	18.18	68.18	68.18	**75.76**	**+57.58**

Table 4. **Evaluation Results on BrowseComp.** We compare the accuracy (%) of EvoMaster against OpenClaw across major subject categories. The best results in each category are highlighted in **bold**. Both methods use GPT-5.4 as the backend model.

Category	OpenClaw (Baseline)	EvoMaster (Ours)	Δ (Gain)
Multi-step Reasoning	18.75	**65.63**	+46.88
Map + Search	25.00	**75.00**	+50.00
Niche Knowledge	47.05	**88.23**	+41.18
Complex Filtering	28.57	**71.43**	+42.86
Overall	**28.33**	**73.33**	**+45.00**

Table 5. **Evaluation Results on Frontier Science Research.** We compare the average normalized score (%) of OpenClaw, GPT-5.4, Muse Spark, and EvoMaster across major scientific subject categories. The best results in each category are highlighted in **bold**. OpenClaw and EvoMaster use GPT-5.4 as the backend model. Due to constraints in time and computational resources, we currently report results from a single run.

Category	OpenClaw (Baseline)	GPT-5.4 (medium)	Muse Spark	EvoMaster (Ours)	Δ (Gain)
Physics	15.00	-	-	**55.00**	+40.00
Chemistry	20.00	-	-	**55.00**	+35.00
Biology	20.00	-	-	**50.00**	+30.00
Overall	18.33	33.00	38.30	**53.33**	**+35.00**

Humanities (+30.05%), indicating strong cross-domain generalization. Overall, structured agent coordination significantly enhances performance over the base model, particularly for multi-step reasoning tasks.

In conclusion, EvoMaster substantially outperforms OpenClaw across all four benchmarks, with relative improvements ranging from 159% (BrowseComp) to 316% (MLE-Bench Lite). This consistent advantage across fundamentally different task types—closed-book knowledge, ML engineering, web retrieval, and scientific reasoning—demonstrates that a foundational agent framework with evolving capabilities provides systematic benefits over general-purpose platforms.

7. Conclusion

In this paper, we introduced **EvoMaster**, a foundational and evolving agent framework designed to catalyze Agentic Science at Scale. Addressing the limitations of siloed and static agent architectures, EvoMaster is built upon the dual pillars of foundational modularity and continuous self-evolution. By deeply integrating iterative refinement loops, experiment-ready harness, and multi-agent collaborative workflows, EvoMaster mirrors the authentic scientific method—enabling agents to autonomously hypothesize, experiment, self-critique, and continuously evolve from trial and error.

Table 6. **Evaluation Results on HLE (Humanity's Last Exam).** We compare the accuracy (%) of EvoMaster against OpenClaw, GPT-5.4 xhigh, and Muse Spark across major subject categories. The best results in each category are highlighted in **bold**. Both methods use GPT-5.4 as the backend model.

Category	OpenClaw (Baseline)	GPT-5.4 (xhigh)	Muse Spark	EvoMaster (Ours)	Δ (Gain)
STEM					
Biology / Medicine	14.86	-	-	**29.28**	+14.42
Chemistry	11.88	-	-	**29.70**	+17.82
Computer Science / AI	12.50	-	-	**37.05**	+24.55
Engineering	7.81	-	-	**21.88**	+14.07
Math	15.06	-	-	**48.16**	+33.10
Physics	13.86	-	-	**31.68**	+17.82
Social Science & Humanities					
Humanities / Social Science	13.99	-	-	**44.04**	+30.05
Others					
Other Categories	7.95	-	-	**43.18**	+35.23
Overall	13.62	36.47	40.92	**41.10**	**+27.48**

Our comprehensive evaluation across four authoritative benchmarks demonstrates that EvoMaster consistently and substantially outperforms the general-purpose framework OpenClaw, achieving relative performance improvements ranging from 159% to 316%. Furthermore, the successful incubation of the **SciMaster** ecosystem validates EvoMaster's cross-disciplinary scalability. By standardizing the foundational abstractions, the framework empowers researchers to rapidly deploy self-improving scientific agents across diverse domains with minimal engineering overhead.

Looking ahead, we envision EvoMaster serving as the foundational substrate for autonomous scientific discovery. As we actively expand the framework to encompass broader disciplines—such as biochemistry, materials science, and complex physical systems—our ultimate goal is to shift the bottleneck of scientific progress from human bandwidth to scalable, self-evolving artificial intelligence.

Impact Statement

This paper presents work whose goal is to advance autonomous scientific discovery through evolving agent frameworks. Potential positive impacts include accelerating routine computational research and making scientific workflow automation more reusable; potential risks include overreliance on agent-generated conclusions, misuse of autonomous search and experimentation tools, and concentration of computational advantages. We view transparent trajectories, reproducible configurations, and human oversight as necessary safeguards when deploying such systems.

References

Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A. J., Bambrick, J., et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, 630: 493–500, 2024.

