DETECTING SCALING FACTORS BEYOND THE MODEL: A REPORTING FRAMEWORK FOR AI AGENT SYSTEMS

Kenta Kitamura
Intellectual Moonshine
Miyazaki, Japan
kitamura.kenta.1988@gmail.com

ABSTRACT

As AI agents take on increasingly complex tasks, it becomes difficult to determine whether their progress results from model scaling or environmental design. Performance in agentic systems often arises from improvements in the workspace and accessible knowledge rather than model inference alone. This phenomenon is particularly evident in mathematical reasoning applied to unsolved problems. Recent reports on solving these problems highlight the critical role of subgoal decomposition and formalization. These processes enable an agent to break down complex claims into verifiable steps while using a proof assistant such as Lean for rigorous verification. Such capabilities are heavily influenced by the specific tools and external information available to the system. To address this confounding issue, we propose the Model, Scaffold, and World Reference (MSW) framework to decompose agent performance into three distinct domains. We also introduce a reporting template based on the four perspectives of Identity, Policy, Budget, and Trace (IPBT) to enhance transparency and reproducibility. By auditing publicly available reports claiming that AI systems have solved unsolved mathematical problems, we show that MSW-IPBT highlights missing information that is necessary to distinguish model contributions from environmental support. This study provides guidance for examining claims that AI systems have solved unsolved problems by distinguishing contributions of model improvements from contributions due to environmental scaling.

1 INTRODUCTION

As AI agents evolve to solve increasingly hard tasks, it becomes difficult to distinguish whether performance breakthroughs stem from model scaling or the design of the surrounding environment (Liu et al., 2024; Yang et al., 2024). Success in agentic systems often relies on improvements in the workspace and accessible world knowledge rather than model inference alone (Zhou et al., 2024; Mialon et al., 2024; Aleithan et al., 2024). To understand what truly drives progress, it is essential to decouple the contributions of the model from its external tools and information access. If these factors are not clearly separated, the true source of an agentic breakthrough remains obscured, making it impossible to determine whether a result indicates a more capable model or simply a more sophisticated environment.

This challenge is also important in the field of mathematics, which has long served as a representative benchmark for measuring AI inference capabilities (Hendrycks et al., 2021; Cobbe et al., 2021; Wang et al., 2026). As traditional problem sets reach saturation, there is a growing shift toward using unsolved mathematical problems as the next frontier for AI evaluation (Glazer et al., 2024; Epoch AI, 2026; Google DeepMind, 2026b). In this domain, success depends heavily on processes such as subgoal decomposition and formalization (Sonoda et al., 2026; Varambally et al., 2026). These methods allow an agent to break down complex claims into verifiable steps and to formalize and verify them in proof assistants such as Lean (De Moura et al., 2015). However, the correctness and effectiveness of subgoal decomposition and formalization depend strongly on the available tools, search scope, and external knowledge, so environmental design can directly shape the final result. Consequently, when reports claim that an AI has solved an unsolved problem, they often conflate model-centric progress with the scaling of the scaffold and world reference.