Anthropic. Claude code and agent SDK. https://docs.anthropic.com/en/docs/agents-and-tools, 2025a.

Anthropic. Agent skills: Overview. https://platform.claude.com/docs/en/agents-and-tools/agent-skills/overview, 2025b.

Anthropic. Model context protocol (MCP). https://modelcontextprotocol.io, 2025c.

BerriAI. LiteLLM: Call 100+ LLM APIs in OpenAI format. https://github.com/BerriAI/litellm, 2024.

Boiko, D. A., MacKnight, R., Kline, B., and Gomes, G. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023.

Bran, A. M., Cox, S., Schilter, O., Baldassari, C., White, A. D., and Schwaller, P. Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 6 (5):525–535, 2024.

Chai, J., Tang, S., Ye, R., et al. SciMaster: Towards general-purpose scientific AI agents, Part I. X-Master as foundation: Can we lead on Humanity's Last Exam? *arXiv preprint arXiv:2507.05241*, 2025.

Chan, J. S., Chowdhury, N., Jaffe, O., Aung, J., Sherburn, D., Mays, E., Starace, G., Liu, K., Maksin, L., Patil, T., et al. MLE-Bench: Evaluating machine learning agents on machine learning engineering. *arXiv preprint arXiv:2410.07095*, 2024.

Chen, Z., Chen, S., Ning, Y., et al. ScienceAgentBench: Toward rigorous assessment of language agents for data-driven scientific discovery. In *The Thirteenth International Conference on Learning Representations*, 2025. arXiv:2410.05080.

Gao, L. et al. Accelerating scientific discovery with AI agents: A community perspective. *arXiv preprint arXiv:2501.04227*, 2025.

Google. Agent development kit (ADK). https://google.github.io/adk-docs/, 2025.

Gottweis, J. et al. Towards an AI co-scientist. *arXiv preprint arXiv:2502.18864*, 2025.

Huang, Q., Vora, J., Liang, P., and Leskovec, J. MLAgentBench: Evaluating language agents on machine learning experimentation. *arXiv preprint arXiv:2310.03302*, 2023.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021.

King, R. D., Rowland, J., Oliver, S. G., Young, M., Aubrey, W., Byrne, E., Liakata, M., Markham, M., Pir, P., Soldatova, L. N., et al. The automation of science. *Science*, 324(5923):85–89, 2009.

LangChain. LangChain: Build context-aware reasoning applications. https://www.langchain.com/, 2025a.

LangChain. LangGraph: Build stateful, multi-actor applications with LLMs. https://langchain-ai.github.io/langgraph/, 2025b.

Lei, Z., Liu, G., et al. EmboCoach-Bench: Benchmarking AI agents on developing embodied robots. *arXiv preprint arXiv:2601.21570*, 2026.

Liu, Z., Cai, Y., Zhu, X., et al. ML-Master: Towards AI-for-AI via integration of exploration and reasoning. *arXiv preprint arXiv:2506.16499*, 2025.

Lu, C., Lu, C., Lange, R. T., Foerster, J., Clune, J., and Ha, D. The AI scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.

Merchant, A., Batzner, S., Schoenholz, S. S., Aykol, M., Cheon, G., and Cubuk, E. D. Scaling deep learning for materials discovery. *Nature*, 624(7990):80–85, 2023.

Miao, T., Dai, J., et al. PhysMaster: Building an autonomous AI physicist for theoretical and computational physics research. *arXiv preprint arXiv:2512.19799*, 2025.

Nam, J., Yoon, J., Chen, J., Shin, J., Arık, S. Ö., and Pfister, T. Mle-star: Machine learning engineering agent via search and targeted refinement. *arXiv preprint arXiv:2506.15692*, 2025.

OpenAI. Agents SDK and guide. https://platform.openai.com/docs/guides/agents, 2025.

OpenClaw Contributors. OpenClaw: Your own personal AI assistant. https://github.com/openclaw/openclaw, 2025.

Pang, X., Tang, S., Ye, R., et al. BrowseMaster: Towards scalable web browsing via tool-augmented programmatic agent pair. *arXiv preprint arXiv:2508.09129*, 2025.

Phan, L., Gatti, A., Han, Z., Li, N., Hu, M., Pham, C., Rana, Z., Shi, E., Choi, C., Agarwal, M., et al. Humanity's last exam. *Nature*, 2025. arXiv:2501.14249. DOI:10.1038/s41586-025-09962-4.

Skarlinski, M. D., Cox, S., Laurent, J. M., et al. Language agents achieve superhuman synthesis of scientific knowledge. *arXiv preprint arXiv:2409.13740*, 2024.

Swanson, K., Wu, D., et al. Virtual lab: AI agents design new nanobody binders for SARS-CoV-2. *arXiv preprint arXiv:2407.16928*, 2024.