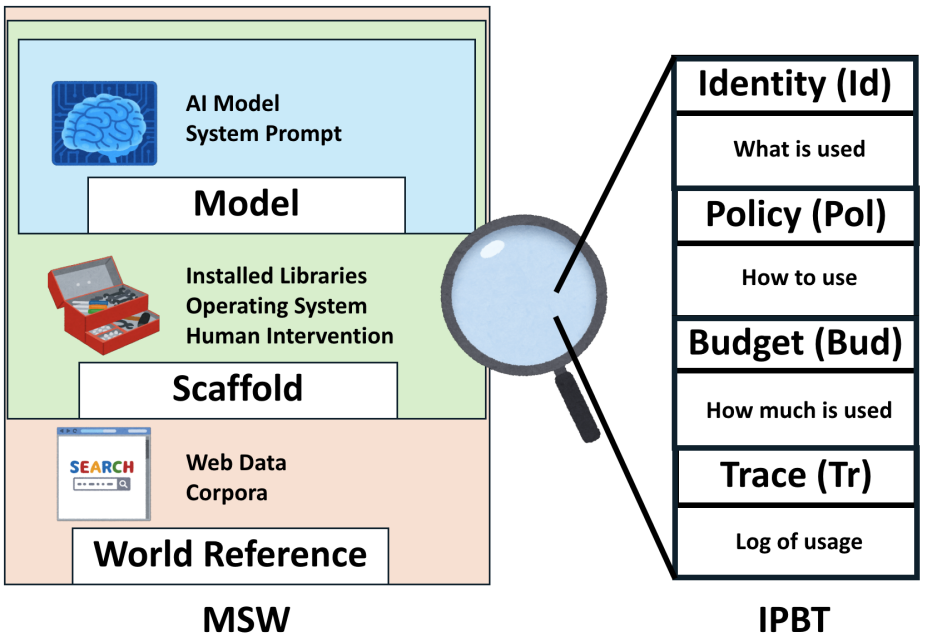

Figure 1: Overview of the MSW-IPBT reporting framework. AI system capabilities are decomposed into Model, Scaffold, and World Reference (MSW), and each domain is audited through the lens of Identity, Policy, Budget, and Trace (IPBT).

1.1 Our contribution

Figure 1 provides an overview of our approach. It separates where capability comes from (MSW; left panel) from what must be disclosed to audit it (IPBT; right panel), which together yield a practical reporting template. In this paper, we make three contributions as follows.

First, in Section 2, to separate model capabilities from environmental factors, we propose the Model, Scaffold, and World Reference (MSW) framework to decompose agent performance into three distinct domains. This framework allows researchers to separate the properties of the model itself from the workbench on which it operates and the external knowledge it references. This framework also corresponds to the concept of decomposing AI agent capabilities in Software Engineering (SWE) (Wong et al., 2026).

Second, in Section 3, we introduce a reporting template based on the four perspectives of Identity, Policy, Budget, and Trace (IPBT) to enhance transparency and reproducibility of AI systems. We call this reporting template MSW-IPBT. The MSW-IPBT reporting template clarifies the comparability and auditability of AI systems.

Third, in Section 4, we demonstrate the utility of the MSW-IPBT framework by auditing publicly available case reports of AI systems claiming to solve unsolved mathematical problems. Our analysis reveals critical gaps in current documentation that prevent a clear distinction between model capabilities and environmental scaling factors.

This paper provides guidance for reframing claims of AI systems solving unsolved mathematical problems as the product of multiple scaling axes, including Model scaling, Scaffold scaling, and World Reference scaling.

2 MSW decomposition of AI system capabilities

It has been pointed out that benchmarks for mathematical problem sets are reaching saturation due to improvements in AI's capabilities (Glazer et al., 2024). Therefore, there is a movement to use

unsolved problems as the next benchmarks for measuring AI's reasoning capabilities (Epoch AI, 2026; Google DeepMind, 2026b).

However, when it comes to unsolved problems, the extent to which an AI's capabilities are counted varies greatly depending on the environment. For unsolved problems, even with the same AI model, the reachability can vary greatly depending on the availability of search, the development of a formalization environment, external memory, verification mechanisms, and the presence or absence of human intervention. As an extreme example, if an AI solves a problem with the guidance of a human expert who can solve the problem without AI, this is an indicator of the environment design and reference access, rather than the AI's logical reasoning ability.

In particular, because AI agents' exploration of unsolved mathematical problems involves complex claims, prior reports have used subgoal decomposition, which divides goals into manageable units, and formalization in a proof assistant, which eliminates ambiguity by translating statements into precise, machine-checkable expressions (Sothanaphan, 2026; Barreto et al., 2026; Chen et al., 2026). Subgoal decomposition extracts the lemmas, special cases, and definitions necessary for the main theorem, clarifying the search order and priorities (Sonoda et al., 2026; Varambally et al., 2026). Formalization removes ambiguity from natural language by translating statements into a formal representation whose correctness can be mechanically checked by a verifier, such as a proof assistant (Azerbayev et al., 2023; Yang et al., 2023). These processes facilitate search progress through repeated generative proposals and formal verification checks. Therefore, when claiming that an AI has solved an unsolved problem, a claim that fails to clarify the environmental and knowledge access conditions under which the solution was reached lacks comparability and reproducibility. It is important not to confuse results with model progress.

To clarify these distinct contributors to performance, we propose decomposing the observed capabilities into three domains consisting of Model, Scaffold, and World Reference (MSW). This classification is consistent with frameworks used to understand AI agents in SWE (Wong et al., 2026) and provides a structured way to interpret mathematical progress by AI.

Model refers to the properties of the AI model itself, such as its weights, and inference settings.

Scaffold is a workbench on which the AI model works, and includes tools, formalization environments, workflows, external memory, and human intervention during and after model execution, as well as orchestration procedures and rules.

World Reference is knowledge from the world outside the scaffold that is referenced during runtime, and includes knowledge on the web, authorized corpora, community artifacts, and literature.

MSW decomposition separates Model progress, Scaffold scaling, and World Reference scaling.

3 MSW-IPBT

While MSW categorizes where agent capabilities reside, identifying these domains alone is not enough to pinpoint the specific scaling levers that drive a breakthrough. To precisely identify the source of progress, a reporting system must clarify four fundamental aspects of the problem-solving process. These aspects include what specific components were used, how they were operated, how many resources were consumed, and whether there are logs to verify the results. To address these requirements, we introduce the four evaluation perspectives of Identity, Policy, Budget, and Trace (IPBT). As shown in Table 1, by mapping these four perspectives across the MSW, the framework provides a structured MSW-IPBT reporting template to isolate the scaling factors.

The IPBT perspectives serve as the essential criteria for reporting transparency and identifying the drivers of success. The Identity perspective specifies what version or name of a component was employed. The Policy perspective defines the rules and settings that determine how that component was operated. The Budget perspective provides a measure of how many resources were consumed. The Trace perspective consists of the logs and records that allow third parties to inspect the actual execution and verify the findings. With reporting based on this systematic approach, a claim that an AI system solved an open problem goes beyond a bare report of success and is accompanied by a transparent explanation of the factors that enabled the result.

Table 1: MSW-IPBT checklist. Each factor is decomposed into Model, Scaffold, World Reference (MSW) and annotated along Identity, Policy, Budget, Trace (IPBT).

Key field	Description	Example
Model (M)		
M-Id	Model Identification	Model Name, Version
M-Pol	Model Usage Settings	Temperature/Top-p, System Prompt
M-Bud	Model Usage Budget	Number of Trials, Max_tokens, API cost
M-Tr	Model Usage Logs	Generation Log, Execution Procedure Record
Scaffold (S)		
S-Id	Scaffold Identification	Formalization Environment, Installed Libraries, Execution Platform
S-Pol	Scaffold Usage Settings	Generation/Verification/Modification Procedures, File Editing Units, Human Intervention Range
S-Bud	Scaffold Usage Budget	Search Iteration Count, Search Depth, CPU/GPU Time
S-Tr	Scaffold Usage Logs	Public Repository, PR/Commit History, Build Log
World Reference (W)		
W-Id	World Reference Identification	Snapshot Date, Permitted Corpus, Accessible Domains
W-Pol	Reference Access Settings	Web Search Availability, Browser Method, Search Query Constraints
W-Bud	Reference Access Budget	Search Limit, Number of Retrieved Documents, Rate Limit, API Credits
W-Tr	Reference Access Logs	Reference URL, Timestamp of Access, List of Retrieved Items

In the Model domain, these are represented as Model-Identity (M-Id), Model-Policy (M-Pol), Model-Budget (M-Bud), and Model-Trace (M-Tr). M-Id is the model identification. M-Pol is the model usage settings. M-Bud is the model usage budget. M-Tr is the model usage logs.

In the Scaffold domain, these are represented as Scaffold-Identity (S-Id), Scaffold-Policy (S-Pol), Scaffold-Budget (S-Bud), and Scaffold-Trace (S-Tr). S-Id is the scaffold identification. S-Pol is the scaffold usage settings. S-Bud is the scaffold usage budget. S-Tr is the scaffold usage logs.

In the World Reference domain, these are represented as World Reference-Identity (W-Id), World Reference-Policy (W-Pol), World Reference-Budget (W-Bud), and World Reference-Trace (W-Tr). W-Id is the World Reference identification. W-Pol is the reference access settings. W-Bud is the reference access budget. W-Tr is the reference access logs.

Concrete examples for each field are provided in Table 1.

4 Case Report

In this section, we use the MSW-IPBT reporting template to audit published cases in which AI has been used to solve unsolved mathematical problems. Through this audit, we demonstrate the feasibility of the MSW-IPBT framework.

4.1 Target Cases

The subjects of the audit were reports of Erdős Problem 728 (E728) (Sothanaphan, 2026), Erdős Problem 1051 (E1051) (Barreto et al., 2026), and Fel's conjecture (Fel) (Chen et al., 2026). These three case reports were independently authored and published by different groups. All problems are reported as solved by AI systems.

Table 2: Checklist for case report by MSW-IPBT reporting template.