The Royal Swedish Academy of Sciences. The Nobel prize in chemistry 2024: Computational protein design and protein structure prediction. https://www.nobelprize.org/prizes/chemistry/2024/summary/, 2024a.

The Royal Swedish Academy of Sciences. The Nobel prize in physics 2024: Machine learning with artificial neural networks. https://www.nobelprize.org/prizes/physics/2024/summary/, 2024b.

Wang, M., Lin, R., Hu, K., Jiao, J., Chowdhury, N., Chang, E., and Patwardhan, T. FrontierScience: Evaluating AI's ability to perform scientific research tasks. *arXiv preprint arXiv:2601.21165*, 2026. https://openai.com/index/frontierscience/.

Wang, X., Li, B., Song, Y., Xu, F. F., Tang, X., Zhuge, M., Pan, J., Song, Y., Li, B., Singh, J., et al. OpenHands: An open platform for AI software developers as generalist agents. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Wang, X., Rosenberg, S., Michelini, J., Smith, C., Tran, H., Nyst, E., Malhotra, R., Zhou, X., Chen, V., Brennan, R., and Neubig, G. The OpenHands software agent SDK: A composable and extensible foundation for production agents. *arXiv preprint arXiv:2511.03690*, 2025b.

Wei, J., Sun, Z., Papay, S., McKinney, S., Han, J., Fulford, I., Chung, H. W., Passos, A. T., Fedus, W., and Glaese, A. BrowseComp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*, 2025.

Yamada, Y., Lu, C., Lu, C., et al. The AI scientist-v2: Workshop-level automated scientific discovery via agentic tree search. *arXiv preprint arXiv:2504.08066*, 2025.

Yang, X., Yang, X., Fang, S., Zhang, Y., Wang, J., Xian, B., Li, Q., Li, J., Xu, M., Li, Y., et al. R&d-agent: An llm-agent framework towards autonomous data science. *arXiv preprint arXiv:2505.14738*, 2025.

Zhang, L., Chen, S., Cai, Y., Chai, J., Chang, J., Chen, K., Chen, Z. X., Ding, Z., Du, Y., Gao, Y., et al. Bohrium+ scimaster: Building the infrastructure and ecosystem for agentic science at scale. *arXiv preprint arXiv:2512.20469*, 2025.

Zhu, X., Cai, Y., Liu, Z., et al. Toward ultra-long-horizon agentic science: Cognitive accumulation for machine learning engineering. *arXiv preprint arXiv:2601.10402*, 2026.

A. Benchmark Descriptions

This appendix provides detailed descriptions of the four evaluation benchmarks used in our experiments.

Humanity's Last Exam (HLE) (Phan et al., 2025). HLE is one of the most challenging closed-book knowledge assessments ever constructed. It aggregates 2,500 questions across dozens of academic disciplines, with mathematics (41%), biology and medicine (11%), computer science (10%), physics (9%), humanities and social sciences (9%), chemistry (7%), and others, contributed by nearly 1,000 experts from over 500 institutions across 50 countries. Approximately 14% of questions are multimodal (requiring image understanding), while 24% are multiple-choice and the remainder are short-answer exact-match.

MLE-Bench (Chan et al., 2024). MLE-Bench curates 75 real machine learning engineering competitions from Kaggle spanning 15 diverse categories including NLP, computer vision, and signal processing. Agents must complete the full ML pipeline including data processing, feature engineering, model selection, training, and submission. The evaluation metric is the overall average any medal rate (achieving at least bronze medal performance on the competition leaderboard).

BrowseComp (Wei et al., 2025). BrowseComp comprises 1,266 complex web information retrieval tasks. BrowseComp requires agents to persistently navigate tens or hundreds of web pages to locate entangled information: facts that cannot be found with a single query but demand creative, multi-step browsing strategies and cross-source validation. The benchmark is intentionally designed to test an agent's ability to adapt search strategies, synthesize information across sources, and persist through challenging retrieval scenarios.

FrontierScience (Wang et al., 2026). FrontierScience evaluates AI agents on frontier scientific reasoning across three natural science disciplines: physics, chemistry, and biology. It comprises two complementary tracks: the Olympiad track contains 100 problems designed by international competition gold medalists (at IPhO, IChO, and IBO level), evaluated via short-answer exact-match; the Research track contains 60 original research subtasks designed by PhD scientists, graded on a 10-point rubric for scientific rigor and methodology. FrontierScience targets natural science reasoning, a domain particularly relevant to Agentic Science.

B. Limitations

While EvoMaster represents a substantial step toward scalable agentic science, we acknowledge the following limitation in its current scope:

- **Integration with physical environments.** At present, EvoMaster is primarily optimized for *in silico* and computational research workflows. It lacks native support for executing tasks that require direct manipulation of physical experimental apparatuses, such as automated cloud labs or robotic synthesis hardware. Expanding the `Session` abstraction to bridge this gap and interface seamlessly with standard laboratory automation protocols remains a vital avenue for future research.