Key field	Contents
M-Id	Model Name
M-Pol	Generation Conditions
M-Bud	Generation Budget
M-Tr	Generation Log
S-Id	System Configuration
S-Pol	Workflow, Human Intervention
S-Bud	System Usage Budget
S-Tr	Execution Log
W-Id	Reference Information
W-Pol	Reference Method
W-Bud	Reference Limit
W-Tr	Reference Log

Table 3: Audit results for three case reports across 12 MSW-IPBT fields. $\times$ indicates that the report contains no mention of the corresponding field, and ✓ indicates otherwise.

Key field	E728	E1051	Fel
M-Id	✓	✓	✓ or $\times$
M-Pol	✓	$\times$	$\times$
M-Bud	$\times$	$\times$	$\times$
M-Tr	✓	✓	✓
S-Id	✓	✓	✓
S-Pol	✓	✓	✓
S-Bud	$\times$	$\times$	$\times$
S-Tr	✓	$\times$	$\times$
W-Id	✓	$\times$	$\times$
W-Pol	✓	✓	$\times$
W-Bud	$\times$	$\times$	$\times$
W-Tr	✓	$\times$	$\times$

4.2 MSW-IPBT Checklist for case reports

In the case reports, we organize the operational audit checklist into 12 items, as shown in Table 2. These items constitute an instantiation designed to assess the feasibility of auditing with MSW-IPBT, rather than an exhaustive reporting standard.

In the Model domain, the IPBT fields are defined as follows: M-Id denotes the model name, M-Pol denotes generation conditions, M-Bud denotes the generation budget, and M-Tr denotes the generation log.

In the Scaffold domain, S-Id denotes system configuration, S-Pol denotes the workflow and human intervention, S-Bud denotes the system usage budget, and S-Tr denotes the execution log.

In the World Reference domain, W-Id denotes reference information, W-Pol denotes the reference method, W-Bud denotes the reference limit, and W-Tr denotes the reference log.

For each of the 12 fields, we assign $\times$ when the report contains no mention of the corresponding information, and ✓ otherwise.

4.3 Case Report Results

We audit reports of E728, E1051, and Fel based on the 12 key fields in Table 2. Table 3 shows the audit results.

4.3.1 CASE 1: E728

In the E728 report (Sothanaphan, 2026), the authors report a workflow that combines GPT-5.2 Pro and Aristotle. The interaction was conducted via the ChatGPT interface and the full conversation log is publicly available.

M-Id is ✓ because the publicly released log explicitly identifies the model used (GPT-5.2 Pro), which satisfies the requirement of model name/version disclosure. M-Pol is ✓ because the conversation record reveals how the model was operated in practice, providing observable generation conditions. M-Bud is × because the report does not specify quantitative usage, which is required for a generation budget. M-Tr is ✓ because the complete ChatGPT conversation log functions as a generation trace that enables third-party inspection of the produced outputs and the sequence of interactions.

S-Id is ✓ because the report and log describe the execution setting of the workflow. S-Pol is ✓ because the workflow can be reconstructed from the log. S-Bud is × because the report does not quantify scaffold-side resources. S-Tr is ✓ because the conversation record documents scaffold execution at the level of observable actions.

W-Id is ✓ because the log indicates that external references were used. W-Pol is ✓ because the log shows the method of reference access in operation. W-Bud is × because no explicit limits are reported. W-Tr is ✓ because the log records the referenced items within the conversation context.

4.3.2 CASE 2: E1051

In the E1051 report (Barreto et al., 2026), the authors report the use of the AI agent Aletheia, which is based on Gemini Deep Think (Luong & Mirrokni, 2026). The report characterizes the project as a roughly equal collaboration between the AI system and human contributors, and it links to a GitHub repository containing the model's inputs and outputs (Google DeepMind, 2026a).

M-Id is ✓ because the report identifies the agent system name as Aletheia and states that it is based on Gemini Deep Think. M-Pol is × because the report does not disclose generation conditions. M-Bud is × because no quantitative model-usage budget is reported. M-Tr is ✓ because the report links to public GitHub artifacts that expose at least part of the intermediate work products.

S-Id is ✓ because a rough workflow for the generation, verification, and revision of AI has been published, stating that the contribution is roughly evenly split between human researchers and the AI. S-Pol is ✓ because a workflow for the use of AI has been published. S-Bud is × because there is no description of the system usage budget. S-Tr is × because there is no description of the execution log.

W-Id is × because there is no description of the reference information. W-Pol is ✓ because it shows that a web search was conducted during the process (Luong & Mirrokni, 2026). W-Bud is × because there is no description of the reference access budget. W-Tr is × because there is no description of the reference log.

4.3.3 CASE 3: FEL

In the Fel report (Chen et al., 2026), the authors present a case study in which AxiomProver translates natural-language problem descriptions into Lean formalizations to produce formal proofs. The report provides relatively clear input specifications and output artifacts in a public GitHub repository.

M-Id is ✓ or ×, because while AxiomProver is listed as the system name, it is unclear whether this specifies the underlying model itself. If AxiomProver is the model name, M-Id is ✓, but if not, M-Id is ×. M-Pol is × because the specific conditions of the model are not listed. M-Bud is × because there is no description regarding the generation budget. M-Tr is ✓ because a GitHub output log is provided.

S-Id is ✓ because the Lean version and operating system (OS) are published on GitHub. S-Pol is ✓ because the workflow is published as code on GitHub and further described in the report. S-Bud is × because the system usage budget is not listed. S-Tr is × because the execution log is not listed.

W-Id is $\times$ because the specific reference information used is not listed. W-Pol is $\times$ because it is not mentioned. W-Bud is $\times$ because the reference access budget is not listed. W-Tr is $\times$ because the reference log is not listed.

5 DISCUSSION

5.1 LIMITATIONS

The limitations of this study are as follows.

First, this paper focuses primarily on the mathematical domain. While evaluation using MSW-IPBT may be applicable to fields other than mathematics, mathematics has unique features not found in other fields. For example, rigorous verification is possible in mathematics, and powerful verification tools such as Lean are available (De Moura et al., 2015). When extending this paper's framework to other fields, differences in verification costs and the specificities of mathematics must be taken into account.

Second, the case report in this paper evaluated whether the results could be traced from publicly available information. Namely, we focus on comparability, reproducibility, and auditability. Evaluation of mathematical value or novelty is beyond the scope of this paper.

Third, the evaluation in the case reports is based solely on the presence or absence of specific evaluation items for each item in the MSW-IPBT checklist. These confirmations are influenced by the selection of each evaluation item. The list in Table 2 is only an example of an evaluation method. It is possible to adopt a more detailed evaluation or select different items. Identifying which items are appropriate for which cases remains an important direction for future work.

5.2 INSIGHTS FROM CASE REPORTS

We discuss insights gained from the audit of the three cases.

In the E728 case (Sothanaphan, 2026), the audit revealed the highest level of transparency among the subjects. As shown in Table 3, almost all fields across the Model, Scaffold, and World Reference domains were disclosed, except for the budget items. The availability of public dialogue logs (M-Tr, S-Tr, W-Tr) allows third parties to infer the agentic workspace and the external knowledge accessed. This transparency makes it possible to distinguish the contributions of human intervention from the model's own reasoning, although the lack of budget data (M-Bud, S-Bud, W-Bud) prevents a quantitative analysis of scaling efficiency.

In the E1051 case (Barreto et al., 2026), the report disclosed the model identity and outputs (M-Id, M-Tr), the scaffold identity and a high-level description of the procedure (S-Id, S-Pol), and part of the world-access policy (W-Pol), but several fields critical for reproducibility and auditability were missing. Specifically, the report did not provide detailed inference settings (M-Pol), did not disclose which external world references were referenced (W-Id), and did not release execution traces for the scaffold and world access (S-Tr, W-Tr). Moreover, none of the budget fields were reported (M-Bud, S-Bud, W-Bud). This lack of disclosure hinders the research community's ability to independently verify the results and to conduct a fine-grained audit of environmental scaling.

In the Fel case (Chen et al., 2026), the model attribution (M-Id) is ambiguous. It is unclear whether the reported success should be attributed to a specific LLM version or to the integrated AxiomProver system as a whole. The inference configuration (M-Pol) is not disclosed, while the part of the model outputs (M-Tr) is provided. On the scaffold side, the formalization setup and workflow are described (S-Id, S-Pol), but execution traces are not reported (S-Tr). No information is provided about world reference, including the identity of the referenced world, the access policy, or access traces (W-Id, W-Pol, W-Tr). Budget information is also absent across all components (M-Bud, S-Bud, W-Bud). These missing disclosures hinder reproducible verification and prevent fine-grained auditing of environmental scaling.

A critical finding across all three cases is the complete lack of budget reporting (M-Bud, S-Bud, and W-Bud). Since these factors often define the ceiling of an agent's capability, their omission makes it difficult to determine whether a breakthrough is driven by an improvement in the model's

inherent reasoning power or simply by an increase in the search depth, computational budget, or human-guided iterations.

5.3 Scaling Detection with MSW-IPBT

We discuss the possibility of Scaffold scaling and World Reference scaling. Regarding Model scaling, pre-training scaling and test-time scaling are well-established, and have been confirmed to effectively improve AI capabilities (Hoffmann et al., 2022; Snell et al., 2025). In this study, Scaffold scaling and World Reference scaling were presented as scaling methods other than models that contribute to subgoal decomposition and formalization in mathematics.

Scaffold scaling refers to the phenomenon of increasing the reachability of a solution through design, including tool integration, formalization, workflow, external memory, and validation. External memory provides a concrete example. Artifacts such as work logs and policy memos, including Agents.md (Lulla et al., 2026), provide external memory that improves state management and reuse (Wong et al., 2026). During exploration, it reduces repeated attempts by recording what has already been tried. During formalization, it preserves verified facts, adopted definitions, and failed approaches in a form that can be reused. During verification, it helps diagnose and avoid recurring execution failures by keeping build and runtime traces. Verification design is another key factor. In mathematics, strong proof assistants such as Lean are common (De Moura et al., 2015), but verification can also rely on testing, type checking, or voting-based selection. Lean enables rigorous checking once statements are expressed in its formal language (De Moura et al., 2015), but it may be unavailable in some settings, and formalization itself can be costly. As a result, a system must decide which claims to formalize, when to invoke verification, and how much verification to perform under a finite budget. These policy and budget choices can substantially change the effective reachability of solutions.

World Reference scaling refers to the phenomenon where the reachability of solutions changes as external knowledge is updated. For mathematical unsolved problems, the world is mutable and changes rapidly. New papers, forum discussions, and expansion of formalized libraries can rewrite the outcome. For instance, a problem statement may be revised or reinterpreted, a key lemma may be formalized and added to a library, or curated datasets of definitions and known lemmas may become available. Without disclosing the World References used, it is impossible to determine whether an apparent success reflects improved model inference or simply stronger and fresher external knowledge. Therefore, when unsolved problems are used as benchmarks, World Reference should be treated as an experimental condition that must be specified, rather than an uncontrolled external factor.

6 Related work

6.1 Interactive Agent Evaluation Benchmarks

In recent years, numerous benchmarks have been proposed to evaluate large language models (LLMs) not simply as question-answering systems, but as agents that interact with the environment to achieve their goals. For example, AgentBench presents a framework for evaluating LLMs-as-Agents across multiple environments (Liu et al., 2024). WebArena uses a validator to determine task success in a self-hosted, realistic web environment (Zhou et al., 2024). Mind2Web also provides web operation data (Deng et al., 2023), and ToolLLM evaluates tool usage skills (Qin et al., 2024). In the SWE domain, SWE-bench defines success criteria as code editing and passing tests using real GitHub issues (Jimenez et al., 2024), and SWE-agent demonstrates the impact of agent-facing interface (ACI) design on performance (Yang et al., 2024). Furthermore, GAIA aims to measure real-world skills (Mialon et al., 2024).

However, many of these approaches assume tasks with fixed success conditions and verification procedures. In settings where success conditions themselves are fluid, such as unsolved problems, and the reference world can be updated during the evaluation period, even the same model can achieve significantly different results depending on design differences in search, formalization, external memory, and human intervention. Furthermore, in real-task benchmarks, including SWE-bench, it has been pointed out that leaks and weaknesses in the verifier can inflate apparent success rates (Aleithan et al., 2024). In contrast to these benchmarks that rely on fixed success criteria and

potentially fragile verifiers, we focus on open mathematical problems where both the target and the reference world can shift over time. To make results comparable under such fluid conditions, we use MSW to separate model capability from Scaffold and World Reference effects, and we propose MSW-IPBT as a reporting framework to document these factors.

6.2 Verifier-Based Search and Formal Mathematics

In AI for Mathematics, there has been a growing body of research using proof assistants as verifiers to advance search by cycling between generation and verification (Varambally et al., 2026; Chen et al., 2026). MiniF2F is a benchmark spanning multiple formal systems (Zheng et al., 2022). ProofNet enabled the evaluation of autoformalization by mapping natural language theorems and proofs to Lean formalizations (Azerbayev et al., 2023). LeanDojo integrates Lean environments, data, benchmarks, and tools to provide a reproducible experimental platform (Yang et al., 2023). Furthermore, a separate line of work aims to improve formal proof performance via pre-training on mathematical corpora and tool use, including mathematics-focused LLMs such as Llemma (Azerbayev et al., 2024).

However, the performance of verifier-assisted search is not determined solely by the model. Results can vary significantly depending on the version of the library such as mathlib (The mathlib Community, 2020), the target corpus, the search strategy, the number of iterations and stopping conditions, and even the presence or absence of human intervention. This effect is especially pronounced for unsolved problems, where the available literature, arguments, and formalized lemmas can change over time, making it harder to disentangle model inference from environment updates. This paper differs from existing research by decomposing these dependencies for mathematical unsolved problems into MSW, and by providing the MSW-IPBT reporting template that standardizes disclosure across reports.

6.3 Disclosure Templates

In machine learning, templates have been proposed to encourage disclosure of models and datasets. Model Cards provide a framework for documenting a model's intended use, evaluation conditions, and limitations (Mitchell et al., 2019). Datasheets for Datasets suggest recording the composition, collection, and recommended uses of a dataset (Gebru et al., 2021). Reproducibility checklists have helped to increase the reproducibility of experimental conditions (Pineau et al., 2021).

On the other hand, when AI agents tackle unsolved problems in mathematics, performance can be influenced by factors external to the model, such as runtime context and the external environment, as well as tools, workflow, and human intervention, yet existing templates rarely prioritize these aspects. This paper expands the existing reproducibility framework by positioning external factors as MSW and specifying reporting requirements along the axes of IPBT.

7 Conclusions

This paper addresses a central ambiguity in reported breakthroughs on unsolved mathematical problems by AI systems. It is often unclear how much of an apparent success should be attributed to the underlying model and how much to the surrounding agent system. To cope with this ambiguity, we introduce the MSW framework, which separates contributions from the Model, Scaffold, and World Reference. We also propose an MSW-IPBT reporting template, which specifies what information should be disclosed to support transparent attribution. Finally, through audits of three publicly available case reports, we demonstrate that MSW-IPBT can be applied in practice and that it exposes concrete documentation gaps.

This paper provides guidance for interpreting reported breakthroughs without conflating model progress with gains from Scaffold scaling and World Reference scaling.

LLM Usage Statement

We would like to acknowledge the assistance of LLMs for English grammar and word choice adjustments. The content has been fully reviewed, and the authors remain responsible.

REFERENCES

Reem Aleithan, Haoran Xue, Mohammad Mahdi Mohajer, Elijah Nnorom, Gias Uddin, and Song Wang. Swe-bench+: Enhanced coding benchmark for llms. *arXiv preprint arXiv:2410.06992*, 2024.

Zhangir Azerbayev, Bartosz Piotrowski, Hailey Schoelkopf, Edward W Ayers, Dragomir Radev, and Jeremy Avigad. Proofnet: Autoformalizing and formally proving undergraduate-level mathematics. *arXiv preprint arXiv:2302.12433*, 2023.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen Marcus McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. Llemma: An open language model for mathematics. In *International Conference on Learning Representations*, 2024.

Kevin Barreto, Jiwon Kang, Sang-hyun Kim, Vjekoslav Kovač, and Shengtong Zhang. Irrationality of rapidly converging series: a problem of erdős and graham. *arXiv preprint arXiv:2601.21442*, 2026.

Evan Chen, Chris Cummins, GSM, Dejan Grubisic, Leopold Haller, Letong Hong, Andranik Kurghinyan, Kenny Lau, Hugh Leather, Seewoo Lee, Aram Markosyan, Ken Ono, Manooshree Patel, Gaurang Pendharkar, Vedant Rathi, Alex Schneidman, Volker Seeker, Shubho Sengupta, Ishan Sinha, Jimmy Xin, and Jujian Zhang. Fel's conjecture on syzygies of numerical semigroups. *arXiv preprint arXiv:2602.03716*, 2026.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Leonardo De Moura, Soonho Kong, Jeremy Avigad, Floris Van Doorn, and Jakob von Raumer. The lean theorem prover (system description). In *International Conference on Automated Deduction*, 2015.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2Web: Towards a generalist agent for the web. In *Advances in Neural Information Processing Systems*, 2023.

Epoch AI. Frontiermath: Open problems, 2026. URL `https://epoch.ai/frontiermath/open-problems`. Accessed: 2026-02-17.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64 (12):86–92, 2021.

Elliot Glazer, Ege Erdil, Tamay Besiroglu, Diego Chicharro, Evan Chen, Alex Gunning, Caroline Falkman Olsson, Jean-Stanislas Denain, Anson Ho, Emily de Oliveira Santos, Olli Järviniemi, Matthew Barnett, Robert Sandler, Matej Vrzala, Jaime Sevilla, Qiuyu Ren, Elizabeth Pratt, Lionel Levine, Grant Barkley, Natalie Stewart, Bogdan Grechuk, Tetiana Grechuk, Shreepranav Varma Enugandla, and Mark Wildon. Frontiermath: A benchmark for evaluating advanced mathematical reasoning in ai. *arXiv preprint arXiv:2411.04872*, 2024.

Google DeepMind. Bkkkz26.pdf, 2026a. URL `https://github.com/google-deepmind/superhuman/blob/main/aletheia/BKKKZ26/BKKKZ26.pdf`. Accessed 2026-02-20.

Google DeepMind. Formal conjectures documentation, 2026b. URL `https://google-deepmind.github.io/formal-conjectures/`. Accessed: 2026-02-17.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, 2022.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, 2024.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. Agentbench: Evaluating llms as agents. In *International Conference on Learning Representations*, 2024.

Jai Lal Lulla, Seyedmoein Mohsenimofidi, Matthias Galster, Jie M Zhang, Sebastian Baltes, and Christoph Treude. On the impact of agents. md files on the efficiency of ai coding agents. *arXiv preprint arXiv:2601.20404*, 2026.

Thang Luong and Vahab Mirrokni. Accelerating mathematical and scientific discovery with Gemini Deep Think. `https://deepmind.google/blog/accelerating-mathematical-and-scientific-discovery-with-gemini-deep-think/`, 2026. Accessed: 2026-02-20.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. Gaia: a benchmark for general ai assistants. In *International Conference on Learning Representations*, 2024.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*, 2019.

Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché-Buc, Emily Fox, and Hugo Larochelle. Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program). *Journal of machine learning research*, 22(164):1–20, 2021.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. Toolllm: Facilitating large language models to master 16000+ real-world apis. In *International Conference on Learning Representations*, 2024.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling parameters for reasoning. In *International Conference on Learning Representations*, 2025.

Sho Sonoda, Shunta Akiyama, and Yuya Uezato. Don't eliminate cut: Exponential separations in llm-based theorem proving. *arXiv preprint arXiv:2602.10512*, 2026.

Nat Sothanaphan. Resolution of Erdős problem #728: a writeup of Aristotle's Lean proof. *arXiv preprint arXiv:2601.07421*, 2026.

The mathlib Community. The Lean Mathematical Library. In *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs*, 2020.

Sumanth Varambally, Thomas Voice, Yanchao Sun, Zhifeng Chen, Rose Yu, and Ke Ye. Hilbert: Recursively building formal proofs with informal reasoning. In *International Conference on Learning Representations*, 2026.

Peng-Yuan Wang, Tian-Shuo Liu, Chenyang Wang, Yi-Di Wang, Shu Yan, Cheng-Xing Jia, Xu-Hui Liu, Xin-Wei Chen, Jia-Cheng Xu, Ziniu Li, and Yang Yu. A survey on large language models for mathematical reasoning. *ACM Computing Surveys*, 58(8):1–35, 2026.

Sherman Wong, Zhenting Qi, Zhaodong Wang, Nathan Hu, Samuel Lin, Jun Ge, Erwin Gao, Wenlin Chen, Yilun Du, Minlan Yu, and Ying Zhang. Confucius code agent: Scalable agent scaffolding for real-world codebases. *arXiv preprint arXiv:2512.10398*, 2026.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-agent: Agent-computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems*, 2024.

Kaiyu Yang, Aidan M. Swope, Alex Gu, Rahul Chalamala, Peiyang Song, Shixing Yu, Saad Godil, Ryan J. Prenger, and Anima Anandkumar. LeanDojo: Theorem proving with retrieval-augmented language models. In *Advances in Neural Information Processing Systems*, 2023.

Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. Minif2f: a cross-system benchmark for formal olympiad-level mathematics. In *International Conference on Learning Representations*, 2022.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, 2024.